ECAI 2008
M. Ghallab et al. (Eds.)
IOS Press, 2008

doi:10.3233/978-1-58603-891-5-388

Optimal Coalition Structure Generation In Partition Function Games

Tomasz Michalak, Andrew Dowell, Peter McBurney and Michael Wooldridge
Department of Computer Science,
The University of Liverpool,
L69 3BX
Email: {tomasz, adowell, mcburney, mjw} @ liv.ac.uk

Abstract. [1] In multi-agent systems (MAS), coalition formation is typically studied using *characteristic function game* (CFG) representations, where the performance of any coalition is independent from co-existing coalitions in the system. However, in a number of environments, there are significant *externalities from coalition formation* where the effectiveness of one coalition may be affected by the formation of other distinct coalitions. In such cases, coalition formation can be modeled using *partition function game* (PFG) representations. In PFGs, to accurately generate an optimal division of agents into coalitions (so called CSG problem), one would have to search through the entire search space of coalition structures since, in a general case, one cannot predict the values of the coalitions affected by the externalities *a priori*. In this paper we consider four distinct PFG settings and prove that in such environments one can bound the values of every coalition. From this insight, which bridges the gap between PFG and CFG environments, we modify the existing state-of-the-art anytime CSG algorithm for the CFG setting and show how this approach can be used to generate the optimal CS in the PFG settings.

1 Introduction & Motivation

In multi-agent systems (MAS), coalition formation occurs when distinct autonomous agents group together to achieve something more efficiently than they could accomplish individually. One of the main research issues in co-operative MAS is to determine which division of agents into disjoint coalitions (*i.e.* a coalition structure (CS)) maximizes the total payoff of the system [12, 10]. To this end, coalition formation is typically studied using *characteristic function game* (CFG) representations which consist of a set of agents A and a characteristic function v, which takes, as input, all feasible coalitions $C \subseteq A$ and outputs numerical values reflecting how these coalitions perform. Furthermore, it is assumed that the performance of any coalition is independent from co-existing coalitions in the system. In other words, the value of a coalition C in a structure CS has the same value as it does in another distinct structure CS'. Based on this characteristic of CFGs, Rahwan *et al.* [10] proposed an algorithm that usually generates an optimal CS without searching through the entire space of CSs.

In many real life MAS environments, CFG representations are sufficient to model coalition formation, as the coalitions either do not interact with each other while pursuing their own goals or because such interactions are small enough to be neglected. However, in a number of other environments, there are significant *externalities from coalition formation* (henceforth externalities) where the performance of one coalition may be affected by the formation of another distinct coalition. For example, as more commercial activity moves to the internet, we can expect online economies to become increasingly sophisticated, as is happening, for instance, with real time electronic purchase of wholesale telecommunications bandwidth or computer processor resources. In such contexts, *ad hoc* coalition formation will need to allow for coalition externalities, thus, rendering CFG representation inadequate to model coalition formation. In contrast, externalities are accounted for in the *partition function game* (PFG) representation. A PFG consists of a set of agents A and a partition function which takes, as input, every feasible coalition structure (CS), and for each coalition in each structure, outputs a numerical value that reflects the performance of the coalition in that structure. Now, the value of a coalition C in a structure CS may not have the same value in another distinct structure CS'. This means that it is not generally possible to pre-determine the value of a coalition in a certain CS without actually computing it in this specific CS. Consequently, one must search through the entire space of CSs to guarantee an optimal solution. This presents a major computational challenge as, even for a moderate number of agents, there are billions of structures to search through (for example, for 14 agents there are $190,899,322$ CSs and for 15 agents there are $1,382,958,545$ CSs).

In this paper we contribute to the literature as follows:

- We prove that it is possible to bound the coalition values in two commonly used PFG settings, thus bridging the gap between PFG and CFG environments;
- We show that our theorems regarding bounded values can be used to modify the existing state-of-the-art CSG algorithm for the CFG settings. Consequently, our new algorithm can be applied to generate the optimal CS in these PFG settings;
- Using numerical simulations we demonstrate the effectiveness of our approach which, in a number of cases, is comparable to results obtained for the CFG setting.

Much research effort has been directed at optimal CS generation in the CFG setting. Sandholm *et al.* [12] proposed a new way to rep-

[1] The authors are grateful for financial support received from the UK EPSRC through the project *Market-Based Control of Complex Computational Systems* (GR/T10657/01). The authors are also thankful to Jennifer Mcmanus, School of English, University of Liverpool for excellent editorial assistance.

resent the entire set of CSs in the form of a *coalition structure tree*. For this representation, they developed an algorithm which generates CS values within a finite bound from the optimal value for the entire system. It initially searches the two lowest rows of the tree and then searches from the top downwards either until the whole space has been searched or the running time of the algorithm has expired. Based on this representation, Dang and Jennings proposed a much faster algorithm which, after performing the same initial step as that of Sandholm *et al.,* then searches exclusively through particular coalition structures in the remaining space [3]. Nevertheless, both solutions have drawbacks; notably, that the worst case bounds they provide are relatively low and that both the algorithms must always search the whole space in order to guarantee an optimal solution.

To circumvent these problems Rahwan *et al.* recently proposed a more efficient anytime CSG algorithm for the CFG setting [10]. Using a novel representation of the search space, this algorithm is significantly faster than its existing counterparts. The input to the algorithm are coalitions lists structured according to the distributed coalitional value calculation (DCVC) algorithm presented in [9].

In contrast, in the field of economics, much research has been directed at coalition formation in PFG settings. Particular efforts have been undertaken towards computing both the Shapley value and the core solution in such settings [7, 4]. Furthermore, PFGs have been used to represent coalition formation in many practical applications, such as fisheries on the high seas [8], fuel emissions reduction [5] or Research & Development (R&D) cooperation between firms [2]. Both of the former settings are examples of games with positive externalities, where the decision by one group of countries to reduce fishing activities or fuel emissions may have a positive impact on other countries. In contrast, a R&D cooperation between a group of companies could be modeled using a game with negative externalities since the market positions of some companies could be hindered by the increased competitiveness resulting from a collusion of other companies. An excellent overview of both CFG and PFG approaches in economics is provided in [1].

2 Partition Function Games

For a set of agents $A = \{a_1, \ldots, a_n\}$ and a coalition $C \subseteq A$, a PFG generates a non-negative integer value $v(C; CS)$, where CS is a coalition structure of A and $C \in CS$. Following Halfalir [6], a PFG is said to have weak *positive externalities* if for every three subsets $C, S, T \subseteq A$ where $C \cap S \cap T \neq \emptyset$ and for any structure CS' of $A \setminus (S \cup T \cup C)$ then:

$$v(C; \{S \cup T, C\} \cup CS^{'}) \geq v(C; \{S, T, C\} \cup CS^{'}).$$

In the case where the inequality is $\leq$ the PFG is said to exhibit weak *negative externalities*. Intuitively, this property means that a game has *positive* (respectively, *negative*) *externalities* if a merger between two coalitions makes every other coalition better (worse) off. Furthermore, a PFG is weakly *super-additive (sub-additive)* if for any $S, T \subseteq A$ with $S \cap T \neq \emptyset$ and structure CS' of $A \setminus S \cup T$ then:

$$v(S \cup T; \{S \cup T\} \cup CS^{'}) \;\geq\; (\leq) v(S; \{S, T\} \cup CS^{'}) + v(T; \{S, T\} \cup CS^{'}).$$

Intuitively, this means that a PFG is super-additive (sub-additive) if two coalitions C_i and C_j in a structure, say $CS = C_1, \ldots C_i, C_j, \ldots, C_k$, join together to form coalition $C' = C_i \cup C_j$ then the value of C' in the structure $CS' = C', C_1, \ldots C_{i-1}, C_{i+1}, C_{j-1}, C_{j+1}, \ldots, C_k$ is at least (at most) as large as the sum of the values of C_i and C_j in CS. Classic results in game theory tell us that for super-additive CFGs (where for any two disjoint coalitions S, T $v(S \cup T) \geq v(S) + v(T)$) the optimal CS is the grand coalition (*i.e.* the coalition containing every agent in the system), whereas in sub-additive CFGs (where for any two disjoint coalitions S, T $v(S \cup T) \leq v(S) + v(T)$) the optimal structure is the *CS of singletons, i.e.* the structure where all the agents act as individuals.[2] We now show, with the aid of an example (taken from [6]), that this does not necessarily hold in a super- (sub-) additive PFG setting. Consider the following super-additive PFG for $A = \{1, 2, 3\}$, where, in addition, there are negative externalities:

- $v((i); \{(1), (2), (3)\}) = 4$ for $i = 1, 2, 3$;
- $v((j, k); \{(i), (j, k)\}) = 9$ and $v((i); \{(i), (j, k)\}) = 1$ for all $i, j, k \in A$ where $i \neq j \neq k$; and
- $v(A; \{A\}) = 11$.

Clearly, the super-additive requirement is met but the grand coalition is not the optimal structure since $v(A; \{A\}) = 11 < \sum_{i=1}^{3} v((i);\ \{(1), (2), (3)\}) = 12$. Thus, this example shows that the grand coalition is not always the optimal structure in a super-additive PFG with negative externalities. Equally, for the same A, suppose that the values of the partition function are as follows:

- $v((i); \{(1), (2), (3)\}) = 3$ for $i = 1, 2, 3$;
- $v((j, k); \{(i), (j, k)\}) = 2$ and $v((i); \{(i), (j, k)\}) = 7$ for all $i, j, k \in A$ where $i \neq j \neq k$; and
- $v(A; \{A\}) = 4$.

In this game, the sub-additivity property is met but the CS of singletons is not the optimal CS, due to the positive externalities. This shows that this structure is not always the optimal in sub-additive PFGs with positive externalities. Thus, the classic results from the CFG setting do not always hold in the PFG one. Consequently, in this paper, we shall study four classes of PFG:

1. super-additive games with positive externalities (PF^{+}_{sup});
2. super-additive games with negative externalities (PF^{-}_{sup});
3. sub-additive games with positive externalities (PF^{+}_{sub});
4. sub-additive games with negative externalities (PF^{-}_{sub}).

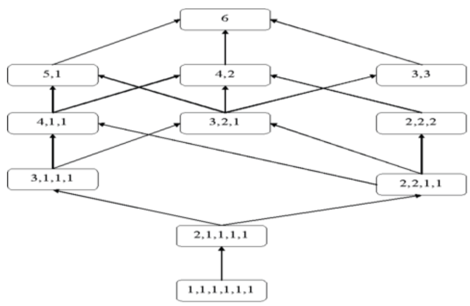

Figure 1: Paths for a six agent setting

The Sandholm *et al.* tree representation of the CS space, briefly described in Section 2, is very useful in solving the CSG prob-

[2] There also exists similar definitions for the strong positive and negative externalities and strong super- and sub-additivity, in which signs $\leq$ and $\geq$ are replaced with $<$ and $>$. In the remainder of this paper, whenever we refer to externalities and additivity, we mean their weak forms. Note that strong relationships are a subset of weak ones.

lem for PFGs. Figure 1 displays a modified version of the Sandholm *et al.* tree for six agents, where nodes (hereafter *configurations*) represent subspaces of CSs containing coalitions of particular sizes indicated by the number (cf. [11]). For instance, the configuration $\{5, 1\}$ denotes the subspace of all CSs containing exactly two coalitions of size 5 and 1 for 6 agents *i.e* $\{(12345), (6)\}$, $\{(12346), (5)\}$, $\{(12356), (4)\}$, $\{(12456), (3)\}$, $\{(13456), (2)\}$ and $\{(1), (23456)\}$. The arrows between the subspaces show how a merger of two coalitions converts one CS to the other. For example, the arrow from $\{4, 1, 1\} \rightarrow \{4, 2\}$ shows how the merge of the two coalitions of size 1 converts the configuration $\{4, 1, 1\}$ to $\{4, 2\}$.

The notion of weakness implies that there can be many CSs with the optimal value. Therefore, in actual fact, we should speak about a *set of optimal coalition structures* which, in a special case, might contain every feasible CS; this could occur, for example, when all weak externalities are zero and weak super- (sub-) additivity does not increase (decrease) the combined value of merging coalitions.

Theorem 1 *In PF^+_{sup} (PF^-_{sub}) the grand coalition (the coalition structure of singletons) always belongs to the set of optimal coalition structures. Furthermore, assuming that super- (sub-) additivity is not weak and both the positive and negative externalities are not weak then in PF^+_{sup} (PF^-_{sub}) the grand coalition (the coalition structure of singletons) is the only optimal structure.*

Proof: Consider PF^+_{sup} (PF^-_{sub}). Beginning with configuration $\{1, 1, 1 \ldots, 1\}$, it is possible to reach configuration $\{n\}$ by a variety of paths. Assume that we move from a coalition structure CS in configuration G of size k to a structure CS' in configuration G' of size $k - 1, \forall k = n, \ldots, 2$. In such a case, CS' must contain one coalition which is the union of exactly two coalitions in $CS \in G$ and $k - 2$ 'other' coalitions in CS which were not involved in the merge. Due to the super-additivity (sub-additivity) property, the value of the merged coalition in CS' must be greater than (less than) or equal to the sum of the component coalitions in CS. Furthermore, as a result of the positive (negative) externalities, the value of the other coalitions in CS' must not be smaller (bigger) than in CS. Consequently, the value of $CS' \in G'$ is not smaller (not bigger) than the value of $CS \in G$. Without loss of generality, this is applicable to every path, thus the configuration $\{n\}$ ($\{1, 1, 1 \ldots, 1\}$) must contain a structure whose value is not smaller than values of other CSs in every other configuration. Hence, the grand coalition (structure of singletons) always belongs to the set of optimal coalition structures in PF^+_{sup} (PF^-_{sub}).

Waiving the assumption of weakness (where the '$\leq$' and '$\geq$' signs are replaced with '$<$' and '$>$', respectively, in both super- and sub-additivity as well as positive and negative externality) then the above proof remains valid and it is not difficult to show that in the PF^+_{sup} (PF^-_{sub}) setting the grand coalition (structure of singletons) is the only optimal structure. ∎

It immediately follows that for both these PFGs, it is not necessary to search the entire CS space to find the optimal CS.

3 Bounded Coalition Values in PF^-_{sup} and PF^+_{sub}

In the PFG setting, each coalition (with the exception of the grand coalition and some coalitions in the second level of the Sandholm *et al.* tree) may have many values, depending on which CS it belongs to. This means that we cannot determine an exact value of a coalition in a particular structure without actually searching it. However, we will now show that, by searching only certain paths in the Sandholm *et al.* representation, it is possible to bound the value of every coalition in the entire tree. As the PF^-_{sup} problem is dual to the PF^+_{sub} problem, our result can be presented for both classes of games simultaneously.

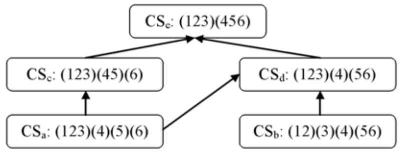

Figure 2: An extract from Sandholm *et al.* tree for 6 agents

Theorem 2 *Consider the PF^-_{sup} (PF^+_{sub}) setting and the coalition C_x in the structures $CS' = \{C_x, (i_1), \ldots, (i_{n-|C_x|})\}$ and $CS'' = \{C_x, C_y\}$ where $(i_1), \ldots, (i_{n-|C_x|}) \notin C_x$ and $C_y = A \setminus C_x$. The value of C_x in CS' is the greatest (smallest) value of C_x in every coalition structure it belongs to, or $\forall C_x \in CS$, $v(C_x; CS') \geq (\leq) v(C_x; CS)$. The value of C_x in CS'' is the smallest (greatest) value of C_x in every coalition structure it belongs to, or $\forall C_x \in CS$, $v(C_x; CS'') \leq (\geq)\ v(C_x; CS)$.*

Proof: First consider the value of C_x in CS' (*i.e.* $v(C_x; CS')$). In Figure 1, CS' can belong to any configurations in the following path: $\{1, 1, 1, 1, 1, 1\} \rightarrow \{2, 1, 1, 1, 1\} \rightarrow \{3, 1, 1, 1\} \rightarrow \{4, 1, 1\} \rightarrow \{5, 1\}$. Every coalition C_x such that $|C_x| \geq 1$ which appears in any configuration in this path is the only coalition that is formed. This guarantees that $v(C_x; CS')$ has never been affected by a negative (positive) externality. Conversely, in all the other configurations where C_x appears, other non-trivial coalitions co-exist whose creation, by definition, have induced negative (positive) externality on C_x. In such configurations the values of C_x will be at most (least) equal to $v(C_x; CS')$ since, as is visible in Figure 1, one can always reach any other configuration containing CSs with C_x starting from CS'.[3] Since, on such a path, C_x is only subject to negative (positive) externalities, $v(C_x; CS')$ must be at least as big (small) as in any other CS. Therefore, $v(C_x; CS')$ is the greatest (smallest) value of C_x in every CS that it belongs to.

Now consider the value of C_x in CS'' (*i.e.* $v(C_x; CS'')$). C_x is a part of both CS' and CS'', therefore, it is always possible to find a path which starts from CS' and leads to CS'', *i.e.* $CS' \rightarrow \ldots \rightarrow CS''$. Since C_x is only subject to consecutive negative (positive) externalities, the value of C_x will decrease (increase) or at most (least) remain the same, every time one traverses this path, moving from one configuration to another. Consequently, $v(C_x; CS'')$ will not be greater (smaller) than $v(C_x; CS')$ or the value of C_x in any other configuration on this path. Similarly, starting from any other configuration containing C_x, it is always possible to find a path leading to CS''. Since C_x is subject to consecutive negative (positive) externalities through such paths, the above argument is equally compelling. Therefore, the value of $C_x \in CS''$ is the smallest (greatest) value of C_x in every coalition structure it belongs to. ∎

Consider a few elements of the original Sandholm *et al.* tree in Figure 2. Theorem 2 says that under PF^-_{sup} $\forall (123) \in CS$, $v((123); CS_a) \geq v((123); CS)$ (where CS is any structure containing (123)). Initially, it may seem possible for $v((123); CS_d)$ to be higher than $v((123); CS_a)$ because the former structure emerged

[3] With the exception of CS', $CS''$and the grand coalition, any coalition C_x might have a number of different values in one configuration, as it belongs to a number of distinct CSs. Thus, we use the plural for "values" and "coalition structures".

after agent 3 joined coalition (12) in CS_b and, due to super-additivity property, $v((123); CS_d)$ could become much higher than $v((123); CS_a)$. However, in actual fact, this cannot happen because of the assumed negative externalities. It is always possible to find a path from $\{(123), (4), (5), (6)\}$ to any other CS that contains (123) and on such a path the value of (123) is only subject to negative externalities. Consequently, $v((123); CS_d)$ cannot be higher than $v((123); CS_a)$. Such reasoning can also show that $v((123); CS_e)$ is the smallest value of (123) in Figure 2 and similar reasoning can be used to back up our claims for the PF^+_{sub} setting.

4 CSG Algorithm For The PFG Setting

The Rahwan *et al.* CSG algorithm relies on the fact that coalition values are always constant in the CFG setting. This makes it possible to collect a number of basic statistics at the very beginning to assess which configurations are most promising and which not. In the PFG setting, coalition values depend on the CS they belong to, so such a technique is not generally feasible. However, for both PF^-_{sup} and PF^+_{sub}, Theorem 2 allows us to construct bounds on the values of every coalition in every CS. Subsequently, we can use these bounds to construct upper and lower bounds for each configuration. In other words, our theorem bridges the gap between both settings, making it possible to modify the existing state-of-the-art CSG algorithm so that it can generate a set of optimal CSs in the PF^-_{sup} (PF^+_{sub}) setting, often without searching the entire CS space. Let L_s denote the (structured) list containing all coalitions of size s. [4] Our CSG algorithm can be summarized as follows:

Step 1. Compute the value of the grand coalition. For every coalition C in list $L_s : 1 \leq s = |C| < n$, compute its value in the CSs where: (i) all the other agents not in C form coalition $C' = A \setminus C$, and (ii) every other agent not in C acts alone. These are the maximum and minimum (minimum and maximum) values of each coalition in the entire CS space and are stored in lists L^{max}_s and L^{min}_s which are structured as in the DCVC algorithm (see [9]);

Step 2. Partition the search space into configurations. Prune those which were searched in Step 1;

Step 3. Compute the upper bounds of every remaining configuration G, denoted UB_G, using the lists of maximum values from Step 1, *i.e.* $UB_G = \sum_{\forall s \in G} maxL^{max}_s$. Set the upper bound of the entire system UB to be the value of the highest upper bound, *i.e.* $UB = maxUB_G$ and set the lower bound to be $max\{v(CS^*_N), max\{Avg_G\}\}$, where $Avg_G = \sum_{\forall s \in G} avgL^{min}_s$ is the lower bound for the average value of each configuration G and CS^*_N is the CS with the highest value found thus far. Order the configurations w.r.t. the value of UB_G;

Step 4. Prune away those subspaces which cannot deliver a CS greater than LB, *i.e.* $UB_G < LB$;

Step 5. Search the configuration with the highest upper bound, updating LB to be the highest value of the structure found thus far (CS^*_N). During the search process, a refined branch and bound technique should be used;

Step 6. Once the search of the configuration in Step 5 is completed, check whether $v(CS^*_N) = UB$ or all configurations have been searched or pruned. If any of these conditions hold then the optimal CS has been found. Otherwise, go to Step 4.

In Step 1 we compute the maximum (minimum) and the minimum (maximum) values of each coalition C in the entire tree. Storing both numbers *per* coalition requires twice as much memory as in the CFG setting, but ensures that the highest and lowest values of each list L_s can be computed. This makes it possible to determine upper and lower bounds for each configuration as well as the upper bound of the entire system. Furthermore, in contrast to Rahwan *et al.*, we cannot compute an exact average value of all the coalitions of size $s_i, \forall i = 1, \ldots, m$, for a given configuration $G = \{g_{s_1}, \ldots, g_{s_m}\}$. However, it is possible to compute a lower bound for such an average value using L^{min}_s as no average value can be smaller than the one computed for the lists containing minimum values. In addition, the upper bound for each configuration G can be defined as the sum of maximal values that every coalition of size s in $CS \in G$ can take, *i.e.* $\sum_{\forall s \in G} maxL^{max}_s$.

In the PFG setting, partitioning and pruning of the search space is done as in the Rahwan *et al.* algorithm for the CFG setting. Also, the process of searching through the promising subspaces is similar. In particular, certain techniques ensure that no redundant calculations are performed, *i.e.* no CS is considered twice. However, the branch-and-bound rule needs to be modified for the PFG setting. This rule prevents traversing hopeless paths while constructing CSs in the considered configuration.

Branch and Bound Rule Suppose that $G^* = \{g_{s_1}, g_{s_2}, g_{s_3}, g_{s_4}\}$ is the configuration with the highest upper bound, which has not yet been searched. In the CFG setting, the branch and bound rule of Rahwan *et al.* goes as follows. Suppose the algorithm has already added coalitions $C_{g_{s_1}}$, $C_{g_{s_2}}$ to the CS under construction. When adding the next coalition from list $L_{g_{s_3}}$, the rule ignores cases which together with $\max L_{g_{s_4}}$, would render the values of the CS less than the current LB of the entire system. From Theorem 2, instead of exact values of coalitions which we do not know beforehand, we can use the maximum values as computed in Step 1 and incorporate this rule to both PF^-_{sup} and PF^+_{sub} settings. However, with only maximum values, such a branch and bound rule is likely to be less effective than in the original setting.

Anytime properties When the arguments of Sandholm *et al.* [12] are applied to the upper bounds of values of coalitions, it can be proven that after Step 1 of our algorithm, where levels 1, 2 and n have already been searched, the value of CS^*_N is no smaller than $\lfloor \frac{n}{2} \rfloor$ of the optimal CS, *i.e.* $\lfloor \frac{n}{2} \rfloor \times v(CS^*_N) \geq CS^*$. Furthermore, updating the lower bound in Step 5 ensures that if we were to continuously stop and restart the algorithm, every time we stopped, we would always have a current optimal structure CS^*_N with a value at least as big as the value obtained before we re-started. Therefore, the algorithm retains its anytime properties in the PFG setting.

5 Numerical Simulations

To the best of our knowledge, the CSG algorithm for the PFG setting proposed in this paper is the only one in the literature; thus there is no benchmark algorithm that can be used for a numerical comparison. Although it would be possible to adapt the CFG dynamic programming techniques for the PFG setting, due to lack of space, we will compare our results to the CSG algorithm for the CFG setting instead. As noted at the beginning of the paper, this solution has already been proven to be significantly superior w.r.t. dynamic programming alternatives, because it does not need to search all the feasible CSs. We will show that, in many cases, our modification of this algorithm for the PFG setting also only searches through a fraction of the CS space, thus saving a vast amount of calculation time.

Simulations are performed for the PF^-_{sup} setting. When the new

[4] see [9] for more details

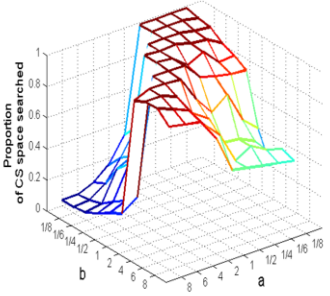

Figure 3: Simulation results for PF^{-}_{sup} setting

coalition is formed, the 'gain' from super-additivity is accounted for by adding a factor $\frac{\alpha}{a}$ to its value. In addition, the 'loss' from the externality on the other coalitions in the structure is accounted for by multiplying their values by factors $\frac{b-\beta}{b}$, where $\alpha, \beta \in [0, 1)$ are randomly-generated uniform variables and $a, b \geq 1$ are constants. We assume that in the system there are 10 agents, from which 115,975 CSs can be formed.[5] In Step 1 2028 CSs are searched, *i.e.* the grand coalition, the CSs of singletons and $2C^{10}_2 + 2C^{10}_3 + ... + 2C^{10}_8 + C^{10}_9$ other CSs. This amount accounts for 1.75% of the search space.

The vertical axis on Figure 3 represents the proportion of the CS space searched, whereas a and b are indicated on the x and y axes, respectively. As the values of a and b increase, the 'gain' from super-additivity and the 'loss' from externalities decrease. We performed our simulations 25 times for each combination of a and b. The surface shown in Figure 3 is the average proportion of space searched by our algorithm. Furthermore, as the original CSG algorithm for the CFG setting for the uniform distribution of coalition payoffs searches on average about 2.5%, and this result is independent from a and b, we do not report it in Figure 3.

We observe that when the 'gain' from super-additivity is high and the 'loss' from the negative externality is low, only a minimal proportion (under 4 $\%$) of the space need be searched in order to compute the optimal structure. In fact, in such cases, the grand coalition or a coalition in the first few levels of the Sandholm *et al.* tree is usually the optimal structure. Consequently, it would seem that the smaller the externality, the more the PF^{-}_{sup} setting becomes like the super-additive CFG setting, thus explaining why so little of the space is searched. Conversely, when the 'gain' from super-additivity is low and the 'loss' from the negative externality is high, only a fraction of the search space was searched. This time, the PF^{-}_{sup} setting becomes more akin to the sub-additive CFG setting, so that the CS of singletons or a CS with a relatively small number of cooperating agents tends to be optimal. However, in situations where the 'loss' from the externality and the 'gain' from the super-additivity are both either high or low, it seems that pruning is ineffective since nearly all of the search space has to be searched in order to guarantee an optimal outcome (more than 98% in many cases). This is due to an inherent characteristic of the PF^{-}_{sup} setting: namely, that the values of the structures in each configuration are dependent on the value of the structures in the configuration in the previous level (see Figure 1). Consequently, when the gain from the super-additivity and the loss from externalities are of a similar magnitude, the extreme values of CSs in different configurations are more likely to be akin, making pruning techniques less effective. This effect is magnified by the use of the uniform distribution since CSs' values in all configurations tend to be relatively dispersed.

[5] The particular challenge of simulations in the PFG setting is that (in contrast to the CFG setting) one must generate the values of all CSs beforehand. Furthermore, during the random generation of coalition values, it is important to ensure that all the CSs meet PF^{-}_{sup} (PF^{+}_{sub}) properties. Consequently, we restrict our simulations to 10 agents and 115,975 CSs. Although this is less than the system of 27 agents considered for the CFG setting (*cf.* [11]), such a system in the PFG setting would require generating a CS space with more than 5.24×10^{20} CSs.

6 Conclusion $\&$ Future Work

In this paper, we considered coalition structure formation in the presence of coalition externalities, a novel topic in the multi-agent system literature. We modeled coalition formation with a partition function game (PFG), and considered four cases: (1) super-additive games with positive externalities (PF^{+}_{sup}), or (2) negative externalities (PF^{-}_{sup}); (3) sub-additive games with positive externalities (PF^{+}_{sub}); or (4) negative externalities (PF^{-}_{sub}). For cases (1) and (4), we proved that computing the optimal structure is straightforward, because either the grand coalition or the CS of singletons belong to the set of optimal CS. In contrast, this is not true for cases (2) and (3), where any CS can belong to the set of optimal coalition structures. Therefore, for these two cases we proved that it is possible to bound the value of each coalition. From this insight, we modified the existing state-of-the-art anytime CSG algorithm for the CFG setting and show how it can be used to generate the optimal CS in these two PFG settings. In future work, we plan to study the numerical performance of the new algorithm under different distributional assumptions regarding coalition values, and also develop a distributed version of our approach.

REFERENCES

[1] F. Bloch, 'Non-cooperative models of coalition formationin games with spillovers', *in Carraro, C. (ed.), Endogenous Formation of Economic Coalitions*, ch. 2 pp. 35–79, (2003).

[2] E. Catilina and R.Feinberg, 'Market power and incentives to form research consortia', *Review of Industrial Organization*, **28**(2), 129–144, (2006).

[3] V. Dang and N. Jennings, 'Generating coalition structures with finite bound from the optimal guarantees', in *AAMAS*, New York, USA, (2004).

[4] K. Do and H. Norde, 'The shapley value for partition function form games', *Discussion Paper, Tilburg University, Center for Economic Research*, (2002).

[5] M. Finus and B. Rundshagen, 'Endogenous coalition formation in global pollution control. a partition function approach.', *Working Paper No. 307, University of Hagen.*, (2001).

[6] I. Hafalir, 'Efficiency in coalition games with externalities', *Games and Economic Behaviour*, **61**(2), 209–238, (2007).

[7] L. Kóczy, 'A recursive core for partition function form games', *Theory and Decision*, **63**(1), 41–51, (2007).

[8] P. Pintasssilgo, 'A coalition approach to the management of high seas fisheries in the presence of externalities', *Natural Resource Modeling*, **2**(16), 175–197, (2003).

[9] T. Rahwan and N. Jennings, 'Distributing coalitional value calculations among cooperating agents', in *AAAI*, pp. 152–157, Pittsburgh, USA, (2005).

[10] T. Rahwan, S. Ramchurn, V. Dang, A. Giovannucci, and N. Jennings, 'Anytime optimal coalition structure generation', in *Proceedings of AAAI 2007*, (2007).

[11] T. Rahwan, S. Ramchurn, V. Dang, A. Giovannucci, and N. Jennings, 'Near-optimal anytime coalition structure generation', in *IJCAI*, Hyderabad, India, (2007).

[12] T. Sandholm, K. Larson, M. Andersson, O. Shehory, and F. Tohme, 'Coalition structure generation with worst case guarantees', *AI*, **1-2**(111), 209–238, (1999).