DELIS-TR-398

Behavioral Coalition Structure Generation

Rossi, G.

2006
Behavioral coalition structure generation

Giovanni Rossi

November 2006

Department of Computer Science
University of Bologna
Mura Anteo Zamboni 7
40127 Bologna (Italy)
Recent Titles from the UBLCS Technical Report Series

2006-10 Fast and Fair Event Delivery in Large Scale Online Games over Heterogeneous Networks (PhD Thesis), Palazzi, C.E., April 2006.
2006-14 Performative Patterns for Designing Verifiable ACLs, Dragoni, N., Gaspari, M., April 2006.
2006-17 A Secure Peer Sampling Service as a Hub attack countermeasure, Jesi, G. P., Gavidia, D., Gamage, C., van Steen, M.
2006-18 Modified Realizability and Inductive Types, Asperti, A.; Tassi, E., June 2006
2006-23 Emergent Social Rationality in a Peer-to-Peer System Marcozzi, A.; Hales, D., October 2006
2006-24 Reconstruction of the Protein Structures from Contact Maps Margara, L.; Vassura, M.; di Lena, P.; Medri, F.; Fariselli, P.; Casadio, R., October 2006
2006-25 Lambda Types on the Lambda Calculus with Abbreviations Guidi, F., November 2006
Behavioral coalition structure generation

Giovanni Rossi

November 2006

Abstract
Coalition structures are partitions of agents, i.e., collections of pair-wise disjoint coalitions, called blocks, whose union yields the entire population. Given a coalitional game, assigning a worth to each coalition, the worth of coalition structures obtains as the sum of blocks’ worth. Optimal coalition structures have maximal worth. For generic coalitional game, searching optimal coalition structures is NP-hard. Here, a behavioral coalition structure generation algorithm is presented. It specifies a selfish behavior leading agents to move across blocks depending on how coalitions allocate their worth across members. This latter issue is addressed through cooperative game theory. Given that optimality is sought through agents’ selfish behavior (rather than searched directly), the system is adaptive: if the underlying coalitional game varies over time, then whenever a change occurs the algorithm redirects agents toward new optimal coalition structures.

1. This work is partially supported by the EU within the 6th Framework Programme under contract 001907 (DELIS).
2. Department of Computer Science, University of Bologna, Mura Anteo Zamboni 7, 40126 Bologna, Italy.
1 Introduction

Coalition formation is a key topic in multiagent systems, as there exist several activities that coalitions carry out better than single agents. In particular, in e-commerce coalitions often manage to buy (or sell) at lower (higher) unit prices than individuals alone (see [15], [17], [18], [33], [16]). More generally, the whole system may periodically receive tasks that only certain \(k\)-cardinal coalitions, \(k > 2\), can perform (see [28], [25]). It is common practice to treat this topic in terms of cooperative games: some coalitional game, quantifying the worth of coalitions, is given; it has to be neither superadditive nor subadditive, meaning that the sum of the worth of any two disjoint coalitions may be greater, equal or smaller than the worth of their union. That is, merging is profitable in some cases and unprofitable in some other cases (see [29]).

A coalition structure (or partition of agents) is a non-void collection of non-void and pairwise disjoint coalitions, or blocks, whose union yields the entire population. In particular, from a global perspective, an optimal coalition structure is one where the sum of blocks’ worth is maximized. If this worth is quantified by a coalitional game which is neither subadditive nor superadditive, than optimal coalition structures consist, in general, of a number of blocks strictly greater than 1 and strictly smaller than the number of agents. The issue next becomes how to push the system toward optimality.

In game theory, the main concern is with players’ strategic behavior in coalition formation games (see [30]). These latter are multistage (non-cooperative) games where each player or agent, at each stage, chooses what coalition she would like to join in the next stage. Then, a mechanism maps collective choices onto coalition structures. Hence, players’ behavior identifies a time sequence of coalition structures. In turn, players’ choices, at each stage, crucially depend on what they get as members of different coalitions. This introduces the general issue concerning the solution of coalitional games (see [27]), which is central in cooperative game theory. Roughly speaking, a solution is a criterion for sharing the fruits of cooperation between cooperating players, i.e., a rule for distributing coalitions’ worth across members.

In artificial intelligence the main concern is with computation. In fact, if the worth of coalitions is known, then finding an optimal coalition structure is equivalent to determining the winners in combinatorial auctions (see [26]), a well known NP-hard problem. The focus is on search algorithms for finding coalition structures that are optimal or close enough to optimality. Behaviorally, the idea is that agents aim at settling into ‘good’ partitions, i.e., partitions that are optimal or almost optimal. Hence, as soon as any such a partition is found, agents are assumed to (cooperatively) group accordingly. In this framework, the task to be performed by coalitions may be precisely a search, within some assigned portion of admissible coalition structures, toward optimality.

These two alternative approaches, although somehow opposite to each other (as each one disregards the other’s main concern), aim both at a common target: if coalition member payoffs are properly defined, then coalition formation games are potential games (see [30], [21]), and equilibrium (or potential maximizer) strategies yield optimal coalition structures. The model presented here does not formalize a strategic coalition formation game. In particular, when agents choose a coalition they want to belong to, whether such a coalition should accept them, as new members, or not, from a purely strategic viewpoint, is a not an issue addressed here. New members are always (automatically) accepted by coalitions, although these latter subsequently decide how to reward the former. In fact, the strategic behavior of coalitions consists precisely in properly rewarding members. Under this respect, the approach developed here is typically a cooperative game-theoretical one, as the solution problem associated with coalitional games plays a crucial role.

The number \(B_n\) of partitions of a \(n\)-set, \(n \geq 0\), is the \(n\)-th Bell number, determined through recursion by \(B_n = \sum_{0 \leq m \leq n-1} \binom{n-1}{m}B_m\), with \(B_0 = 1\) (see [1], [10] and [23]). In large popula-
tions the number of distinct coalition structures gets so large that no search of optimal coalition structures seems conceivable (see [25], p. 217). Conversely, coalition structure generation (toward optimality) may be left to agents’ self-interested behavior, with limited information and computationally bounded. In this way, behavioral coalition structure generation becomes adaptive: if the underlying coalitional game (i.e., the worth of coalitions and coalition structures) changes over time, then the generation process pushes the system toward new optimal coalition structures. In the model proposed below, agents basically make periodic comparisons with (randomly chosen) other agents, moving into the other agent’s block whenever this latter receives a strictly higher payoff. Computing such a payoff in the best way is precisely the role of coalitions. In addition, given that the underlying coalitional game might happen to give worth \( > 0 \) to 1-cardinal coalitions and worth 0 to all remaining coalitions (see [25], Theorem 2, p. 216), mutations (i.e., individual agents leaving their current coalition and forming their own, 1-cardinal block) must also occur with small but strictly positive probability.

The paper is organized as follows. The next section contains the needed preliminaries, mainly from cooperative game theory and with special emphasis on the solution problem associated with coalitional games. Section 3 introduces type-symmetric coalitional games (where the whole population is partitioned into types and the worth of coalitions depends only on the number of members of each type), and "dynamic coalitions" (each of whose members is characterized by her own membership age). Sections 4 and 5 define, respectively, coalitional behavior (i.e., how coalitions reward their members depending on their type and membership age) and agent behavior (i.e., when and why agents move from their current coalition or block to another one). Section 6 contains some concluding remarks, mainly focused on possible simulation setups. Finally, section 7 is an appendix where the potential approach is shown to provide some game-theoretical results linking efficient worth-sharing within coalitions and optimal (and stable) coalition structures.

2 Preliminaries

Given a finite population \( N = \{1, \ldots, n\} \), subsets \( A \subseteq N \) are coalitions, and a coalition structure, or partition \( P = \{A_1, \ldots, A_{|P|}\} \), is a non-void collection of non-void and pair-wise disjoint coalitions \( A_1, \ldots, A_{|P|} \subseteq N \) - the blocks of \( P \) - whose union yields \( N \). Power set \( 2^N = \{A : A \subseteq N\} \) is the set of all coalitions, and set functions \( v : 2^N \to R \) are coalitional games, assigning a worth to each coalition. Also, if \( \mathcal{P}^N \) denotes the set of all coalition structures, then partition functions \( f : \mathcal{P}^N \to R \) are global games, assigning a worth to each coalition structure (see [6]). Given coalitional game \( v \), consider global game \( f^v \) where the worth of partitions obtains as the sum of blocks’ worth as defined by \( v \), i.e.,

\[
f^v(P) = \sum_{A \in P} v(A) \text{ for every } P \in \mathcal{P}^N.
\]

Such a game \( f^v \) is additively separable (see [6], [7], [22]); in particular, it is additively separated by \( v \). Optimal coalition structures \( P^* \) satisfy

\[
f^v(P^*) \geq f^v(P) \text{ for every } P \in \mathcal{P}^N.
\]

A coalitional game is superadditive if for every \( A, B \in 2^N \) such that \( A \cap B = \emptyset \)

\[
v(A \cup B) \geq v(A) + v(B),
\]

while if \( \geq \) is replaced with \( \leq \), then \( v \) is subadditive. The concern here is with optimal coalition structures when the underlying coalitional game \( v \) is neither superadditive nor subadditive (see

UBLCS-2006-27 3
[29]. In fact, if \( v \) is superadditive, then
\[
v(N) \geq \sum_{A \in P} v(A) \text{ for every } P \in \mathcal{P}^N.
\]
Similarly, if \( v \) is subadditive, then
\[
\sum_{i \in N} v(i) \geq \sum_{A \in P} v(A) \text{ for every } P \in \mathcal{P}^N.
\]
That is, if \( v \) is superadditive, then the coarsest coalition structure is optimal, while if \( v \) is subadditive, then the finest coalition structure is optimal.

Given some coalitional game as input, searching optimal coalition structures is NP-hard. In fact, this is equivalent to determining the winners in combinatorial auctions, in which case \( N \) is a set of commodities to be sold, while coalitional game \( v \) gets defined through buyers’ bids for combinations of such commodities. The ultimate branch-and-bound search algorithm for this problem, called BOB (branching-on-bids), performs very well (see [26]). Nevertheless, if the underlying coalitional game is unknown and may vary over time, then BOB cannot be used in its current form, as it becomes necessary to constantly “explore” the worth of coalitions, precisely because such a worth may change anytime. In other words, BOB needs the known worth of coalitions as input, but this worth might well vary while BOB performs the search.

Through a totally different route, the coalition structure prevailing at each time can be generated by agents’ individual behavior. Formally, a behavioral coalition structure generation algorithm defines some (randomized) agent behavior yielding a time sequence \( P^0, P^1, \ldots, P^t, \ldots \in \mathcal{P}^N \) of coalition structures. In particular, each agent may periodically compare her payoff with that received by some randomly chosen other agent; if the former exceeds or equals the latter, then she remains a member of her current coalition; otherwise, she moves into the other agent’s coalition. Although rather simply, this clearly generates a sequence of coalition structures, whose blocks are here termed dynamic coalitions in order to emphasize that some members are older than others.

This behavioral coalition structure generation, simple though it is, raises a main question: how to define coalition members’ payoffs? That is, suppose some coalition \( A = A^t \in \mathcal{P}^t \) prevails at time \( t \). Then, how should coalitional worth \( v(A) \) be shared between members \( i \in A \)? This leads to observe that in a behavioral coalition structure generation algorithm not only individual behavior, but also coalitional behavior (i.e., how to divide the worth of cooperation) has to be defined. Here, the worth of coalitions is shared between members according to some dynamic version of weighted Shapley values (see [27], [13] and [20]), i.e., applying to dynamic coalitions.

The Shapley value \( \phi^* \) of coalitional games \( v \) solves the problem of dividing a total worth \( v(N) \) between all members \( i \in N \). In particular, it maps \( v \) onto an \( n \)-dimensional real-valued vector \( ^*\phi(v) = (^*\phi_1(v), \ldots, ^*\phi_n(v)) \) defined by
\[
^*\phi_i(v) = \sum_{A \subseteq N \setminus i} \frac{v(A \cup i) - v(A)}{n(n-1)} \text{ for every } i \in N.
\]
That is, each player \( i \in N \) receives a share which is a weighted average of her marginal contributions \( v(A \cup i) - v(A) \), \( A \subseteq N \setminus i \). Among other axioms, \( ^*\phi \) satisfies symmetry, i.e., for every pair \( i, j \in N \), if \( v(A \cup i) - v(A) = v(A \cup j) - v(A) \) for every \( A \subseteq N \setminus \{i, j\} \), then \( ^*\phi_i(v) = ^*\phi_j(v) \).

The Möbius inversion \( \mu^v : 2^N \to \mathbb{R} \) of coalitional games \( v \) is
\[
\mu^v(A) = \sum_{B \subseteq A} (-1)^{|A \setminus B|} v(B) \text{ for every } A \in 2^N.
\]
Note that
\[ v(A) = \sum_{B \subseteq A} \mu^v(B) \text{ for every } A \in 2^N. \] (2)

This expression is a combinatorial analog of the fundamental theorem of calculus (see [24]). That is, \( \mu^v \) is the analog of the derivative of \( v \), as the latter quantifies the worth of coalitions, while the former quantifies the net worth added by coalitions with respect to all proper subcoalitions.

It must be stressed that even if the worth of any coalition is positive (as commonly assumed), still the Möbius inversion takes, in general, both positive and negative values. In terms of \( \mu^v \), the Shapley value takes form
\[ \phi_i^*(v) = \sum_{A \subseteq N \setminus i} \frac{\mu^v(A \cup i)}{|A| + 1} \text{ for every } i \in N. \] (3)

This allows to note that \( \phi^* \) also satisfies efficiency, i.e., the sum of all \( n \) shares yields the entire worth to be shared, as
\[ \sum_{i \in N} \phi_i^*(v) = \sum_{A \in 2^N} |A| \cdot \frac{\mu^v(A)}{|A|} = \sum_{A \in 2^N} \mu^v(A) = v(N) \]
by (2). For a complete axiomatic characterization of \( \phi^* \) see [27] and [31].

### 3 Type-symmetry and membership age

As already mentioned, this paper provides a full description of a novel coalition structure generation process based on agents’ selfish (and computationally bounded) behavior. In particular, this process is designed to face a very complex setting, as any (generic) instance results in a NP-hard problem. In addition, different instances (i.e., different underlying coalitional games) follow one after the other in time. In practice, at any time the only available information is the worth of those coalitions prevailing at that time, as well as the worth of all proper subcoalitions of these latter. Accordingly, the performance of any algorithm designed to face such a situation may well be tested through simulations (rather than approximated analytically). In fact, the whole model described below is conceived to be tested through simulations with large populations, which shall be object of future work. In view of this target, the algorithm is described for the case where agents are divided into types and their (marginal) contributions to coalitions depend only on their type. This dramatically reduces the dimension of coalitional games regarded as vectors, as explained hereafter.

A coalitional game \( v : 2^N \to \mathbb{R} \) is symmetric if the worth of coalitions depends only on their cardinality. That is, there is a \( \gamma : \{0, 1, \ldots, n\} \to \mathbb{R} \) such that \( v(A) = \gamma(a) \) for every \( A \in 2^N \), where \( |A| = a \). Geometrically, \( v \) is a \( 2^n \)-dimensional real-valued vector, as there are \( 2^n \) distinct coalitions (one of which is the empty set, whose worth is always 0). Clearly, if \( v = \gamma \) (i.e., if \( v \) is symmetric), then only \( n+1 \) real numbers \( \gamma(a), 0 \leq a \leq n \) have to be specified. For any coalition \( A \), let \( b \) denote the cardinality of any subset \( B \subseteq A \). The Möbius inversion of symmetric games is
\[ \mu^v(A) = \sum_{B \subseteq A} (-1)^{|A \setminus B|} v(B) = \sum_{0 \leq b \leq a} \binom{a}{b} (-1)^{a-b} \gamma(b). \]

**Example 1** Symmetric game \( v = \gamma \) defined by
\[ v(A) = \gamma(a) = \frac{a}{1 + (a - m^*)^2} \]
3 Type-symmetry and membership age

for some integer $m^* \leq n$ allows to appreciate the general idea behind the whole model presented below in a rather simple environment. Let $n = m_* \cdot m^*$, with $m^* > m_* \geq 2$ and where $m_*$ denotes $n$’s smallest (but $> 1$) divisor. Now, if $m^*$ has several divisors, then the underlying symmetric coalitional game can be made periodically change by choosing at random any of these divisors and substituting it into the above expression in place of $m^*$. For example, if $n = 400$, then $m_* = 2$ and $m^* = 200$, so that from time to time the underlying symmetric game can be changed by randomly drawing some divisor of 200 (say 2, 4, 5, 10, 20, 50, 100), and substitute it into the above expression as a new value of $m^*$. Note that

$$
\max_{p \in P^N} \sum_{A \in p} \gamma(|A|) \frac{n}{m^*} \frac{m^*}{1 + (m^* - m^*)^2} = n,
$$

i.e., for any drawing the worth of optimal coalition structures is the same.

Type-symmetry is a sophistication of symmetry. Formally, assume $N$ is partitioned into $K$ types, with $N_k$ denoting the type-$k$ population, $1 \leq k \leq K$. In particular, set $n_k = |N_k|, 1 \leq k \leq K$. Then, coalitional game $v$ is type-symmetric if the worth $v(A)$ of any coalition $A \in 2^N$ depends only on the number $|A \cap N_k|$ of members of each type $k = 1, \ldots, K$ (see [3]). For every $A, B \in 2^N$, let

$$
\bar{a} = (\bar{a}_1, \ldots, \bar{a}_K) = (|A \cap N_1|, \ldots, |A \cap N_K|),
$$

$$
\bar{b} = (\bar{b}_1, \ldots, \bar{b}_K) = (|B \cap N_1|, \ldots, |B \cap N_K|),
$$

$$
a = |A| = \sum_{1 \leq k \leq K} \bar{a}_k \text{ as well as } b = |B| = \sum_{1 \leq k \leq K} \bar{b}_k.
$$

Integer-valued vectors $\bar{a}, \bar{b}$ define the type of coalitions $A, B$. If $v$ is type-symmetric, then there is a $\gamma : Q_+^K \to \mathcal{R}$, where $Q_+ = \{0, 1, 2, \ldots\}$, such that $v(A) = \gamma(\bar{a})$ for every $A \in 2^N$. For $x, y \in Q_+^K$, let $x \leq y$ mean that $x_k \leq y_k, 1 \leq k \leq K$. Also let $0 = (0, \ldots, 0)$. For any coalition $A$ of type $\bar{a}$, the number of distinct types of coalitions $B \subseteq A$ is $\prod_{1 \leq k \leq K} \bar{a}_k$. This is the number of distinct $\bar{b}$ such that $0 \leq \bar{b} \leq \bar{a}$. In particular, the number of distinct types of coalitions in the entire population is $\prod_{1 \leq k \leq K} n_k$. Concerning M"obius inversion,

$$
\mu^v(A) \sum_{B \subseteq A} (-1)^{|A \setminus B|} v(B) \prod_{0 \leq b \leq \bar{a}} \prod_{1 \leq k \leq K} \left(\frac{\bar{a}_k}{\bar{b}_k}\right) (-1)^{\bar{a} - \bar{b}} \gamma(\bar{b}). \tag{4}
$$

In fact, for any type $\bar{b}$ of a subset of $A$, i.e., $0 \leq \bar{b} \leq \bar{a}$, there are precisely $\prod_{1 \leq k \leq K} (\bar{a}_k)$ distinct subsets of $A$ of type $\bar{b}$.

Define some integer $H > 1$ to be the memory of the model. More precisely, the algorithm shall define agents’ behavior as random movements from a block to another (of prevailing coalition structures), biased toward optimal coalition structures. Accordingly, for any block it is possible to distinguish agents in terms of their membership age. In fact, in terms of their members’ age, coalitions have a further detailed identification, apart their type. Formally, dynamic coalitions are coalitions $A^t = A \in 2^N$ that prevail at some time or (simulation) stage $t$. Their members’ age is formalized through a vector $\underline{a} = (\underline{a}_1, \ldots, \underline{a}_H) \in Q_+^H$ such that $\underline{a}_h$ is the number of $A$-members with age $h$. All members whose age is $H$ or more are treated the same. In this respect, $H$ is precisely the memory of the generation process. Vector $\underline{a}$ shall be the age of coalition $A$. If the underlying game is type-symmetric, then dynamic coalitions are also characterized by a type, that is to say, each coalition $A$ shall be characterized by a matrix $\bar{a} \in Q_+^{K \times H}$ such that $\underline{a}_{kh}$ is the number of agents in $A = A^t$ (i.e., at stage $t$) whose type is $k$ and whose membership age is $h$. 
4 Coalitional behavior

As already mentioned, the algorithm presented in this paper ideally consists of two parts. At a lower level, it specifies when and why agents leave their current coalition and move into another (possibly 1-cardinal) one. At an higher level, it specifies how coalitions reward their members, i.e., coalitional behavior. In fact, agents shall move whenever they perceive not to be rewarded enough. This section is clearly devoted to this latter part.

For any coalitional game \( v \) and coalition \( A \), define \( v^A : 2^N \rightarrow R \) by

\[ v^A(B) = v(A \cap B) \text{ for every } B \in 2^N. \]

In cooperative game theory, \( v^A \) is referred to as the restriction of \( v \) to \( A \). Let \( P = P^t = \{ A^t_1, \ldots, A^t_K \} \) denote the generic coalition structure prevailing at some stage \( t \). As a preliminary step, consider the case where the worth of each block \( A \) of the prevailing partition is shared between members according to the Shapley value of \( v^A \). That is, each agent \( i \in A \) is rewarded according to

\[ ^*\phi_i(v^A) = \sum_{B \in 2^A \setminus 2^A_i} \frac{\mu^v(B)}{|B|}, \]

where \( 2^A \setminus 2^A_i = \{ B \subseteq A : i \notin B \} \). In fact, this is the Aumann-Dreze value for games with coalition structure (see [2]), which uses expression (3) for restricted games.

In order to adapt this sharing rule to the case under concern, with type-symmetric coalitional games \( v = \gamma : \{ \bar{a} : 0 \leq \bar{a} \leq \{n_1, \ldots, n_K \} \} \rightarrow R \), and dynamic coalitions \( A \) with membership age, some additional notation is needed. Firstly, let \( h^t : N \rightarrow \{1, \ldots, H\} \) and \( k : N \rightarrow \{1, \ldots, K\} \) define agents’ age and type, respectively. That is to say, \( h^t(i) \) is the membership age of agent \( i \) in dynamic coalition \( A = A^t \) prevailing at generic stage \( t \), while \( k(i) \) is the type of \( i \in A \), that never changes. Also let \( H^A = \{ h \in H : \bar{o}_h > 0 \} \) denote the set of ages \( h \) such that \( A \) has at least one member with age \( h \). Similarly, let \( K^A = \{ k \in K : \bar{o}_k > 0 \} \) denote the set of types \( k \) such that \( A \) has at least one member with type \( k \). Secondly, given any type \( \bar{a} \) of coalitions \( A \), for every type \( \bar{b} \) of subcoalitions \( B \subseteq A \), the value \( \mu^\gamma(\bar{b}) \) attained by Möbius inversion on this latter type is given by expression (4) above, i.e., recursively,

\[ \mu^\gamma(\bar{b}) = \gamma(\bar{b}) - \sum_{\bar{d} < \bar{b}} \prod_{1 \leq k \leq K} \frac{\bar{b}_k}{\bar{d}_k} \mu^\gamma(\bar{d}), \]

where \( \bar{d} < \bar{b} \iff \bar{d} \leq \bar{b}, \bar{d}_k < \bar{b}_k \) for at least one type \( k \). Furthermore, there are precisely \( \prod_{1 \leq k \leq K} (\bar{a}_k/\bar{b}_k) \) distinct subsets of \( A \) of this type \( \bar{b} \) (see above). For \( 0 < \bar{b} \leq \bar{a} \), the whole of these values has to be shared in a way such that any two members of same type and age are equally rewarded. In particular, for any type \( \bar{b} \leq \bar{a} \), let

\[ H^\bar{b} = \{ h \in H^A : \bar{o}_h > 0 \text{ for some } B \subseteq A \text{ of type } \bar{b} \}. \]

In words, \( H^\bar{b} \) is the set of ages \( h \) such that there is at least one \( B \subseteq A \) of type \( \bar{b} \) with at least one member with age \( h \). Then, let every agent \( i \) such that \( k(i) = k, h^t(i) = h \) who is in a coalition (or block) \( A = A^t \) with type-age matrix \( \bar{a} = \bar{a}_{kh}, k = 1, \ldots, K, h = 1, \ldots, H \) be rewarded according to

\[ ^*\phi^AT_i(v^A) = \sum_{0 \leq \bar{b} \leq \bar{a}, \bar{b}_h > 0} \frac{\mu^\gamma(\bar{b}) \cdot \prod_{1 \leq k \leq K} \frac{(\bar{a}_k/\bar{b}_k)}{\bar{b}_k}}{|\{ k' \in K^A : \bar{o}_{h'} > 0 \}|} \cdot \frac{h}{\bar{b}_h \cdot \sum_{h' \in H^\bar{b}} h'}. \]

Here \( AT \) stands for age-type and \( \mu^\gamma(\bar{b}) \) is defined by expression (5) above. In words, for every type \( \bar{b} \) such that \( \bar{b}_h > 0 \) (as \( k(i) = k \)), the mass placed by Möbius inversion on all subcoalitions \( B \subseteq
A of this type (i.e., the sum $\mu^\gamma(\hat{b}) \cdot \prod_{1 \leq k'' \leq K} \binom{a_{k''}}{b_{k''}}$ of all values taken by $\mu^\gamma$ on such subcoalitions) is firstly divided equally over all types $k'$ such that $\hat{b}_{k'} > 0$. Next, within these types (net added) worth is further shared between ages $h' \in H^b$ so that for any two members of same type, if one is older than the other, then the former receives a higher payoff than the latter. In particular, the share assigned to type $k \in K^A$ is divided between ages $h \in H^A$ in a way such that for any $h, h' \in H^A$, $h < h'$, the sum of all payoffs assigned to players with age $h$ (and type $k$) divided by the sum of all payoffs assigned to players with age $h'$ (and type $k$) equals $\frac{h}{h'}$. Lastly, all agents of same age and type are rewarded the same, of course.

5 Agent behavior

While coalitional behavior (i.e., how to reward members) results from a rather sophisticated computational procedure (see above), agents’ behavior is mainly randomized, and results from very simple computations. Basically, it consists of a random part and a drift part. First of all, at each stage $t$ only $n \cdot \alpha$ agents consider whether to move from their current block or not, with $0 < \alpha < 1$ and $n \cdot \alpha \in \mathbb{Q}$, and those agents who make such a decision have to be randomly selected at each stage.

Concerning the random part, let $\epsilon$ be an arbitrarily small, strictly positive real number and $P^t = P = \{A_1, \ldots, A_{|P|}\}$ denote the generic coalition structure prevailing at some stage $t$. If an agent $i \in A \in P$ happens to be among those who decide whether to stay or move at stage $t$, then with probability $\epsilon$ she randomly draws a number $k \in \{0, 1, \ldots, |P|\}$ and gets into block $A_k$. In particular, if $i \in A_k$, then she remains in her current coalition. On the other hand, if $k = 0$, then she mutates, i.e., she forms a 1-cardinal block of coalition structure $P^{t+1}$ prevailing at stage $t + 1$.

Concerning the drift part, with probability $1 - \epsilon$ the agent compares the payoff she received in the previous stage $t - 1$ with that received (in $t - 1$) by a randomly chosen agent (of same age and type) from another block. If the former exceeds or equals the latter, then she remains in her current block. Otherwise, she moves into the other agent’s block. Clearly, rather than using only the payoff received in the previous stage, agents may well make comparisons in terms of a weighted average of the last $H$ payoffs they received, where $H$ is the memory of the model (see above). Concerning the initial stage $t = 0$, some random coalition structure $P^0$ has to be generated, and then agents have to be rewarded as members of its blocks, so that they have an initial payoff for making comparisons at stage $t = 1$, when the algorithm actually begins to work.

5.1 Fine-tuning $\epsilon$ and $\alpha$

As already mentioned, the idea behind the coalition structure generation process described here is that at each stage the only available information is the worth of all coalitions prevailing at that stage, as well as the worth of all proper subcoalitions of these latter. Poor though it is, such an information may be useful for checking whether the underlying coalitional game changes or not. In turn, whenever a change is perceived to have occurred, the exploring activity (aiming at identifying those coalitions with new greater worth) must increase, while it may be kept to a minimum otherwise. In other words, the generation process is asked to be adaptive. In turn, this implies that the system must constantly explore. Nevertheless, this exploration may be more or less intense depending on previous performance.

For the sake of concreteness, allow the algorithm to keep track, at any stage $t$, of the worth $f(P^{t'})$ attained on all $T$ previous coalition structures. That is to say, $\min\{0, t - T\} \leq t' \leq t - 1$, where $T$ is some multiple of the memory $H$ of the model. Also let $0 < \epsilon_\ast < 1$ be an arbitrarily small, strictly positive real number, and set

$$\epsilon(t) = \epsilon_\ast \text{ if } f(P^{t-1}) - f(P^{t-2}) \geq 0,$$

(7)
The measure of the algorithm performance over the last stage. In particular, if the underlying coalitional game changes at stage \( t \), then most likely \( f(P^t) - f(P^{t-1}) \leq 0 \). Accordingly, the poorer previous performance, the wider current exploration. Equivalently, the poorer previous performance, the greater the random component of collective behavior, as defined by \( \epsilon(t) \), with respect to the drift one, as defined by \( 1 - \epsilon(t) \). Conversely, if in the previous stage performance was (reasonably) good (i.e., if \( f(P^t) - f(P^{t-2}) \geq 0 \)), then current exploration may be left at its minimum \( \epsilon_\ast \).

The whole argument may be averaged as follows. Define

\[
\Delta^H f_{t-1} = \sum_{1 \leq h \leq H} \binom{H+1}{2} \cdot [f(P^t-h) - f(P^{t-h-1})],
\]

where \( \binom{H+1}{2} = \sum_{1 \leq h \leq H} h \) and \( H \) is the memory of the model. In words, \( \Delta^H f_{t-1} \) is a weighted average of the past \( H \) variations \( f(P^t-h) - f(P^{t-h-1}) \), \( 1 \leq h \leq H \), with weights strictly increasing in \( h \), so that the more recent the variation, the higher its weight. This weighted average provides a measure of the algorithm performance over the last \( H \) stages. Now let

\[
\epsilon(t) = \epsilon_\ast \text{ if } \Delta^H f_{t-1} \geq 0,
\]

\[
\epsilon(t) = \max\{\min\{\Delta^H f_{t'} : \min\{0, t - T\} \leq t' \leq t - 1\} : \epsilon_\ast\} \text{ otherwise.}
\]

Then, the above (non-averaged) fine-tuning for \( \epsilon(t) \) defined by (7) and (8) corresponds to the special case where \( H = 1 \) (and \( T \) is an arbitrary integer \( > 0 \)) in (9) and (10).

The same argument applies to \( \alpha \). For reasons of space, the only averaged form (paralleling (9) and (10)) is formalized hereafter. Let \( 0 < \alpha_\ast < 1 \) be such that \( n \cdot \alpha_\ast \in \mathbb{Q} \), and set

\[
\alpha(t) = \alpha_\ast \text{ if } \Delta^H f_{t-1} \geq 0,
\]

\[
\alpha(t) = \max\{\min\{\Delta^H f_{t'} : \min\{0, t - T\} \leq t' \leq t - 1\} : \alpha_\ast\} \text{ otherwise.}
\]

Then, at each stage \( t \), let \( \lfloor n \cdot \alpha(t) \rfloor \) be the number of agents who decide whether to move or stay at this stage, where \( \lfloor x \rfloor \) denotes the largest integer \( \leq x \). Again, given this number, which agents actually make the decision have to be randomly chosen. Accordingly, at each stage \( t \), there are (approximately) \( \lfloor n \cdot \alpha(t) \rfloor \cdot \epsilon(t) \) agents who make random movements. This may be thought as the amount of "resources" that the algorithm devotes to exploring, i.e., the random component. Similarly, there are (approximately) \( \lfloor n \cdot \alpha(t) \rfloor \cdot (1 - \epsilon(t)) \) agents who decide whether to move or
not depending on the result of a comparison. This is the drift component. Lastly, \(n - \lfloor n \cdot \alpha(t) \rfloor\) agents simply remain in the coalition inherited from the previous stage \(t - 1\).

In order to avoid any confusion, notice that although some agents are said to remain, at any stage \(t\), in the same coalition where they were at the previous stage \(t - 1\), still this does not mean that such a coalition remains itself the same from \(t - 1\) to \(t\). In fact, many agents, at each stage, leave a block and get into another one. Yet, those who do not move, are said to remain, of course. This is precisely the reason why coalitions are here termed “dynamic”.

6 Concluding remarks

This paper describes a novel coalition structure generation algorithm intended to face a challenging dynamic problem. In fact, as already mentioned, for given underlying coalitional game, searching optimal coalition structures is NP-hard. Yet, if the worth of coalitions is merely known to (possibly) vary over time, but no information about the frequency and magnitude of variations is available, then it is not clear how to define optimal time-patterns of coalition structures, as there is no way to generate coalition structures maximizing global worth at each stage. In other words, it is not clear how to define a solution of this problem. A reasonable approach is to compare different coalition structure generation processes in terms of their performance, thus providing an ideal setting where to conceive and conduct simulations. In fact, this shall be precisely the object of future research. Accordingly, this section is devoted to briefly survey possible simulation setups, avoiding details as far as possible.

Firstly, the underlying game must be type-symmetric. As already mentioned, this dramatically reduces its size (as input). For example, with 400 agents, a generic game consists of \(2^{400}\) real numbers, while if agents are partitioned into 10 types with 40 agents of each type, then a type-symmetric game consists of \(40^{10} = 2^{30} \cdot 5^{10}\) numbers.

Secondly, any of the underlying coalitional games (each prevailing over some interval during the simulation) must be rather simple (such as some type-symmetric sophistication of \(\gamma\) in Example 1), so that the worth of optimal coalition structures can be easily computed in advance. In fact, the performance of any algorithm may be primarily measured through the sum over all stages of the difference between the worth of optimal coalition structures on the one side, and the worth of those coalition structures the algorithm is able to generate on the other side. The larger this sum, the poorer performance.

A further issue concerns the comparison between different generation processes. In this respect, behavioral coalition structure generation, as described thus far, can be slightly modified itself, yielding an additional algorithm. In particular, while still taking into account membership-age and type, coalitions may well allocate their worth across members in a rather simple way, which does not make use of Möbius inversion (or, equivalently, of the Shapley value of restricted games), although this latter allows for the widest choice of efficient worth-sharing (see [5]), and enables to evaluate how agents (i.e., types) interact with each other within any coalition (see [9]; see also [8] for \(k\)-additive interaction). In fact, roughly speaking, coalitions might firstly divide their whole worth into a number of shares which equals the number of distinct types that they contain in a way such that scarcer types get more. Secondly, they may divide such shares between members of each type in a way such that elder get more. This is addressed in the appendix in terms of the potential approach.

From another viewpoint, behavioral coalition structure generation constitutes a decentralized approach to the dynamic problem described above. In fact, agents only know their payoff and coalitions only know the worth of all their subcoalitions (as well as their membership-age and type matrix, of course). In other words, neither agents nor coalitions ever get the overall picture. Conversely, a centralized approach would directly search optimal coalitional structures
through some branch-and-bound algorithm such as BOB (see [26]). Nevertheless, as already mentioned, BOB clearly needs some known value of coalitions (i.e., some known underlying coalitional game) as input in order to perform the search. But this search in NP-hard, i.e., it takes quite long, and in the meanwhile the actual game may well vary. This might be a more serious problem when the underlying game changes rather frequently (although possibly not much in magnitude, where this latter may be measured through the \( \ell_2 \)-norm \( \|v - v'\|_2 = \left( \sum_{A \in 2^N} (v(A) - v'(A))^2 \right)^{1/2} \), with \( v, v' \) denoting two generic games one of which follows after the other). Therefore, in practice, just like the behavioral algorithm described above, this dynamic search approach should also devote some resources (i.e., some agents) to exploration, i.e., to checking the actual worth of coalitions. Here again, the number of agents devoted to exploration at any stage could depend on most recent performance (see expressions (7) and (8) above with \( T = 1 \)), and might well be 0 as long as no change is perceived.

It must also be mentioned that a search restricted to the coarsest two levels of partition lattice \( P^N \) provides interesting results: in the worst case, the ratio of the worth of optimal coalition structures to the worth of the worthiest coalition structures searched in this way equals \( n \) (see [25]). Yet, there are \( 2^{n-1} \) coalition structures in the coarsest two levels of \( P^N \). Therefore, if \( n \) is large (and the frequency of changes in the underlying coalitional game is high), then a dynamic version of this non-exhaustive (faster) search might well fail to outperform some dynamic exhaustive (slower) one.

The next section is an appendix discussing the theoretical soundness of the generation process described thus far. It focuses on the potential of both coalitional games (possibly with given coalition structure) and coalition formation (strategic) games, showing how the Shapley value relates to optimal coalition structures and to their stability. Roughly speaking, this latter is intended as follows: if coalition members are rewarded according to the Shapley value of the associated restricted game, then once an optimal coalition structure is reached no player has an incentive to move. Although the Shapley value seems to cause some computational problems in dynamic coalition formation design (as it may be exponentially hard to compute, see [14], p.43), such problems might deserve further investigation in case computations are done through Möbius inversion.

**Acknowledgements**

I thank Stefano Arteconi, Lorenzo Donatiello, David Hales and Edoardo Mollona for helpful conversations.
Appendix: the potential approach

The potential originated in physics, and was subsequently adapted both to cooperative (see [4], [11], [12] and [32]) and non-cooperative or strategic games (see [21] and [30]). Here, it is useful for appreciating the link between efficient worth-sharing within coalitions and optimal (and stable) coalition structures.

Let $G^N$ denote the set of coalitional games $v : 2^N \to \mathbb{R}$. Traditional assumptions are $v(\emptyset) = 0$ and monotonicity, i.e., $B \subseteq A \subseteq N \Rightarrow v(B) \leq v(A)$. A potential is a mapping $\Psi : G^N \to \mathbb{R}$ defined by

$$\sum_{i \in N} D_i \Psi(v) = v(N) \text{ for every } v \in G^N,$$

where $D_i \Psi(v) = \Psi(v) - \Psi(v^{N \setminus i})$,

$v^{N \setminus i}(A) = v(A \setminus i)$ for every $A \in 2^N$ and $\Psi(v^\emptyset) = 0$. By induction on the number $n$ of players, it may be checked that there exists a unique potential taking form

$$\Psi(v) = \sum_{A \subseteq N} \frac{\mu^v(A)}{|A|}.$$ In turn, this yields

$$D_i \Psi(v) = \sum_{A \subseteq N \setminus i} \frac{\mu^v(A \cup i)}{|A| + 1} = \phi_i(v) \text{ for every } i \in N.$$

That is, there exists a unique potential (function), defined on the space of all games, with respect to which players’ marginal contributions are always efficient (i.e., add up to $v(N)$). In particular, such marginal contributions are defined by the Shapley value (see [11], [12] and [4]).

This argument extends to games with coalition structure $(v, P) \in G^N \times P^N$ as follows. Define a potential of games with coalition structure to be a mapping $\Psi : G^N \times P^N \to \mathbb{R}$ such that

$$\sum_{i \in N} D_i \Psi(v, P) = \sum_{A \in P} v(A) = f^v(P) \quad (11)$$

for every $(v, P) \in G^N \times P^N$ (see expression (1) section 2 above), where

$$D_i \Psi(v, P) = \Psi(v, P) - \Psi(v^{N \setminus i}, P^{N \setminus i}),$$

$P^{N \setminus i}$ is the partition of $N \setminus i$ induced by $P = \{A_1, \ldots, A_{|P|}\}$, that is to say, $P^{N \setminus i} = \{A_1 \setminus i, \ldots, A_{|P|} \setminus i\}$, and $\Psi(v^\emptyset, P^\emptyset) = 0$.

Claim 2 There exists a unique potential defined by

$$\Psi(v, P) = \sum_{A \in P} \sum_{B \subseteq A} \frac{\mu^v(B)}{|B|}$$

for every $(v, P) \in G^N \times P^N$, implying

$$D_i \Psi(v, P) = \sum_{B \subseteq A \setminus i} \frac{\mu^v(B \cup i)}{|B| + 1} = \phi_i(v^A)$$

for every $i \in A$ and every $A \in P$. 

UBLCS-2006-27
Proof. By induction on the number \(|N| = n\) of players. For \(n = 1\) the statement is easily checked to be true. Accordingly, assume it is also true for all \(n < k \geq 1\). Then, for \(n = k\)

\[
\sum_{1 \leq i \leq k} [\Psi(v, P) - \Psi(v_{N\setminus i}, P_{N\setminus i})] = \sum_{A \in P} v(A) \Rightarrow
\]

\[
\Rightarrow k \cdot \Psi(v, P) - \sum_{1 \leq i \leq k} \sum_{A \in P_{N\setminus i}} \sum_{B' \subseteq A'} \mu^v(B') = \sum_{A \in P} \sum_{B \subseteq A} \mu^v(B) \Rightarrow
\]

\[
\Rightarrow k \cdot \Psi(v, P) = \sum_{A \in P} \sum_{B \subseteq A} \mu^v(B) \cdot (1 + \frac{k - |B|}{|B|}) \Rightarrow \Psi(v, P) = \sum_{A \in P} \sum_{B \subseteq A} \mu^v(B) \frac{|B|}{B'}
\]
as desired. (See also [32].) ■

This result reads as follows: if, for any given coalitional game \(v\), agents’ behavior (globally) tends to coalition structures \(P\) where \(\sum_{A \in P} \sum_{i \in A} \phi_i(v^A)\) attains a maximum, then such a behavior also tends to optimal coalition structures.

The potential approach extends to strategic games, where it yields a stronger result concerning the stability of optimal coalition structures. A strategic game \(\Gamma\) with player set \(N\) is a triple \(\Gamma = (N, S, u)\) where every player \(i \in N\) has strategy set \(S_i\), and thus \(S = S_1 \times \cdots \times S_n\) is the set of \(n\)-tuple of strategies, while \(u : S \to R^n\), with \(u(s) = (u_1(s), \ldots, u_n(s))\), is such that \(u_i(s)\) is the utility (representing preferences) of player \(i \in N\) when the \(n\)-tuple of chosen strategies is \(s \in S\). A strategic game is a potential game if it admits a (ordinal) potential, i.e., a function \(\Psi : S \to R\) such that for every player \(i \in N\), for every pair of strategies \(s_i, s'_i \in S_i\) for this player, and for every \(n-1\)-tuple \(s_{-i} = \{s_j : j \in N \setminus i\}\) of strategies for other players, the following holds:

\[
\Psi(s_i, s_{-i}) - \Psi(s'_i, s_{-i}) > 0 \Leftrightarrow u_i(s_i, s_{-i}) - u_i(s'_i, s_{-i}) > 0,
\]

(see [21]). Potential games admit at least one (pure-strategy Nash) equilibrium. Put it differently, if a potential function exists, then it surely attains a maximum (of course). In fact, if \(\Gamma = (N, S, u)\) admits a potential \(\Psi\), then any equilibrium of game \(\Gamma = (N, S, u^\phi)\), where \(u^\phi_i = \Psi\) for every \(i \in N\), is also an equilibrium of potential game \(\Gamma\). That is, the original game and the game where each player’s utility is replaced with the potential itself have the same equilibria. A potential \(\Psi\) is said exact if

\[
\Psi(s_i, s_{-i}) - \Psi(s'_i, s_{-i}) = u_i(s_i, s_{-i}) - u_i(s'_i, s_{-i})
\]

for every \(i \in N\), for every pair \(s_i, s'_i \in S_i\), and for every \(n-1\)-tuple of strategies for players \(s_{-i} = \{s_j : j \in N \setminus i\}, j \in N \setminus i\).

In strategic coalition formation games, some coalitional game \(v : 2^N \to R\) is given, and the strategy set is

\[
S_i = 2^N \setminus 2^{N\setminus i} = \{A \subseteq N : i \in A\}
\]

for every \(i \in N\). That is, every player has to choose some coalition to join. A full specification of the game requires to define: (1) a mechanism mapping \(n\)-tuple of chosen coalitions onto coalition structures, and (2) players’ payoffs as members of each of the \(2^{n-1}\) coalitions they may join.

Concerning (1), there are two main mechanisms:

(M1) Given \(n\)-tupla \(s = (A^1, \ldots, A^n) \in 2^N \setminus 2^{N\setminus 1} \times \cdots \times 2^N \setminus 2^{N\setminus n}\) of chosen coalitions to join, the resulting coalition structure \(P_{s,M1} \in P^N\) obtains as follows: for every \(A \in 2^N\)

\[
A \in P_{s,M1} \iff A_i = A \text{ for every } i \in A.
\]

That is, a coalition forms iff (if and only if) all members have chosen to join precisely that coalition. Otherwise, if \(A_j = A'_i\) for some \(j \in A^i\setminus i\), then player \(i\) ends up isolated, i.e., constituting a 1-cardinal block of \(P_{s,M1}\) (see [19]).
Concluding remarks

(M2) Given an n-tupla \( s = (A^1, \ldots, A^n) \in 2^N \setminus 2^N \setminus 1 \times \cdots \times 2^N \setminus 2^N \setminus n \) of chosen coalitions to join, the resulting coalition structure \( P_{s,M2} \in P^N \) obtains as follows: for every \( i, j \in N \)

\[ A^i = A^j \iff \{i, j\} \subseteq A \in P_{s,M2}. \]

That is, any two players choosing to join the same coalition end up members of the same block of the prevailing partition \( P_{s,M2} \). In this case, in order for a player \( i \in N \) to end up isolated, there must be no other player \( j \in N \setminus i \) choosing to join the same coalition as \( i \).

Concerning (2), the issue is how to efficiently reward each player for each coalition structure \( P = P_{s,M1} \) or \( P_{s,M2} \) that may prevail through mechanisms (M1) or (M2). That is, for each prevailing block \( A \in P \), and for each member \( i \in A \) of such a block, a share \( \phi_i(v^A) \) must be specified such that

\[ \sum_{i \in A} \phi_i(v^A) = v(A), \]

where \( v^A(B) = v(A \cap B) \) for every \( B \in 2^N \) is a restricted game (see above).

In particular, consider the following two sharing rules:

(S1) The worth of singletons, if any, is given to singletons. The remaining worth, i.e., the worth of each block minus the sum over members of their own worth, as singletons, is divided equally among members. That is,

\[ \phi^{S1}_i(v^A) = v(i) + \frac{v(A) - \sum_{j \in A} v(j)}{|A|}. \]

(S2) The worth of each block \( A \) of the prevailing partition is shared between members according to the Shapley value of \( v^A \). That is,

\[ \phi^{S2}_i(v^A) = \sum_{B \subseteq 2^A \setminus 1} \frac{\mu^v(B)}{|B|} = \phi_i(v^A). \]

(This is, in fact, the Aumann-Dreze value for games with coalition structure, see \([2]\) and above).

It turns out that the coalition formation game defined by mechanism (M1) and sharing rule (S1) is a potential game. In particular, if \( v(i) = 0 \) for every \( i \in N \), then an exact potential for this game is \( \Psi^1 \) defined by

\[ \Psi^1(s) = \sum_{A \in P_{s,M1}} \frac{v(A)}{|A|} \]

for every \( s \in 2^N \setminus 2^N \setminus 1 \times \cdots \times 2^N \setminus 2^N \setminus n \) (see \([30]\)). Similarly,

**Claim 3** The coalition formation game defined by mechanism (M2) and sharing rule (S2) is a potential game. In particular, an exact potential for this game is \( \Psi^2 \) defined by

\[ \Psi^2(s) = \sum_{A \in P_{s,M2}} \sum_{B \subseteq A} \frac{\mu^v(B)}{|B|} \]

for every \( s \in 2^N \setminus 2^N \setminus 1 \times \cdots \times 2^N \setminus 2^N \setminus n \).

**Proof.** Let \( s, s' \in 2^N \setminus 2^N \setminus 1 \times \cdots \times 2^N \setminus 2^N \setminus n \) be such that \( s_j = s'_j \) for every \( j \in N \setminus i \) while \( s_i \neq s'_i \) for some \( i \in N \). For singletons \( i \in N \), let \( i \in 2^N \) denote the corresponding 1-cardinal coalition. Conversely, when such a singleton is regarded as a 1-cardinal block of some prevailing partition \( P \), parentheses are used, i.e., \( \{i\} \in P \). There are three distinct cases to check:
Case (i): \( \{ i \} \in P_{s,M_2}, P'_{s',M_2} \). It is immediately observed that

\[
\Psi^2(s) - \Psi^2(s') = u_i(s) - u_i(s') = 0.
\]

Case (ii): \( \{ i \} \notin P_{s,M_2} \) but \( \{ i \} \in P'_{s',M_2} \) (or, equivalently, \( \{ i \} \in P_{s,M_2} \) but \( \{ i \} \notin P'_{s',M_2} \)). Accordingly, if \( i \in A \in P_{s,M_2} \), then

\[
\Psi^2(s) - \Psi^2(s') = \sum_{D \subseteq 2^{A \setminus \{i\}}} \frac{\mu^v(D)}{|D|} = \phi^S_2(v^A) - \phi^S_2(v^i) = u_i(s) - u_i(s').
\]

Case (iii): There are \( A, B \in 2^N \setminus 2^{N \setminus \{i\}} \) such that \( |A|, |B| \geq 2, A \in P_{s,M_2} \) and \( B \in P'_{s',M_2} \). Then, given mechanism \( M_2 \), it must be \( B \setminus i \in P_{s,M_2}, A \setminus i \in P'_{s',M_2} \). In turn, this implies \( \Psi^2(s) - \Psi^2(s') = \sum_{D \subseteq 2^A \setminus 2^{A \setminus \{i\}}} \frac{\mu^v(D)}{|D|} - \sum_{D' \subseteq 2^B \setminus 2^{B \setminus i}} \frac{\mu^v(D')}{|D'|} = \phi^S_2(v^A) - \phi^S_2(v^B) = u_i(s) - u_i(s'), \)

thus completing the proof. ■

Hence, non-cooperative game theory offers two coalition formation games that are potential games, and thus admit at least one equilibrium. In particular, every equilibrium corresponds to a stable coalition structure. Each of these games is defined through (i) a mechanism mapping \( n \)-tuple of chosen coalitions onto coalition structures and (ii) a sharing rule. In fact, in view of claim 2 above, mechanism \( M_2 \) and sharing rule \( S_2 \) define a potential game whose equilibria correspond to optimal coalition structures. Furthermore, mechanism \( M_2 \) is definitely more aggregating than \( M_1 \), as in the former any two players choosing a common coalition to join do join (possibly into a proper subset of the chosen coalition), while in the latter a coalition forms iff all members have chosen precisely that coalition.

The idea of large populations leads to relax the assumption of purely strategic behavior when designing coalition structure generation. In particular, apart from the random/exploring part, in the model described here agents basically make periodic comparisons with (randomly chosen) other agents, moving into the other agent’s block whenever this latter receives a strictly higher payoff. This means that the mechanism mapping \( n \)-tuple of chosen coalitions (or strategies) onto coalition structures is non-strategic, in some sense, in that coalitions always accept new members. In this respect, the model is very aggregating. That is to say, it is much ‘closer’ to mechanism \( M_2 \) than to \( M_1 \). All this leads to conclude that a coalition structure generation process based on sharing rule \( S_2 \) (and mechanism \( M_2 \)) should outperform one based on sharing rule \( S_1 \) (and mechanism \( M_1 \)).

References


REFERENCES


