



TI 2008-104/1

Tinbergen Institute Discussion Paper

An Algorithm for Computing the Nucleolus of Disjunctive Additive Games with an Acyclic Permission Structure

*René van den Brink*¹

*Ilya Katsev*²

*Gerard van der Laan*¹

¹ VU University Amsterdam, and Tinbergen Institute;

² St. Petersburg Institute for Economics and Mathematics, Russian Academy of Sciences.

Tinbergen Institute

The Tinbergen Institute is the institute for economic research of the Erasmus Universiteit Rotterdam, Universiteit van Amsterdam, and Vrije Universiteit Amsterdam.

Tinbergen Institute Amsterdam

Roetersstraat 31
1018 WB Amsterdam
The Netherlands
Tel.: +31(0)20 551 3500
Fax: +31(0)20 551 3555

Tinbergen Institute Rotterdam

Burg. Oudlaan 50
3062 PA Rotterdam
The Netherlands
Tel.: +31(0)10 408 8900
Fax: +31(0)10 408 9031

Most TI discussion papers can be downloaded at
<http://www.tinbergen.nl>.

An algorithm for computing the nucleolus of disjunctive additive games with an acyclic permission structure¹

René van den Brink² Ilya Katsev³ Gerard van der Laan⁴

October 31, 2008

¹This research was partly carried out while the second author was visiting the Tinbergen Institute, VU University Amsterdam, on NWO-grant 047.017.017 within the framework of Dutch-Russian cooperation.

²J.R. van den Brink, Department of Econometrics and Tinbergen Institute, VU University, De Boelelaan 1105, 1081 HV Amsterdam, The Netherlands. E-mail: jrbrink@feweb.vu.nl

³I.V. Katsev, St. Petersburg Institute for Economics and Mathematics, Russian Academy of Sciences, Tchaikovsky st. 1, 191187 St. Petersburg. E-mail:katsev@yandex.ru

⁴G. van der Laan, Department of Econometrics and Tinbergen Institute, VU University, De Boelelaan 1105, 1081 HV Amsterdam, The Netherlands. E-mail: glaan@feweb.vu.nl

Abstract

A situation in which a finite set of players can obtain certain payoffs by cooperation can be described by a *cooperative game with transferable utility*, or simply a TU-game. A (single-valued) solution for TU-games assigns a payoff distribution to every TU-game. A well-known solution is the nucleolus.

A cooperative *game with a permission structure* describes a situation in which players in a cooperative TU-game are hierarchically ordered in the sense that there are players that need permission from other players before they are allowed to cooperate. The corresponding restricted game takes account of the limited cooperation possibilities by assigning to every coalition the worth of its largest feasible subset.

In this paper we consider the class of non-negative additive games with an acyclic permission structure. This class generalizes the so-called peer group games being non-negative additive games on a permission tree. We provide a polynomial time algorithm for computing the nucleolus of every restricted game corresponding to some disjunctive non-negative additive game with an acyclic permission structure. We discuss an application to market situations where sellers can sell objects to buyers through a directed network of intermediaries.

Keywords: TU-game, additive game, acyclic permission structure, disjunctive approach, peer group game, nucleolus, algorithm, complexity.

AMS subject classification: 91A12, 5C20

JEL code: C71

1 Introduction

A *cooperative game with transferable utility*, or simply a TU-game, is a finite set of players and for any subset (coalition) of players a worth representing the total payoff that the coalition can obtain by cooperating. A payoff vector is a vector which gives a payoff to each of the players, i.e., each component corresponds to precisely one of the players. A payoff vector is efficient if the sum of the payoffs is equal to the worth of the grand coalition consisting of all players. A set-valued solution for TU-games assigns a set of payoff vectors (possibly empty) to every TU-game. A single-valued solution assigns precisely one payoff vector to every TU-game. A solution is said to be efficient if for every game any payoff vector assigned by the solution is efficient. The most well-known efficient set-valued solution is the *Core* (Gillies, 1953). The two most well-known efficient single-valued solutions are the *Shapley value* (Shapley, 1953) and the *nucleolus* (Schmeidler, 1969).

In this paper we assume that the players in a TU-game are part of some hierarchical structure that is represented by a directed graph such that some players need permission from other players before they are allowed to cooperate within a coalition. In the literature two approaches to these *games with a permission structure* can be found. In the *conjunctive* approach, as considered in Gilles, Owen and van den Brink (1992) and van den Brink and Gilles (1996), it is assumed that each player needs permission from all its predecessors in the directed graph before it is allowed to cooperate. Alternatively, in the *disjunctive* approach as developed in Gilles and Owen (1994) and van den Brink (1997), a player needs permission to cooperate of at least one of its direct predecessors (if it has any). So, according to the conjunctive approach a coalition is feasible if and only if for any player in the coalition, all its predecessors are also in the coalition, whereas according to the disjunctive approach a coalition is feasible if and only if for any player in the coalition at least one of its predecessors (if it has any) is also in the coalition. Following an approach similar to that of Myerson (1977) for games with limited communication (graph) structure, in both the conjunctive and disjunctive approach to games with a permission structure a restricted game is derived. In games with a permission structure the restricted game assigns to every coalition the worth of its largest feasible subset. Applying well-known solutions as the Shapley value, Core or nucleolus to such restricted games yields solutions for games with a permission structure.

A special subclass of games with a permission structure arises from peer group situations, as introduced in Brânzei, Fragnelli and Tijs (2002). A *peer group situation* is a triple consisting of a set of players, a hierarchical structure represented by a rooted (directed) tree, and for each player a non-negative real number representing its potential individual (economic) contribution to the society of all players. This yields an associated TU-game being the additive game in which the worth of any coalition is equal to the sum

of the individual potentials of its members. In a rooted directed tree there is one top node (not having a predecessor), while any other node has precisely one predecessor. So, in case the hierarchical structure on the player set is a rooted directed tree, the conjunctive approach and the disjunctive approach as described above, coincide. The restricted game of the associated TU-game with respect to such a permission structure is called a *peer group game*. These peer group games have many interesting applications, see Brânzei *et al.* (2002). Clearly, in a peer group game the worth of a coalition is the sum of the individual potentials of the members of the largest feasible subset of the coalition. Since the top player is always in this set when it belongs to the coalition, and the largest feasible set is the empty set for any coalition not containing the unique top player, it follows that the top player is a veto player, i.e., any coalition not containing the top player has zero worth in the restricted game. In this paper we generalize peer group games by considering non-negative additive games on acyclic permission structures (which are not necessarily trees). Consequently, these games need not to contain a veto player¹.

In Arin and Feltkamp (1997) an exponential time algorithm has been given for veto-rich games, i.e., games that have at least one veto player. For the special case of a peer group game a polynomial time algorithm is given in Brânzei, Solymosi and Tijss (2005) and for a class of games satisfying so-called weak digraph monotonicity and weak digraph concavity conditions with respect to acyclic, quasi-strongly connected permission structures, a polynomial time algorithm is given in van den Brink, Katsev and van der Laan (2008). This latter class of games is more general than the class of peer group games, but the acyclic permission structure is still required to be quasi-strongly connected. In this paper we provide a polynomial time algorithm to compute the nucleolus of the restricted game induced by situations in which the associated TU-game is still additive, but in which we allow for any acyclic permission structure, so allowing more than one top player. This considerably widens the applications, for example we can consider situations where sellers can sell objects to buyers through a directed network of intermediaries. Whereas quasi-strongly connected networks only can be used for situations with one seller, weakening this by only requiring that the network is acyclic, we can also study such situations with more than one seller. The algorithm presented in this paper computes the nucleolus in polynomial time through a number of iterations. In each iteration a subgame with an acyclic, quasi-strongly connected permission structure is considered and the algorithm developed in van den Brink *et al.* (2008) is used to compute the nucleolus for this subgame.

This paper is organized as follows. Section 2 is a preliminary section containing cooperative TU-games (with special attention for the nucleolus), directed graphs and games

¹A further generalization are the so-called interior operator games where the game is non-negative and additive and the set of feasible coalitions is an antimatroid, see Bilbao *et al.* (2005, 2008).

with a permission structure. In Section 3 we provide the algorithm to find the nucleolus for disjunctive non-negative additive games with an acyclic permission structure. Section 4 shows that the algorithm indeed computes the nucleolus for this class of games. In Section 5 we discuss some properties of the algorithm, while Section 6 discusses its complexity, showing that it indeed finds the nucleolus in polynomial time. Section 7 illustrates the algorithm with an example of a market situation where sellers can sell objects to buyers through a directed network of intermediaries. Finally, there is an appendix with the algorithm of van den Brink *et al.* (2008) that is used in each iteration of the algorithm to compute the nucleolus of a non-negative additive subgame with an acyclic, quasi-strongly connected permission structure.

2 Preliminaries

2.1 TU-games

A situation in which a finite set of players can obtain certain payoffs by cooperating can be described by a *cooperative game with transferable utility*, or simply a TU-game, being a pair (N, v) , where $N \subset \mathbb{N}$ is a finite set of n players and $v: 2^N \rightarrow \mathbb{R}$ is a characteristic function on N such that $v(\emptyset) = 0$. For any coalition $S \subseteq N$, $v(S)$ is the worth of coalition S , i.e., the members of coalition S can obtain a total payoff of $v(S)$ by agreeing to cooperate. For simplicity, for a single player i we denote its worth $v(\{i\})$ by $v(i)$. When there is no confusion about the player set N we often denote the game (N, v) just by its characteristic function v . We denote the collection of all characteristic functions on N by \mathcal{G}^N and $n = |N|$ denotes the cardinality of N . Further, for game $v \in \mathcal{G}^N$ and coalition $K \subseteq N$ we define the *subgame* $v_K \in \mathcal{G}^K$ by $v_K(S) = v(S)$ for all $S \subseteq K$.

A payoff vector for a game is a vector $x \in \mathbb{R}^n$ assigning a payoff x_i to every $i \in N$. In the sequel, for $S \subseteq N$ we denote $x(S) = \sum_{i \in S} x_i$. A payoff vector is efficient if $x(N) = v(N)$ and it is individually rational if $x_i \geq v(i)$ for every $i \in N$. The imputation set $I(N, v)$ of game (N, v) is given by

$$I(N, v) = \{x \in \mathbb{R}^n \mid x(N) = v(N) \text{ and } x_i \geq v(i) \text{ for every } i \in N\},$$

i.e., $I(N, v)$ is the set of all efficient and individually rational payoff vectors. A (set-valued) solution F on \mathcal{G}^N assigns a set $F(N, v) \subset \mathbb{R}^n$ of payoff vectors to every characteristic function $v \in \mathcal{G}^N$. The most well-known set-valued solution is the Core assigning to every $v \in \mathcal{G}^N$ the set

$$C(N, v) = \{x \in I(N, v) \mid x(S) \geq v(S) \text{ for all } S \subset N\},$$

i.e., it is the set of all imputations that are stable in the sense that no coalition can do better by separating from the grand coalition. The Core of (N, v) is non-empty if and only if the game is *balanced*, see e.g. Bondareva (1962) or Shapley (1967).

Another (set-valued) solution is the *Kernel* assigning to every $v \in \mathcal{G}^N$ the set

$$K(N, v) = \{x \in I(N, v) \mid [s_{ij}(x) = s_{ji}(x)] \text{ or } [s_{ij}(x) > s_{ji}(x) \text{ and } x_j = v(j)] \text{ for all } i, j \in N\},$$

i.e., the set of all imputations such that for each pair of players i and j the *complaint* $s_{ij}(x) = \max\{v(S) - x(S) \mid i \in S, j \notin S\}$ of i against j is at least equal to the complaint of j against i , with equality whenever j gets more than its individual worth $v(j)$.

A solution F is said to be single-valued if it assigns to any $v \in \mathcal{G}^N$ a unique payoff vector. One of the two most well-known single-valued solutions is the *nucleolus* (Schmeidler, 1969).² To define the nucleolus, consider a given characteristic function $v \in \mathcal{G}^N$, and let $x \in \mathbb{R}^n$ be a payoff vector. Then the excess $e(S, x)$ of a coalition $S \subseteq N$ is defined by

$$e(S, x) = v(S) - x(S).$$

Further, let $E(x)$ be the $(2^n - 2)$ -component vector that is composed of the excesses of all coalitions $S \subset N$, $S \neq \emptyset$, in a non-increasing order, so $E_1(x) \geq E_2(x) \geq \dots \geq E_{2^n-2}(x)$. Then the nucleolus $Nuc(N, v)$ of the game (N, v) is the unique imputation which lexicographically minimizes the vector-valued function $E(\cdot)$ over the imputation set. Formally,

$$Nuc(N, v) = \{x \in I(N, v) \text{ such that } E(x) \preceq_L E(y) \text{ for all } y \in I(N, v)\},$$

where \preceq_L denotes the lexicographic order of vectors. It is well-known that $Nuc(N, v) \in K(N, v)$ and that $Nuc(N, v) \in C(N, v)$ when $C(N, v) \neq \emptyset$. So, when the game has a nonempty Core, then the nucleolus is in the intersection of the Kernel and the Core.

In a game (N, v) , a coalition S is called *inessential* if it has a partition $\{S_1, \dots, S_r\}$ with $r \geq 2$, such that $v(S) \leq \sum_{j=1}^r v(S_j)$. Coalitions which are not inessential are called *essential*. Notice that single player coalitions are always essential. It is straightforward to observe that for an inessential coalition S it holds that

$$e(S, x) \leq \sum_{j=1}^r e(S_j, x), \text{ for all } x \in \mathbb{R}^n.$$

Therefore the Core, and thus also the nucleolus, is independent of inessential coalitions, as was noticed by Huberman (1980). In fact, in any n player game there are at most $(2n - 2)$ coalitions which actually determine the nucleolus, see Brune (1983) and Reijnierse and Potters (1998). However, as noticed by Brânzei *et al.* (2005), identifying these coalitions is no less laborious as computing the nucleolus itself.

²The other one is the *Shapley value* (Shapley, 1953).

2.2 Directed graphs

A directed graph or *digraph* is a pair (N, D) where $N = \{1, \dots, n\}$ is a finite set of nodes (representing the players) and $D \subseteq N \times N$ is a binary relation on N . Given (N, D) and $S \subseteq N$, the graph $(S, D(S))$ is the subgraph on S given by $D(S) = \{(i, j) \in D \mid i, j \in S\}$. In the sequel we simply refer to D for a digraph (N, D) and to $D(S)$ for the subgraph $(S, D(S))$. For $i \in N$ the nodes in $S_D(i) := \{j \in N \mid (i, j) \in D\}$ are called the *successors* of i , and the nodes in $P_D(i) := \{j \in N \mid (j, i) \in D\}$ are called the *predecessors* of i . For a set $T \subseteq N$, let $S_D(T) = \cup_{i \in T} S_D(i)$ denote the union of the sets of successors of the players in T , respectively $P_D(T) = \cup_{i \in T} P_D(i)$ the set of all predecessors the players in T . Further, $T_D = \{i \in N \mid P_D(i) = \emptyset\}$ denotes the set of *top nodes* in D , being the set of nodes not having a predecessor.

For given D on N , a *path* between i and j in N is a sequence of distinct nodes (i_1, \dots, i_m) such that $i_1 = i$, $i_m = j$, and $\{(i_k, i_{k+1}), (i_{k+1}, i_k)\} \cap D \neq \emptyset$ for $k = 1, \dots, m-1$. A set of nodes $T \subseteq N$ is *connected* in digraph D if there is a path between any two nodes in T that only uses arcs between nodes in T , i.e., if for every $i, j \in T$ there is a path (i_1, \dots, i_m) between i and j such that $\{i_1, \dots, i_m\} \subseteq T$. A *component* in D is a maximally connected set T of nodes, i.e., T is connected and $T \cup \{i\}$ is not connected for every $i \in N \setminus T$. A path (i_1, \dots, i_m) from i_1 to i_m is a *directed path* in D if $(i_k, i_{k+1}) \in D$ for $k = 1, \dots, m-1$. For a set of nodes $T \subset N$ and $i \notin T$, we call a directed path (i_1, \dots, i_m) a path from T to i if $i_1 \in T$ and $i_m = i$. For every (directed) path $p = (i_1, \dots, i_m)$ we denote the set of nodes on that path by $H(p) = \{i_1, \dots, i_m\}$. A directed path (i_1, \dots, i_m) , $m \geq 1$, in D is a *cycle* in D if $(i_m, i_1) \in D$. We call digraph D *acyclic* if it does not contain any cycle. Note that acyclicity of a digraph D implies that D is irreflexive, i.e., $(i, i) \notin D$ for all $i \in N$. The collection of all acyclic digraphs on N is denoted by \mathcal{D}^N . For any $D \in \mathcal{D}^N$ the set of top nodes T_D is not empty and for every $i \notin T_D$ there is a path from T_D to i . The collection of all paths p from T_D to $i \notin T_D$ is denoted by \mathcal{P}_i .

A digraph is called *quasi-strongly connected* if there exists a node $i_0 \in N$, such that for every $j \neq i_0$ there is a directed path from i_0 to j . When D is, moreover, acyclic then $T_D = \{i_0\}$, i.e., i_0 is the unique top node in N . A quasi-strongly connected digraph $D \in \mathcal{D}^N$ is a rooted directed *tree* with the root being the top node i_0 if there is precisely one path from top-node i_0 to every other node.

By $\widehat{S}_D(i)$ we denote the set of nodes that can be reached from i by a directed path, i.e., $j \in \widehat{S}_D(i)$ if and only if there exists a sequence of players (h_1, \dots, h_t) such that $h_1 = i$, $h_{k+1} \in S_D(h_k)$ for all $1 \leq k \leq t-1$, and $h_t = j$.³ The players in $\widehat{S}_D(i)$ are called the *subordinates* of i , and the players in $\widehat{P}_D(i) := \{j \in N \mid i \in \widehat{S}_D(j)\}$ are called the *superiors* of i . For two different nodes i and j , we say that node $j \in N$ is a *complete subordinate*

³So, $\widehat{S}_D(i)$ is the set of successors of i in the *transitive closure* of D .

of node $i \in N$ in D if every path from T_D to j contains node i . The set of complete subordinates of node i is denoted by $\overline{S}_D(i)$, i.e.,

$$\overline{S}_D(i) = \{j \in N \setminus \{i\} \mid i \in H(p) \text{ for every } p \in \mathcal{P}_j\}.$$

Also we call the players in the set

$$\overline{P}_D(i) = \{j \in N \setminus \{i\} \mid j \in H(p) \text{ for every } p \in \mathcal{P}_i\}$$

the *complete superiors* of i in D .

2.3 Restricted games

In this paper we assume that the players in a TU-game are part of a hierarchical structure that is represented by a directed graph, referred to as a *permission structure*, such that some players need permission from other players before they are allowed to cooperate within a coalition. A triple (N, v, D) with (N, v) a TU-game and (N, D) a digraph with the player set N as the set of nodes is called a *game with a permission structure*. In the sequel we assume that $D \in \mathcal{D}^N$.

Assumption 2.1 (N, D) is acyclic.

As noticed in the introduction we can distinguish between the conjunctive and disjunctive approach. In this paper we consider the *disjunctive* approach as developed in Gilles and Owen (1994) and van den Brink (1997), where a player needs permission to cooperate of at least one of its direct predecessors (if it has any). Therefore a coalition is feasible if and only if for any player in the coalition at least one of its predecessors (if it has any) is also in the coalition. So, for permission structure (N, D) , the set of disjunctive feasible coalitions is given by

$$\Phi^D = \{S \subseteq N \mid P_D(i) \cap S \neq \emptyset \text{ for all } i \in S \setminus T_D\}.$$

For any $S \subseteq N$, let $\sigma(S) = \cup\{T \in \Phi^D \mid T \subseteq S\}$ be the largest disjunctive feasible subset of S in D .⁴ By Assumption 2.1 we have that for every $S \subseteq N$ with $\sigma(S) \neq \emptyset$, the subgraph $(\sigma(S), D(\sigma(S)))$ is acyclic.

Given the triple (N, v, D) with $v \in \mathcal{G}^N$ and $D \in \mathcal{D}^N$, under the disjunctive permission structure the induced *restricted game* $r: 2^N \rightarrow \mathbb{R}$ is given by

$$r(S) = v(\sigma(S)) \text{ for all } S \subseteq N. \tag{2.1}$$

⁴Every coalition having a unique largest feasible subset follows from Φ^D being closed under union.

If $|T_D| = 1$ then D is quasi-strongly connected and for the unique top player i_0 it holds that $r(S) = 0$ when $i_0 \notin S$, i.e., the restricted game is a veto-rich game with top-player i_0 as veto player.⁵ If D is a rooted directed tree, then $|P_D(i)| = 1$ for all $i \neq i_0$ and the conjunctive and disjunctive approach coincide. In this case the triple (N, v, D) is a *peer group situation* when the game (N, v) is a non-negative additive game (see Brânzei *et al.* (2002)), i.e., there exist non-negative real numbers a_i , $i \in N$, such that

$$v(S) = \sum_{i \in S} a_i, \quad S \subseteq N.$$

Then the restricted game (N, r) as defined in (2.1) is a so-called *peer group game* and is given by

$$r(S) = v(\sigma(S)) = \sum_{\{i \in S \mid \widehat{P}_D(i) \subseteq S\}} a_i.$$

A peer group game (N, r) is a monotone veto-rich game and has a non-empty Core. In particular the payoff vector $x \in \mathbb{R}_+^n$ given by $x_{i_0} = v(N)$ and $x_i = 0$, $i \neq i_0$, belongs to the Core. On the other hand it holds that $r(S) \leq \sum_{i \in S} a_i$ and thus also the payoff vector $x \in \mathbb{R}_+^n$ given by $x_i = a_i$ for all $i \in N$ belongs to the Core. As noticed in Brânzei *et al.* (2002), this class of games with permission structure contains several interesting applications. Also the two games v and its restriction r are games as considered in Ni and Wang (2007) to study polluted river games. In particular, their LR polluted river is a non-negative additive game, and their DR polluted river game is the restriction R of this game on the permission structure that represents the linear ordering of the players along the river from upstream to downstream.

3 A polynomial time algorithm for the nucleolus based on quasi-strongly connectedness

In the remaining of this paper we consider a non-negative additive game with acyclic permission structure (N, v, D) . Since $\sigma(S) \subseteq S$ for any $S \subseteq N$, also in this case we have that $r(S) = v(\sigma(S)) \leq \sum_{i \in S} a_i$. Therefore the payoff vector $x \in \mathbb{R}_+^n$ given by $x_i = a_i$ for all $i \in N$ is in the Core and thus the Core is nonempty and contains the nucleolus.

We first show that for any non-negative additive game with acyclic permission structure (N, v, D) , there exists a subset $K \subseteq N$ with the properties that (i) the subgraph $(K, D(K))$ is an acyclic, quasi-strongly connected permission structure and (ii)

⁵A game (N, v) is a *veto-rich* game if it contains a *veto player* being a player i such that $v(S) = 0$ when $i \notin S$.

$\sum_{j \in K} x_j = \sum_{j \in K} a_j$ for any payoff vector $x \in \mathbb{R}^n$ in the Core of the restricted game (N, r) , so also for the nucleolus. These properties play an important role in an algorithm to compute the nucleolus of the restricted game obtained from a non-negative additive game with acyclic permission structure (N, v, D) . In fact the nucleolus is obtained by computing the nucleolus of a sequence of smaller non-negative additive games with quasi-strongly connected permission structures. By definition the properties hold for $K = N$ when the graph has only one top player. So, in the sequel of this section we only consider the case that $|T_D| \geq 2$. The first lemma is obvious and stated without proof.

Lemma 3.1 *For every acyclic permission structure (N, D) it holds: j is a (complete) superior of i if and only if i is a complete subordinate of j .*

For a top player $t \in T_D$, let the set U^t be defined by $\overline{S}_D(t) \cup \{t\}$, i.e., U^t contains top player t together with all its complete subordinates. Observe that $T_D \setminus U^t \neq \emptyset$, because $|T_D| \geq 2$ and $(T_D \setminus \{t\}) \cap U^t = \emptyset$. Further, for $i \in \overline{S}_D(t)$, define

$$U_i = \cup \{H(p) \mid p \text{ is a directed path from } t \text{ to } i\},$$

i.e., the set $U_i \subset N$ is the union of all players on all directed paths from top player t to its complete subordinate i . Then we have the next lemma.

Lemma 3.2 *Let (N, D) be an acyclic permission structure, $t \in T_D$ a top player, and $i \in \overline{S}_D(t)$ a complete subordinate of t . Then $U_i \subseteq U^t$.*

Proof. By definition we have that $t \in U^t$. Since t is a complete superior of i , it follows that t is a complete superior of any $h \in U_i \setminus \{t\}$. Suppose not. Then for some $h \in U_i \setminus \{t\}$, there is a path from a top player $t' \neq t$ to h , and so also a path from t' to i , contradicting that t is a complete superior of i . By Lemma 3.1 we have that any h in U_i is a complete subordinate of t , and thus $h \in U^t$. \square

Since $U_i \subseteq U^t$ for all $i \in U^t = \overline{S}_D(t) \cup \{t\}$ (see Lemma 3.2) it follows that

$$U^t = \begin{cases} \{t\} & \text{if } \overline{S}_D(t) = \emptyset, \\ \cup_{\{i \in \overline{S}_D(t)\}} U_i & \text{otherwise.} \end{cases}$$

So, U^t is the union of all sets U_i of the complete subordinates of top player t when t has at least one complete subordinate, and $U^t = \{t\}$ otherwise.

The next proposition shows the existence of a subset $K \subseteq N$ such that the subgraph $(K, D(K))$ is an acyclic, quasi-strongly connected permission structure.

Proposition 3.3 *Let (N, D) be an acyclic permission structure. Then for every $t \in T_D$ it holds that the subgraph $(U^t, D(U^t))$ is an acyclic, quasi-strongly connected permission structure with t its unique top node.*

Proof. First, when $\bar{S}_D(t) = \emptyset$, then $U^t = \{t\}$ and the statement is true. Otherwise, let i be a complete subordinate of t . Obviously t is the unique top-player in the subgraph $(U_i, D(U_i))$. Further, the subgraph $(U_i, D(U_i))$ is acyclic and quasi-strongly connected. Acyclicity follows from the acyclicity of D and the fact that $D(U_i) \subseteq D$. Quasi-strongly connectedness follows from acyclicity of D and the fact that U_i is the union of all directed paths from t to i . Then the result follows because $U^t = \cup_{i \in \bar{S}_D(t)} U_i$. \square

The next lemma states that when N is partitioned in two disjunctive feasible sets, for both sets it holds that the total payoff of its players in every core payoff vector of the restricted game is equal to their own value.

Lemma 3.4 *Let $A, B \in \Phi^D$ be two disjunctive feasible coalitions in a non-negative additive game with acyclic permission structure (N, v, D) such that $A \cap B = \emptyset$ and $A \cup B = N$. Then $x(A) = \sum_{i \in A} a_i$ and $x(B) = \sum_{i \in B} a_i$ for every core element $x \in C(N, r)$.*

Proof. By definition of the restricted game (N, r) and feasibility of A we have that $r(A) = v(\sigma(A)) = v(A) = \sum_{i \in A} a_i$ and $r(B) = v(\sigma(B)) = v(B) = \sum_{i \in B} a_i$. Now, let $x \in C(N, r)$. Then $x(A) \geq r(A)$ and $x(B) \geq r(B)$. From the second inequality we obtain that $x(A) = r(N) - x(B) \leq \sum_{i \in N} a_i - \sum_{i \in B} a_i = \sum_{i \in A} a_i = r(A)$. Hence $x(A) = r(A) = \sum_{i \in A} a_i$. Analogous $x(B) = r(B) = \sum_{i \in B} a_i$. \square

We now state the final result of this section.

Proposition 3.5 *Let (N, D) be an acyclic permission structure and $t \in T_D$. Then $x(U^t) = \sum_{i \in U^t} a_i$ and $x(N \setminus U^t) = \sum_{i \in N \setminus U^t} a_i$ for every core element $x \in C(N, r)$.*

Proof. By definition we have that U^t is disjunctive feasible. To show that also $N \setminus U^t$ is disjunctive feasible, consider a player $i \in N \setminus U^t$. Since $i \neq t$ and i is also not a complete subordinate of t , there is a path from $T_D \setminus \{t\}$ to i . Hence $N \setminus U^t$ is disjunctive feasible. So, both U^t and $N \setminus U^t$ are disjunctive feasible and thus the proposition follows by applying Lemma 3.4 to $A = U^t$ and $B = N \setminus U^t$. \square

From the Propositions 3.3 and 3.5 it follows that for any $t \in T_D$ it holds that $U^t \subset N$ satisfies both properties that $(U^t, D(U^t))$ is an acyclic, quasi-strongly connected permission structure and $\sum_{j \in U^t} x_j = r(U^t) = \sum_{j \in U^t} a_j$ for every $x \in C(N, r)$, so also for the nucleolus. This property is now used to compute the nucleolus of the restricted game (N, r) obtained from a non-negative additive game with acyclic permission structure (that is not necessarily quasi-strongly connected) in a finite number of steps. At each step we compute the nucleolus of a smaller additive game with an acyclic, quasi-strongly connected permission structure by applying the $\mathcal{O}(n^4)$ algorithm of van den Brink *et al.* (2008) for

such games. This algorithm is an adaptation of the algorithm of Arin and Feltkamp (1997), which computes the nucleolus of veto-rich games in exponential time. Note that permission games with an acyclic, quasi-strongly connected permission structure are indeed veto-rich games, which is not true for the iver the algorithm, we introduce one more notation. Let (N, D) be an acyclic permission structure, $t \in T_D$ be one of the top players and $K = N \setminus U^t$. Then the set $D^K \in \mathcal{D}^K$ on the set of players K is given by

$$(i, j) \in D^K \text{ if and only if } (i, j) \in D \text{ and } P_D(j) \cap U^t = \emptyset. \quad (3.2)$$

So, for two players $i, j \in K$, (i, j) is an arc in D^K if and only if (i, j) is an arc in D and j does not have a predecessor in U^t . Stated differently, D^K contains all arcs in $D(K)$, except the arcs (i, j) such that j is a successor of top player t or of one of its complete subordinates. Finally, we assume that the players are enumerated (labeled) by the numbers $1, 2, \dots, n$ in such a way that for any $i, j \in \{1, \dots, n\}$ it holds that $i < j$ if $(i, j) \in D$. From the theory on acyclic directed graphs it is well-known that such a enumeration exists. Observe that this assumption implies that node $1 \in T_D$.

Algorithm

Step 1 Set $k = 1$, $N_1 = N$, $D_1 = D$ and $t_1 = 1$. Goto Step 2.

Step 2 Consider the non-negative additive game with acyclic, quasi-strongly connected permission structure $(U^{t_k}, v_k, D_k(U^{t_k}))$ with

$$v_k(U) = v(U) \text{ for all } U \subseteq U^{t_k}. \quad (3.3)$$

Let r_k be the restricted game of $(U^{t_k}, v_k, D_k(U^{t_k}))$. Goto Step 3.

Step 3 Apply the (polynomial time) algorithm of van den Brink *et al.* (2008) (see the appendix) to find the nucleolus of the restricted game (U^{t_k}, r_k) . Assign $y_i = Nuc_i(U^{t_k}, r_k)$ to every $i \in U^{t_k}$. Goto Step 4.

Step 4 If $U^{t_k} = N_k$ then Stop. Otherwise, goto Step 5.

Step 5 Define $N_{k+1} = N_k \setminus U^{t_k}$ and $D_{k+1} \in \mathcal{D}^{N_{k+1}}$ by $D_{k+1} = D_k^{N_{k+1}}$ (i.e., $D_{k+1} = D^K$ as defined in formula (3.2) with $D = D_k$ and $K = N_k \setminus U^{t_k}$). Define $t_{k+1} \in T_{D_{k+1}}$ as the top player in D_{k+1} with the lowest label ($t_{k+1} \leq h$ for every $h \in T_{D_{k+1}}$). Consider the set $U^{t_{k+1}}$ consisting of t_{k+1} and all its complete subordinates in the graph (N_{k+1}, D_{k+1}) . Set $k = k + 1$ and return to step 2.

4 The algorithm works

In this section we prove that the algorithm indeed finds the nucleolus of the original non-negative additive game with acyclic permission structure (N, v, D) . As a first observation, according to Proposition 3.5 it must hold that $\sum_{i \in U^1} Nuc_i(N, r) = \sum_{i \in U^1} a_i$, so the total payoff $\sum_{i \in U^1} a_i$ assigned in the first iteration of the algorithm to the players in U^1 is indeed equal to the total payoff that the players in U^1 attain at the $Nuc(N, r)$. Of course, we still have to prove that the individual payoffs assigned at the first iteration are the individual payoffs for the players in U^1 in $Nuc(N, r)$, and subsequently for the payoffs assigned at any next iteration. This will be proved by using the Davis-Maschler reduced game property. For a game (N, v) , let $T \subset N$ be a nonempty coalition and $x \in \mathbb{R}^n$ a payoff vector. Then the Davis-Maschler reduced game on T at x is the game (T, v_T^x) given by $v_T^x(T) = v(N) - x(N \setminus T)$ and $v_T^x(S) = \max_{Q \subseteq N \setminus T} (v(S \cup Q) - x(Q))$, $S \subset T$, $S \neq T$. Observe that in the definition of the reduced game only the values x_j of the players $j \in N \setminus T$ appear.

Property 4.1 Davis-Maschler reduced game property

For a game (N, v) , let x be the Nucleolus of (N, v) . Then for any $T \subset N$ it holds that

$$Nuc_i(N, v) = Nuc_i(T, v_T^x), \text{ for all } i \in T.$$

In the sequel we will denote the characteristic function of the Davis-Maschler reduced game with respect to the nucleolus $x = Nuc(N, v)$ and coalition $T \subset N$ just by v' , if there is no confusion⁶. Recall that for game $v \in \mathcal{G}^N$ and coalition $T \subset N$, the *subgame* $v_T \in \mathcal{G}^T$ is given by $v_T(S) = v(S)$ for all $S \subseteq T$. We now have the following proposition with respect to U^t for some $t \in T_D$. For notational simplicity, in the following we denote $U^t = K$.

Proposition 4.2 *For a non-negative additive game with acyclic permission structure (N, v, D) with $|T_D| \geq 2$, let $t \in T_D$ be a top player and $K = U^t$. Then the Davis-Maschler reduced game (K, r') of the restricted game (N, r) with respect to the nucleolus $x \in \mathbb{R}^n$ and the set K , coincides with the subgame (K, r_K) of (N, r) .*

Proof. First, observe that K is feasible and thus $r_K(K) = \sum_{i \in K} a_i$. On the other hand $r'(K) = r(N) - x(N \setminus K) = \sum_{i \in N} a_i - \sum_{i \in N \setminus K} a_i = \sum_{i \in K} a_i$, where the first equality is by definition of the reduced game and the second equality by Proposition 3.5. So, $r_K(K) = r'(K) = \sum_{i \in K} a_i$.

⁶In general, the Davis-Maschler reduced game property is stated for an arbitrary solution. Since we apply it here to compute the nucleolus, we only state it in terms of this particular solution.

Next, consider set $U \subset K$. By definition of the Davis-Maschler reduced game we have

$$r'(U) = \max_{S \subseteq N \setminus K} (r(U \cup S) - x(S)). \quad (4.4)$$

We first show that $r'(U) = r'(\sigma(U))$. Therefore it is sufficient to show that for any $S \subseteq N \setminus K$ it holds that $\sigma(U \cup S) = \sigma(\sigma(U) \cup S)$, because for any $S \subseteq N \setminus K$ the value of $r(U \cup S)$ in equation (4.4) is equal to the worth of $\sigma(U \cup S)$ in game v . Since $\sigma(U) \subseteq U$, it is evident that $\sigma(\sigma(U) \cup S) \subseteq \sigma(U \cup S)$. Suppose that the inclusion does not hold the other way around, i.e., there exists some player $i \in \sigma(U \cup S) \setminus \sigma(\sigma(U) \cup S)$. For this player there is a path p from T_D to i consisting of players in $\sigma(U \cup S)$ only, and there does not exist a path from T_D to i consisting of players in $\sigma(\sigma(U) \cup S)$ only. Consider a path $p = (p_1, \dots, p_m)$ from T_D to i with $H(p) \subseteq \sigma(U \cup S)$. We distinguish the following two cases:

- (i). Suppose that p is a directed path from t to i . Since $P_D(j) \subset K$ for all $j \in K$, there is a $k \in \{2, \dots, m\}$ such that $\{p_1, \dots, p_k\} \subseteq U$ and $\{p_{k+1}, \dots, p_m\} \subseteq S$. Since (p_1, \dots, p_k) is a directed path in U , we have that $\{p_1, \dots, p_k\} \subseteq \sigma(U)$. Thus, $H(p) \subseteq \sigma(U) \cup S$. But then $H(p) \subseteq \sigma(\sigma(U) \cup S)$ since p is a directed path with $p_1 = t$.
- (ii). Suppose that p is a directed path from another top-player $t' \neq t$ to i . Since players in $K = U^t$ do not have predecessors in $N \setminus K$, we have that $U \cap H(p) = \emptyset$. Thus, $H(p) \subseteq \sigma(S)$, and so $H(p) \subseteq \sigma(\sigma(U) \cup S)$.

From (i) and (ii) we conclude that $\sigma(U \cup S) = \sigma(\sigma(U) \cup S)$. Hence we have that $r'(U) = r'(\sigma(U))$ and it remains to prove that $r_K(U) = r'(U)$ for any feasible $U \subset K$.

Observe that $r_K(U) = \sum_{i \in U} a_i$ when U is feasible. To find $r'(U)$, we first show that for finding the maximum in (4.4), it is sufficient to consider only sets S such that $U \cup S$ is feasible. If $U \cup S$ is not feasible, then

$$\begin{aligned} r(U \cup S) - x(S) &= r(\sigma(U \cup S)) - x(S) \leq \\ r(U \cup (\sigma(U \cup S) \setminus U)) - x(\sigma(U \cup S) \setminus U) - x(S \setminus (\sigma(U \cup S) \setminus U)) &\leq \\ r(U \cup (\sigma(U \cup S) \setminus U)) - x(\sigma(U \cup S) \setminus U), \end{aligned}$$

where the first inequality follows from $\sigma(U \cup S) \setminus U \subseteq S$ and $\sigma(U \cup S) \setminus U \neq S$ if $U \cup S$ is not feasible. So, in case that $U \cup S$ is not feasible, replacing set S by set $\sigma(U \cup S) \setminus U$ does not decrease $r(U \cup S) - x(S)$ in formula (4.4), and thus this expression is maximized by a coalition S such that $U \cup S$ is feasible.

By definition of r' , it now follows that there is some $S \subseteq N \setminus K$ such that $U \cup S$ is feasible and

$$r'(U) = r(U \cup S) - x(S) = \sum_{i \in U \cup S} a_i - x(S). \quad (4.5)$$

Further, since $U \cup S$ is feasible and $U \subset K = U^t$, we have that also $K \cup S$ is feasible. So, by the fact that $x \in C(N, r)$, we have that $x(K \cup S) = x(K) + x(S) \geq \sum_{i \in K \cup S} a_i$. By Proposition 3.5 we have that $x(K) = a(K)$ and thus $x(S) \geq \sum_{i \in S} a_i$. It follows with equation (4.5) that

$$r'(U) = \sum_{i \in U} a_i + \sum_{i \in S} a_i - x(S) \leq \sum_{i \in U} a_i = r(U). \quad (4.6)$$

From (4.6) and the fact that by definition of the reduced game $r'(U) \geq r(U)$, we conclude that

$$r'(U) = r(U) = \sum_{i \in U} a_i.$$

This proves that $r'(U) = r_K(U)$ for all $U \subset K = U^t$. \square

In the first iteration the algorithm finds the nucleolus of (U^1, r_1) . Clearly, the restricted game (U^1, r_1) of (U^1, v_1, D_1) is equal to the subgame (U^1, r_{U^1}) of (N, r) , which is equal to the Davis-Maschler reduced game according to the proposition above. So, with the Davis-Maschler reduced game Property 4.1, the proposition above shows that in the first iteration the algorithm indeed computes the nucleolus payoffs of the players in U^1 in game (N, r) . For $t \in T_D$ and $K = N \setminus U^t$, the next proposition shows that the Davis-Maschler reduced game with respect to the nucleolus and the set K coincides with the game (K, r_2) , where r_2 is the restricted game of the non-negative game with the reduced acyclic permission structure (K, v_K, D^K) on the set of players K , where (K, v_K) is the subgame of (N, v) on K and D^K is as given in formula (3.2) for $t = 1$, i.e., r_2 is the restricted game used in the second iteration of the algorithm.

Proposition 4.3 *For a non-negative additive game with acyclic permission structure (N, v, D) with $|T_D| \geq 2$, let $t \in T_D$ and $K = N \setminus U^t$. Then the Davis-Maschler reduced game (K, r') of the restricted game (N, r) with respect to the nucleolus payoff vector $x \in \mathbb{R}^n$ and the set K , coincides with the restricted game (K, r_2) of the non-negative additive game with acyclic permission structure (K, v_K, D^K) with D^K as defined in formula (3.2).*

Proof. By definition, $P_{D^K}(j) \subseteq K$ for all $j \in K$, so K is feasible in the reduced graph (K, D^K) . Hence $r_2(K) = \sum_{i \in K} a_i$. On the other hand, $r'(K) = r(N) - x(N \setminus K) = r(N) - x(U^t) = \sum_{i \in N} a_i - \sum_{i \in U^t} a_i = \sum_{i \in K} a_i$, where the first equality is by definition of the reduced game and the second equality by Proposition 3.5. So, $r_2(K) = r'(K)$.

Next, for a set $S \subset K$ we consider two cases.

(i). First, suppose that $P_D(S) \cap U^t = \emptyset$. For all $U \subseteq U^t$ it holds that $\sigma(U \cup S) = \sigma(U) \cup \sigma(S)$, since $P_D(U) \cap S = \emptyset$. So,

$$r(S \cup U) - x(U) = v(\sigma(U) \cup \sigma(S)) - x(U) =$$

$$v(\sigma(U)) + v(\sigma(S)) - x(U) = r(S) + r(U) - x(U) \leq r(S),$$

where the second equality follows from v being an additive game and the (last) inequality follows from the nucleolus being a core element. Hence

$$r'(S) = \max_{U \subseteq U^t} (r(S \cup U) - x(U)) = r(S).$$

On the other hand, for all $i, j \in S$ we have that $(i, j) \in D^K$ if $(i, j) \in D$, because no player $j \in S$ has a predecessor in U^t . Hence, $\sigma(S) = \sigma_K(S)$, where $\sigma_K(S)$ is the largest feasible subset of S in (K, D_K) . So,

$$r'(S) = r(S) = v(\sigma(S)) = v(\sigma_K(S)) = r_2(S).$$

(ii). Secondly we consider the case that $P_D(S) \cap U^t \neq \emptyset$. Now, let $S' \subseteq S$ be given by $S' = \sigma(U^t \cup S) \setminus U^t$. By definition of σ , $U^t \cup S'$ is the maximal feasible subset of $U^t \cup S$ and thus

$$r(U^t \cup S) - x(U^t) = v(U^t \cup S') - x(U^t) = \sum_{i \in U^t \cup S'} a_i - \sum_{i \in U^t} a_i = \sum_{i \in S'} a_i,$$

where the second equality follows by Proposition 3.5. Hence we have that

$$r'(S) = \max_{U \subseteq U^t} (r(S \cup U) - x(U)) \geq r(U^t \cup S) - x(U^t) = \sum_{i \in S'} a_i. \quad (4.7)$$

We now show that this holds with equality. First, recall from the proof of Proposition 4.2 that $\sigma(U \cup S) = \sigma(\sigma(U) \cup S)$ for every $U \subset U^t$ and $S \subset K$. Hence

$$r(U \cup S) - x(U) = r(\sigma(U) \cup S) - x(U) \leq r(\sigma(U) \cup S) - x(\sigma(U)),$$

so that $\max_{U \subseteq U^t} (r(S \cup U) - x(U))$ will be obtained by a feasible set U . Suppose there exists a feasible $U \subseteq U^t$ with $r(U \cup S) - x(U) > \sum_{i \in S'} a_i$. Then, with $S_U \subseteq S$ given by $S_U = \sigma(U \cup S) \setminus U$, it follows in an analogous way as for $U = U^t$ above, that $\sigma(U \cup S) = U \cup S_U$, and thus

$$\begin{aligned} r(U \cup S) - x(U) &= v(U \cup S_U) - x(U) = \sum_{i \in U \cup S_U} a_i - x(U) = \\ &= \sum_{i \in S_U} a_i + \sum_{i \in U} a_i - x(U) \leq \sum_{i \in S_U} a_i \leq \sum_{i \in S'} a_i, \end{aligned} \quad (4.8)$$

where the first inequality follows because U is feasible and the nucleolus lies in the core, and the second inequality because $U \subseteq U^t$ and thus $S_U = \sigma(U \cup S) \setminus U \subseteq \sigma(U^t \cup S) \setminus U^t$. From equations (4.7) and (4.8) it follows that

$$r'(S) = \sum_{i \in S'} a_i = v(S').$$

It remains to prove that also $r_2(S) = v(S')$. By definition we have that $r_2(S) = v_K(\sigma_K(S)) = v(\sigma_K(S))$, where $\sigma_K(S)$ is the maximal feasible subset of S in graph (K, D^K) . So, it remains to show that $\sigma_K(S) = S'$.

We first show that $\sigma_K(S) \subseteq S' = \sigma(U^t \cup S) \setminus U^t$. Consider $i \in \sigma_K(S)$. For such a player i there is a path p from T_{D^K} to i which only contains elements of S . We consider two cases. If p is a path from a top $t' \in T_D$, then $i \in \sigma(S) \subseteq S'$. For the case that p is a path from a top t' in $T_{D^K} \setminus T_D$, then t' is a top in (K, D^K) but not in (N, D) and thus $P^D(t') \cap U^t \neq \emptyset$, implying that in (N, D) there is a path p' from t to t' . Hence, the path p'' consisting of the path p' from t to t' and the path p from t' to i is a path in (N, D) from T_D to i . Since $H(p') \setminus \{t'\} \subseteq U^t$ and $H(p) \subseteq S$, we have that the set of nodes $H(p'') \subseteq U^t \cup S$ and thus $H(p'') \subseteq \sigma(U^t \cup S)$. So $i \in \sigma(U^t \cup S)$ and we can conclude that $i \in \sigma(U^t \cup S) \setminus U^t = S'$.

Next we show the reverse that $S' \subseteq \sigma_K(S)$. Let $i \in S'$. Then $i \in \sigma(U^t \cup S)$ and thus there is path p from T_D to i that consists of elements of $U^t \cup S$. Again there are two cases. If $H(p) \subseteq S$, then $i \in \sigma(S) \subseteq \sigma_K(S)$. Otherwise, p consists of two subpaths p' in U^t and p'' in S that are connected to each other by a link from the last node of p' to the first node of p'' . Let j be that last node of path p'' that has a predecessor in U^t . Then, by construction, j is a top-node in game (K, D^K) and thus the part of p'' from j to i is a path in (K, D^K) from T_{D^K} to i . So $i \in \sigma_K(S)$. \square

By repeated application of the propositions above it follows that the algorithm of Section 3 computes the nucleolus of (N, r) .

Proposition 4.4 *For a non-negative additive game with acyclic permission structure (N, v, D) , the algorithm described in Section 4 finds the nucleolus of (N, r) within a finite number of iterations.*

Proof. In the first iteration the algorithm finds the nucleolus of the restricted game (U^1, r_1) of (U^1, v_1, D_1) , which is equal to the subgame (U^1, r_{U^1}) of (N, r) . By Proposition 4.2 and the Davis-Maschler reduced game property it follows that $Nuc_i(U^1, r_1) = Nuc_i(N, r)$ for all $i \in U^1$. If $U^1 = N$, the algorithm ends with the nucleolus in one iteration. Otherwise, the algorithm continues in iteration 2 with the restricted game $(N \setminus U^1, r_2)$ of $(N \setminus U^1, v, D_2)$. By Proposition 4.3 this restricted game is equal to the DM reduced game on $N \setminus U^1$ with respect to the nucleolus and so the nucleolus payoffs of the game $(N \setminus U^1, r_2)$ are equal to the nucleolus payoffs of the players in $N \setminus U^1$ in the game (N, r) . Repeating the arguments of the first iteration, in the second iteration the algorithm computes the nucleolus payoffs of the game $(N \setminus U^1, r_2)$ for the players $i \in U^{t_2}$, which thus is equal to their nucleolus payoffs in the game (N, r) . Subsequently in each iteration k the algorithm computes the nucleolus payoffs of the set of players in U^{t_k} . Since the number of players is reduced

with at least one in each iteration of the algorithm, the algorithm ends within at most n iterations. \square

5 Properties of the algorithm

In this section we first show several properties of the algorithm. From these properties we then obtain an interesting property of the nucleolus of a non-negative additive game with permission structure, namely that each coalition consisting of a *free player* and its complete subordinates distributes its own value among themselves. A *free player* is a player that does not have a complete superior.

Definition 5.1 *A player $i \in N$ in an acyclic permission structure (N, D) is a **free player** if and only if $\overline{P}_D(i) = \emptyset$.*

Example 5.2 Consider permission structure (N, D) with $N = \{1, 2, 3, 4, 5, 6\}$ and $D = \{(1, 3), (1, 4), (2, 4), (3, 5), (3, 6), (4, 6)\}$. This permission structure has four free players. Two of them are the two top players 1 and 2. Besides the top players the two other free players are 4 and 6.

In the sequel we denote the set of free players in (N, D) by F_D . Notice that $T_D \subseteq F_D$. Recall from Subsection 2.2 that for every non-top player i , the set \mathcal{P}_i denotes the collection of all paths from T_D to i . For a path $p = (i_1, \dots, i_{m-1}, i_m) \in \mathcal{P}_i$ (so $i_1 \in T_D$ and $i_m = i$), in the sequel $H_i(p)$ denotes the set of players $H(p) \setminus \{i\}$, i.e., $H_i(p)$ is the set of all players on the path p except player i itself. We now give two lemmas. The first one is obvious and stated without proof.

Lemma 5.3 *For every acyclic permission structure (N, D) , if j is a (complete) superior of i and k is a (complete) superior of j then k is a (complete) superior of i .*

The second lemma states that for any free player $i \in F_D \setminus T_D$ there exist (at least) two paths p and q in \mathcal{P}_i such that $H_i(p) \cap H_i(q) = \emptyset$, i.e., for any non-top free player i there exist two disjunct paths (except for i itself) from T_D to i . Recall that it is assumed that the players are enumerated (labeled) by the numbers $1, 2, \dots, n$ in such a way that for any $i, j \in \{1, \dots, n\}$ it holds that $i < j$ if $(i, j) \in D$.

Lemma 5.4 *Let (N, D) be an acyclic permission structure and let i be a player in $F_D \setminus T_D$. Then there exist (at least) two paths p and q in \mathcal{P}_i , such that $H_i(p) \cap H_i(q) = \emptyset$.*

Proof. For any two different paths p^h and p^k in \mathcal{P}_i , define $m_{hk} = \max\{j | j \in H_i(p^k) \cap H_i(p^h)\}$ with the convention that $m_{hk} = 0$ if $H_i(p^k) \cap H_i(p^h) = \emptyset$, i.e., m_{hk} is the highest labeled player that is on both paths. Further, define $m = \min_{h,k} m_{hk}$. Suppose $m \geq 1$, i.e., there exist two paths, say p^1 and p^2 with $H_i(p^1) \cap H_i(p^2) \neq \emptyset$, such that (i) m is the highest labeled common node of p^1 and p^2 and (ii) $m_{hk} \geq m$ for any two paths p^h and p^k in \mathcal{P}_i . Since i is a free player, and thus $\overline{P}_D(i) = \emptyset$, there is no player $j \neq i$ that is on all paths in \mathcal{P}_i . Therefore we must have a third path, say $p^3 \in \mathcal{P}_i$, such that $m \notin H_i(p^3)$. Because of (ii) we have that $m_{13} > m$ and $m_{23} > m$, so for both paths p^1 and p^2 it holds that they have a node in common with p^3 with a higher label than m . For $j = 1, 2$, define $m_j = \min\{s > m | s \in H_i(p^j) \cap H_i(p^3)\}$, so m_j is the lowest labeled common node on the paths p^j and p^3 higher than m . Thus $m_1 > m$ and $m_2 > m$ and also $m_1 \neq m_2$, otherwise $m_1 = m_2 \in H_i(p^1) \cap H_i(p^2)$, contradicting (i). Without loss of generality, suppose that $m_1 > m_2$. Now, let p^4 be the path in \mathcal{P}_i that is equal to p^3 from T_D to node m_2 and it is equal to p^2 from m_2 to i . Then $m_{14} < m$, because p^4 coincides with p^2 from node m_2 to i and the highest labeled common node of p^1 and p^2 is node $m < m_2$, and p^4 coincides with p^3 from T_D to m_2 and the smallest labeled common node of p^1 and p^3 higher or equal to m is node $m_1 > m_2$. However, $m_{14} < m$ contradicts (ii). So it follows that $m = 0$, which proves that there exists two paths in \mathcal{P}_i that only have node i in common. \square

Recall that in the first iteration player 1 is chosen to be the top and that the nucleolus payoffs of the players in U^1 are computed. In the second iteration the algorithm continues with the non-negative additive game with permission structure (K, v_K, D^K) , where $K = N \setminus U^1$, (K, v_K) the subgame of (N, v) on K , and D^K the permission structure as obtained in formula (3.2). The next lemma states that $j \neq 1$ is a free player in (K, D^K) if and only if it is a free player in (N, D) .

Lemma 5.5 *Let (N, D) be an acyclic permission structure, $K = N \setminus U^1$ and D^K the permission structure as defined in formula (3.2). Then $F_{D^K} = F_D \setminus \{1\}$.*

Proof. First, observe that U^1 consists of player 1 and all its complete subordinates, so by Lemma 3.1, U^1 does not contain any of the players in $F_D \setminus \{1\}$, and thus $F_D \setminus \{1\} \subseteq K$. We first prove that $F_D \setminus \{1\} \subseteq F_{D^K}$. When $i \in T_D \setminus \{1\}$, then obviously $i \in T_{D^K}$. So, consider $i \in F_D \setminus T_D$. According to Lemma 5.4, there are two different paths in (N, D) from T_D to i . Let p be such a path. We consider two cases.

1. The path p is completely in K and $P_D(j) \cap U^1 = \emptyset$ for any $j \in H(p)$. Then any link $(h, k) \in D$ on the path is also a link in D_K and $p' = p$ is a path in (K, D_K) from $T_D \setminus \{1\} \subseteq T_{D^K}$ to i .

2. There is some $h \in H(p)$ such that $P_D(h) \cap U^1 \neq \emptyset$. With $A(p)$ the set of all players in $H(p)$ having a predecessor in U^1 , let k be the player in $A(p)$ such that $k \geq h$ for

all $h \in A(p)$, i.e., k is the player in $A(p)$ with the highest label. Then, by formula (3.2), any link $(i, k) \in D(K)$ is deleted to obtain D^K , so $k \in T_{D^K}$. Let p' be the path consisting of the part of path p from k to i . Then p' is a path from T_{D^K} to i in $(K, D(K))$. Observe that $p' = (i)$ with $i \in T_{D^K}$ if $k = i$.

It follows that every path p from T_D to i in (N, D) gives a path p' from T_{D^K} to i in $(K, D(K))$. When for some p the path p' reduces to the single player path $p' = (i)$ (when $k = i$ in case 2), then i becomes a top player in $(K, D(K))$ and every path from T_D to i reduces to the single element i . Otherwise, i has two different paths in $(K, D(K))$ when i has two different paths in (N, D) , because $H(p') \subseteq H(p)$ for every path p from T_D to i . So $i \in F_{D^K}$ when $i \in F_D \setminus \{1\}$.

Second, we prove the reverse inclusion that $F_{D^K} \subseteq F_D \setminus \{1\}$. To do so we show that a node $i \neq 1$ which is not free in (N, D) is also not free in (K, D_K) . Let $i \notin F_D \setminus \{1\}$, so i has a complete superior in (N, D) . When there is a complete superior in U^1 then, by Lemma 5.3, also 1 is a complete superior of i . Then $i \in U^1$ and thus i is not in K . It remains to consider the case that $\overline{P}_D(i) \subset K$. Let k be a player in $\overline{P}_D(i)$. For p a path from T_D to i , let p' be the part of the path from k to i . Then there is no player $h > k$ on the path p' that has a predecessor in U^1 , otherwise there is in (N, D) a path from 1 to the predecessor of h in U^1 , then to h and then to i , contradicting that k is a complete superior of i in the graph (N, D) . So, when $(j, l) \in D$ is a link on p' , then also $(j, l) \in D_K$ and thus p' is a path in (K, D_K) . Since this holds for every path p from T_D to i in (N, D) , it follows that k is also a complete superior of i in (K, D_K) , and thus i is not free in (K, D_K) . \square

The next lemma states that a player is chosen as top in one of the iterations of the algorithm if and only if it is a free player in (N, D) . Let A_D denote the set of players that is chosen as top in one of the iterations of the algorithm of Section 3.

Lemma 5.6 *Let (N, v, D) be a non-negative additive game with acyclic permission structure. Then $A_D = F_D$.*

Proof. The proof follows by repeated application of Lemma 5.5. In iteration 1, player 1 is chosen as top. So, player 1 belongs both to A_D and F_D . In iteration 2, player t_2 is chosen as top. This player is determined in Step 5 of the previous iteration as the top player with the lowest label in (N_2, D_2) with $N_2 = N_1 \setminus U^1 = N \setminus U^1$ and $D_2 = D^{N_2}$. By Lemma 5.5 this top belongs to $F_{D_2} = F_D \setminus \{1\}$. From repeated application of the lemma it follows that the top t_k in iteration k is a top of $F_{D_k} = F_{D_{k-1}} \setminus \{t_{k-1}\} = F_D \setminus \{t_1, \dots, t_{k-1}\}$, where (N_k, D_k) is the graph at iteration k . It follows that succeedingly all players of F_D are chosen as top in increasing order of their label. \square

Finally we show that the set of complete subordinates of the chosen top t_k in the graph (N_k, D_k) in iteration k is equal to the set of complete subordinates of the free player

t_k in (N, D) . Moreover we have that $(U^{t_k}, D_k(U^{t_k})) = (U^{t_k}, D(U^{t_k}))$, i.e., the subgraph on U^{t_k} of (N_k, D_k) is equal to the subgraph of U^{t_k} of (N, D) .

Lemma 5.7 *Let (N, v, D) be a non-negative additive game with acyclic permission structure and let t_k be the chosen top in iteration k of the algorithm. Then $\overline{S}_{D_k}(t_k) = \overline{S}_D(t_k)$. Moreover, $(U^{t_k}, D_k(U^{t_k})) = (U^{t_k}, D(U^{t_k}))$.*

Proof. From the proof of Lemma 5.5 it follows that when $k \in N_2 = N_1 \setminus U^1$ is a complete superior of i in $(N_1, D_1) = (N, D)$, then k is also a complete superior of i in (N_2, D_2) . With Lemma 3.1 it follows reversely that $i \in \overline{S}_{D_2}(k)$ when $i \in \overline{S}_{D_1}(k)$. On the other hand, when i is not a complete subordinate of k in (N_1, D_1) , then it also not in (N_2, D_2) because $D_2 \subset D_1$ and either $P_{D_2}(i) = \emptyset$ or there is a path from T_{D_2} to i in (N_2, D_2) without k . So, for any $k \in N_2$ we have that $\overline{S}_{D_2}(k) = \overline{S}_{D_1}(k)$. The first statement of the lemma follows by repeating these arguments for all the remaining top players at any iteration of the algorithm. To show the second statement, let $k \in N_2$ be a complete superior of a player $i \in N_2$. From the last part of the proof of Lemma 5.5 it follows that any link on a path from k to i in (N_1, D_1) is also a link in (N_2, D_2) . So, $(U^{t_2}, D_2(U^{t_2})) = (U^{t_2}, D_1(U^{t_2})) = (U^{t_2}, D(U^{t_2}))$. The result follows from repeating this at any next iteration of the algorithm. \square

We now come to the main result of this section. In Section 4 we have seen that the algorithm of Section 3 at any iteration k computes the nucleolus payoffs in the restricted game (N, r) of the players in U^{t_k} , where U^{t_k} is the set of players consisting of the chosen top t_k in iteration k and all its complete subordinates in (N_k, D_k) . Moreover, the total payoff of the players in U^{t_k} is equal to the sum of their values a_i , $i \in U^{t_k}$. Therefore the next proposition follows from the lemmas above without further proof. It states that for any free player the nucleolus distributes the total contributions of this player and its complete subordinates among themselves.

Proposition 5.8 *Let (N, v, D) be a non-negative additive game with acyclic permission structure. Then $\sum_{i \in \overline{S}_D(k)} Nuc_i(N, r) = \sum_{i \in \overline{S}_D(k)} a_i$ for every $k \in F_D$.*

Finally, when the structure of the graph (N, D) is known ex ante, in particular the set F_D is known ex ante and also for each $k \in F_D$ its set of complete subordinates, it follows without further proof from the lemmas above that the algorithm of Section 3 reduces to $|F_D|$ applications of the algorithm of van den Brink *et al.* (2008). To state this result, for $k \in F_D$, let \bar{r}_k be the restricted game of $(\overline{S}_D(k), v_k, D(\overline{S}_D(k)))$, where v_k is the non-negative additive game on $\overline{S}_D(k)$ given by $v_k(S) = \sum_{i \in S} a_i$, $S \subseteq \overline{S}_D(k)$. Then we have the following proposition.

Proposition 5.9 *Let (N, v, D) be a non-negative additive game with acyclic permission structure. Then $Nuc_i(N, r) = Nuc_i(\overline{S}_D(k), r_k)$, $i \in \overline{S}_D(k)$, $k \in F_D$.*

From this last proposition we can conclude that for any $k \in F_D$, the nucleolus values in the game (N, r) of the players in $\overline{S}_D(k)$ can be computed by applying the algorithm of van den Brink *et al.* (2008) to $(\overline{S}_D(k), v_k, D(\overline{S}_D(k)))$.

6 Complexity of the algorithm

When the structure of the game is known ex ante we can apply Proposition 5.9 and apply the algorithm of van den Brink *et al.* (2008) to any $(\overline{S}_D(k), v_k, D(\overline{S}_D(k)))$, $k \in F_D$. Since the complexity of this algorithm is of $\mathcal{O}(n^4)$, the complexity reduces to $\mathcal{O}(|F_D|a^4)$, where $a = \max_{k \in F_D} |\overline{S}_D(k)|$, because we have $|F_D|$ problems and the k th problem has complexity $\mathcal{O}(|\overline{S}_D(k)|^4)$. In particular, we have that the complexity is of $\mathcal{O}(n^4)$ when $|F_D| = 1$ (and thus $|\overline{S}_D(1)| = n$) and of $\mathcal{O}(n)$ when $|F_D| = n$ (and thus $|\overline{S}_D(k)| = 1$ for all k). Clearly, in the latter (extreme) case we have that (N, D) is the empty graph and (N, v, D) reduces to the additive game, so that every player i gets its own value a_i .

Typically in practice the structure is not known in advance. Also, although we assumed in the previous sections that the players are enumerated by the numbers $1, 2, \dots, n$ in such a way that for any $i, j \in \{1, \dots, n\}$ it holds that $i < j$ if $(i, j) \in D$, in practice such an enumeration will not be known in advance. So, to perform the algorithm at each iteration k first a top node t_k in (N_k, D_k) has to be found and its corresponding set U^{t_k} of complete subordinates in (N_k, D_k) . For the complexity of this search, let us consider the first iteration. It is evident that we can find the collection T_D of top nodes in at most n^2 actions, because we just have to consider each pair of players once. Then we enumerate the top nodes from 1 to k_1 , where $k_1 = |T_D|$ is the number of top nodes in (N, D) . To find the set U^1 of all complete subordinates of top node 1 we can proceed as follows. First, assign label 1 to every successor of node 1. Next, assign label 1 to every successor of every node with label 1 and continue to do this. So, every player in the set $\widehat{S}_D(1)$ of subordinates of top player 1 gets a label 1. Clearly, this requires at most $\mathcal{O}(n^2)$ actions. Next, repeat this procedure for every other top node, so for every player in the set $\widehat{S}_D(j)$ of top node j , $j = 1, \dots, k_1$. So, a node can receive multiple labels. The set U^1 consists of all nodes that only receive label 1 and can be found in at most $k_1 \mathcal{O}(n^2) < \mathcal{O}(n^3)$ actions. In fact this procedure gives in at most $\mathcal{O}(n^3)$ actions all top nodes $j \in T_D$ and their sets U^j of complete subordinates.

We can now use the result of Proposition 5.9 to modify the algorithm, namely instead of adapting the graph after the first iteration according to Step 5, we first apply k_1 times the algorithm of van den Brink *et al.* (2008), namely to subgame with permission

structure $(U^k, v_k, D(U^k))$ for every top player $k \in T_D$. The complexity of each application is given by $\mathcal{O}(|U^k|^4)$.

After having eliminated all top players and their complete subordinates we now adapt the graph on the remaining set of players $N \setminus (\cup_{k \in T_D} U^k)$ analogously as described in Step 5 of the algorithm. Let M be this set, then for each player $j \in M$ we consider whether or not j has a predecessor in (N, D) belonging to $\cup_{k \in T_D} U^k$. If so, all edges (h, j) in the subgraph $(M, D(M))$ are deleted and j becomes a top node in the remaining graph. Observe that such a top node in the new graph is a free player in (N, D) . This requires at most $\mathcal{O}(n^2)$ actions and yields also the set of top nodes in the remaining graph. Next, repeating the procedure as described above for the set of new top nodes, in $\mathcal{O}(m^3)$ actions, where $m = |M|$, the sets U^j of complete subordinates of the new top nodes can be found. Then we apply again the algorithm of van den Brink *et al.* (2008) to each new top node j and its set of complete subordinates U^j . After that we apply the procedure of finding the new graph and new set of top players for the remaining nodes and so on.

Summarizing, starting the algorithm the number of actions to find the set of top nodes and their sets of complete subordinates is of $\mathcal{O}(n^3)$. This has to be repeated at most $n-1$ times to find the set of all free players (each free player is a top node at some stage) and their sets of complete subordinates. So, the number of actions to find all free players and their sets of complete subordinates is (at most) of $\mathcal{O}(n^4)$. For each free player $k \in F_D$ the complexity of the algorithm of van den Brink *et al.* is of $\mathcal{O}(|U^k|^4)$. Since $\sum_{k=1}^s |U^k| = n$, where $s = |F_D|$, the total complexity of applying the algorithm s times, namely for each free player, is of $\mathcal{O}(n^4)$. So, both the complexity of finding all free players and their sets of complete subordinates and the total complexity to find the nucleolus payoffs for every free player with its set of complete subordinates, is given by $\mathcal{O}(n^4)$. Hence the total complexity of the algorithm of Section 3 to find all nucleolus payoffs is of $\mathcal{O}(n^4)$, showing that the algorithm finds the nucleolus in polynomial time.

7 An example

In this section we illustrate the computation of the nucleolus for non-negative additive disjunctive games with a permission structure by giving an example concerning a market situation where sellers can sell objects to buyers through a (directed) network of intermediaries. First, we give a simple example without intermediaries⁷.

Example 7.1 Consider a situation where there is one seller (player 1) and one buyer (player 2) who can realize a non-negative surplus $a > 0$ from trade. The corresponding

⁷This is a special case of the assignment game, introduced by Shapley and Shubik (1972).

assignment game on $N = \{1, 2\}$ is given by $v(\{1\}) = v(\{2\}) = 0$ and $v(\{1, 2\}) = a$. Note that the restricted game (N, r) on the permission structure $D = \{(1, 2)\}$ (or $D = \{(2, 1)\}$) is the same as (N, v) . Clearly, the nucleolus of this game yields an equal division of the surplus, i.e., $Nuc(N, v) = Nuc(N, r) = (\frac{a}{2}, \frac{a}{2})$. But as soon as there are two or more sellers (say players $1, \dots, n-1$), such that the buyer (player n) can realize the surplus a with any one of the sellers and cannot generate more surplus by trading with more sellers (for example, the sellers all own one item of a good for which they have reservation value zero, and the buyer wants only one item of the good and is prepared to pay at most a for it) then the characteristic function of the assignment game on $N = \{1, \dots, n\}$ is $v(S) = a$ if $n \in S$ and $|S| \geq 2$, and $v(S) = 0$ otherwise. Again the restricted game on the permission structure $D = \{(i, n) \mid i \in \{1, \dots, n-1\}\}$ (i.e., the permission structure where all sellers are predecessor of the buyer) is the same as (N, v) . Now, it is clear that $Nuc_n(N, v) = Nuc_n(N, r) = a$ and $Nuc_i(N, v) = Nuc_i(N, r) = 0$ for all $i \in \{1, \dots, n-1\}$ since this is the unique Core payoff vector. This also follows immediately from Proposition 5.8 by observing that r is the restricted game of the non-negative additive game with $a_i = 0$ for $i = 1, \dots, n_1$ and $a_n = a$ and that player n is a free player in (N, D) and thus receives its own value. So, similar as in a linear Bertrand price competition game, as soon as there is more than one seller, the surplus fully goes to the buyer. (A similar story holds if there is only one seller but more buyers.) \square

Next, we consider an example of a market situation as described above, but buyers and sellers may not be able to trade directly with each other, but need intermediaries to connect them.

Example 7.2 Consider a market situation with two sellers (players 1 and 2) and four buyers (players 7, 8, 9 and 10) who cannot trade directly with each other but need intermediaries. Consider the permission structure D on $N = \{1, \dots, 10\}$ given by $D = \{(1, 3), (1, 4), (2, 4), (2, 5), (2, 6), (3, 7), (4, 7), (4, 8), (5, 9), (6, 9), (6, 10)\}$, see Figure 1. For every buyer-seller pair that wants to make a deal, it is sufficient to use only one of the intermediaries they are both connected with. For example, for seller 1 and buyer 7 it is necessary and sufficient to use either intermediary 3 or intermediary 4, while seller 1 and buyer 8 need intermediary 4 to trade. Suppose that each seller owns at least four items of the good and each buyer wants one item. Buyer $i \in \{7, 8, 9, 10\}$ is prepared to pay $a_i > 0$ for the item. The reservation value of the sellers and all intermediaries is zero. This can be modelled as the game with permission structure (N, v, D) with $N = \{1, \dots, 10\}$, D as given above, and v the non-negative additive game with $a_i = 0$ for $i = 1, \dots, 6$ and $a_i > 0$ for $i = 7, \dots, 10$, so that $v(S) = \sum_{i \in S} a_i = \sum_{i \in S \cap \{7, 8, 9, 10\}} a_i$ for all $S \subseteq N$.

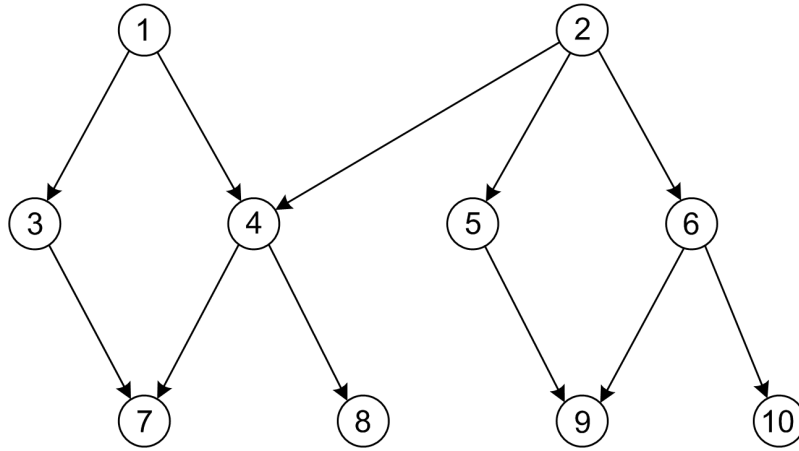


Figure 1.

The nucleolus of the restricted game (N, r) can be computed using the algorithm of Section 3 and the properties of Section 5. Notice that the players 1 and 2 are top players, and that the set of free players is given by $F_D = \{1, 2, 4, 7\}$. Using Proposition 5.9 it follows that we can find the nucleolus of (N, r) by considering the four subgames with permission structure given in Figure 2, namely one subgame for each of the four free players.

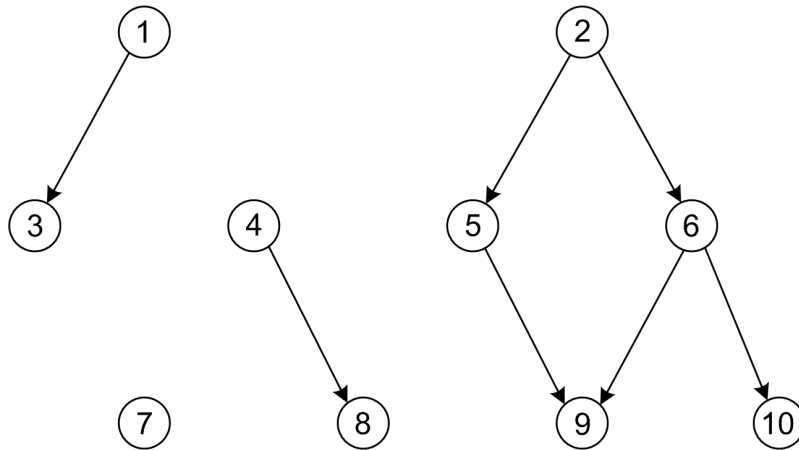


Figure 2

The nucleoli for the four subgames are as follows.

1. Consider the game with permission structure (N_1, v_1, D_1) given by $N_1 = \{1, 3\}$, $v_1 = v|_{N_1}$ and $D_1 = D|_{N_1} = \{(1, 3)\}$. Since v_1 is the null game assigning worth zero to all coalitions in N_1 , it follows that $Nuc_1(N_1, r_1) = Nuc_3(N_1, r_1) = 0$.
2. For the game with permission structure (N_2, v_2, D_2) given by $N_2 = \{2, 5, 6, 9, 10\}$, $v_2 = v|_{N_2}$, and $D_2 = D|_{N_2} = \{(2, 5), (2, 6), (5, 9), (6, 9), (6, 10)\}$, we have that $v_2(S) = \sum_{i \in S \cap \{9, 10\}} a_i$ for all $S \subseteq N$. Then the restricted game is given by

$$r_2(S) = \begin{cases} a_9 & \text{if } S \in \{\{2, 5, 9\}, \{2, 6, 9\}, \{2, 5, 6, 9\}\} \\ a_{10} & \text{if } S \in \{\{2, 6, 10\}, \{2, 5, 6, 10\}\} \\ a_9 + a_{10} & \text{if } S \in \{\{2, 6, 9, 10\}, \{2, 5, 6, 9, 10\}\} \\ 0 & \text{otherwise.} \end{cases}$$

Applying the algorithm of van den Brink *et al.* (2008) (see the Appendix) we find for any pair of values $a_9, a_{10} \geq 0$ that $Nuc_2(N_2, r_2) = \frac{a_9}{2} + \frac{a_{10}}{3}$, $Nuc_5(N_2, r_2) = 0$, $Nuc_6(N_2, r_2) = \frac{a_{10}}{3}$, $Nuc_9(N_2, r_2) = \frac{a_9}{2}$ and $Nuc_{10}(N_2, r_2) = \frac{a_{10}}{3}$.

3. For the game with permission structure (N_3, v_3, D_3) given by $N_3 = \{4, 8\}$, $v_3 = v|_{N_3}$ and $D_3 = D|_{N_3} = \{(4, 8)\}$, we have that $v_3(S) = \sum_{i \in S \cap \{8\}} a_i$ for all $S \subseteq N$. So, the restricted game is given by

$$r_3(S) = \begin{cases} a_8 & \text{if } S = \{4, 8\} \\ 0 & \text{otherwise.} \end{cases}$$

We find that $Nuc_4(N_3, r_3) = Nuc_8(N_3, r_3) = \frac{a_8}{2}$.

4. The game with permission structure (N_4, v_4, D_4) is given by $N_4 = \{7\}$, $v_4 = v|_{N_4}$ and $D_4 = \emptyset$. Clearly $v_4(S) = r_4(S) = a_7$ for $S = \{7\}$, and thus $Nuc_7(N_4, r_4) = a_7$.

By Proposition 5.9 we then have $Nuc(N, r) = (0, \frac{a_9}{2} + \frac{a_{10}}{3}, 0, \frac{a_8}{2}, 0, \frac{a_{10}}{3}, a_7, \frac{a_8}{2}, \frac{a_9}{2}, \frac{a_{10}}{3})$. \square

In the example above, there was only one level of intermediaries, but the algorithm also works if there are more levels of intermediaries between buyers and sellers.

Appendix: A polynomial time algorithm to compute the nucleolus of a non-negative additive game with an acyclic, quasi-strongly connected permission structure

In this appendix we recall the polynomial time algorithm of van den Brink *et al.* (2008). Although they use this algorithm to compute the nucleolus for a more general class of games

(those that satisfy the so-called weak digraph monotonicity and weak digraph convexity conditions) we only apply it to non-negative additive games with an acyclic, quasi-strongly connected permission structure in step 3 of the algorithm to compute the nucleolus for non-negative additive games with a (more general) acyclic permission structure. This algorithm computes the nucleolus in $\mathcal{O}(n^4)$ time. It is an adaptation of the algorithm of Arin and Feltkamp (1997) which computes the nucleolus of veto-rich games in exponential time. In the following, $\Omega^D = \Phi^D \setminus \{N\}$ denotes the collection of all feasible coalitions not equal to N .

Consider a given non-negative additive game with an acyclic, quasi-strongly connected permission structure (N, v, D) .

Algorithm

Step 1 Set $k = 0$, $U_0 = N$, $v_0 = v$, $D_0 = D$ and $r_0 = r$. Goto Step 2.

Step 2 Find $U_{k+1} \subset U_k$ satisfying

$$\tau(U_{k+1}, r_k) = \tau^*(r_k) \text{ and } |U_{k+1}| = \max_{\{U \in \Omega^{D_k} \mid \tau(U, r_k) = \tau^*(r_k)\}} |U|,$$

where $\tau^*(r_k) = \min_{U \in \Omega^{D_k}} \tau(U, r_k)$ with $\tau(U, r_k) = \frac{r_k(U_k) - r_k(U)}{|U_k \setminus U| + 1}$. Assign $y_j = \tau^*(r_k)$ to every player $j \in U_k \setminus U_{k+1}$. Goto Step 3.

Step 3 If $U_{k+1} = \{1\}$ then Goto Step 4. If $U_{k+1} \neq \{1\}$, let i_{k+1} be the unique top-player of the subgraph $(U_k \setminus U_{k+1}, D_k(U_k \setminus U_{k+1}))$ of the digraph (U_k, D_k) restricted to $U_k \setminus U_{k+1}$. Define game (U_{k+1}, v_{k+1}) by setting for every $U \subseteq U_{k+1}$,

$$v_{k+1}(U) = \begin{cases} v_k(U) & \text{if } P_{D_k}(i_{k+1}) \cap U = \emptyset \\ v_k(U \cup (U_k \setminus U_{k+1})) - \tau(U_{k+1}, r_k)|U_k \setminus U_{k+1}| & \text{else,} \end{cases} \quad (7.9)$$

digraph (U_{k+1}, D_{k+1}) given by

$$(i, j) \in D_{k+1} \text{ if } \begin{cases} (i, j) \in D_k \text{ or} \\ U_{k+1}. \end{cases} \quad (7.10)$$

and let r_{k+1} be the restricted game of $(U_{k+1}, v_{k+1}, D_{k+1})$. Set $k = k + 1$. Goto Step 2.

Step 4 Assign $y_1 = v(N) - \sum_{j \in N \setminus \{1\}} x_j$. Stop.

References

- Arin, J., and Feltkamp, V. (1997), The nucleolus and kernel of veto-rich transferable utility games, *International Journal of Game Theory* 26, 61-73.
- Bilbao, J.M., Jiménez-Losada, A., Lebrón, E., and Chacón, C. (2005), Values for interior operator games, *Annals of Operations Research*, 137, 141-160.
- Bilbao, J.M., Chacón, C., Jiménez-Losada, A., and Lebrón, E. (2008), Convexity properties for interior operator games, *Annals of Operations Research*, 158, 117-131.
- Bondareva, O. (1962), The theory of the Core in an n -person game, *Vestnik Leningrad. Univ.* 13, 141-142 (in Russian).
- Brânzei, R., Fragnelli, V., and Tijs, S. (2002), Tree connected line graph peer group situations and line graph peer group games, *Mathematical Methods of Operations Research* 55, 93-106.
- Brânzei, R., Solymosi, T., and Tijs, S. (2005), Strongly essential coalitions and the nucleolus of peer group games, *International Journal of Game Theory* 33, 447-460.
- Brink, R. van den (1997), An axiomatization of the disjunctive permission value for games with a permission structure, *International Journal of Game Theory* 26, 27-43.
- Brink, R. van den, and Gilles, R. P. (1996), Axiomatizations of the conjunctive permission value for games with permission structures, *Games and Economic Behavior* 12, 113-126.
- Brink, R. van den, I. Katsev and G. van der Laan (2008), Computation of the nucleolus for a class of disjunctive games with a permission structure, Discussion Paper TI 08-060/1, Tinbergen Institute, Amsterdam/Rotterdam, The Netherlands.
- Brune, S. (1983), On the regions of linearity for the nucleolus and their computation, *International Journal of Game Theory* 12, 47-80.
- Gilles, R. P., and Owen, G. (1994), Cooperative games and disjunctive permission structures, Department of Economics, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.
- Gilles, R. P., Owen, G., and Brink, R. van den (1992), Games with permission structures: the conjunctive approach, *International Journal of Game Theory* 20, 277-293.

- Gillies, D.B. (1953), *Some Theorems on n-Person Games*, Princeton University Press, Princeton, NJ.
- Huberman, G. (1980), The nucleolus and essential coalitions, in: A. Bensoussan and J. Lions (eds.) *Analysis and Optimization of Systems*, Lecture Notes in Control and Information Sciences 28, Springer, Berlin, pp. 416-422.
- Kohlberg, E., (1979), On the nucleolus of a characteristic function game, *SIAM Journal of Applied Mathematics* 20, 62-66.
- Muto, S., Potters, J., and Tijs, S. (1989), Information market games, *International Journal of Game Theory* 18, 209-226.
- Myerson, R. B. (1977), Graphs and cooperation in games, *Mathematics of Operations Research* 2, 225-229.
- Ni, D., and Y. Wang (2007), Sharing a polluted river, *Games and Economic Behavior*, 60, 176-186.
- Reynierse, J., and Potters, J. (1998), The \mathcal{B} -nucleolus of TU-games, *Games and Economic Behavior* 24, 77-96.
- Schmeidler, D. (1969), The nucleolus of a characteristic function game, *SIAM Journal on Applied Mathematics* 17, 1163-1170.
- Shapley, L. S. (1953), A value for N-person games, in: *Contributions to the Theory of Games, Vol II* (eds. H. W. Kuhn, and A. W. Tucker), Princeton University Press, Princeton, 307-317.
- Shapley, L.S. (1967), On balanced sets and cores, *Naval Research Logistics Quarterly* 14, 453-460.
- Shapley, L., and M. Shubik (1972), The assignment game I: the core, *International Journal of Game Theory*, 1, 111-130.