Social Exchange Networks With Distant Bargaining☆

Konstantinos Georgiou[1], George Karakostas[2,*], Jochen Könemann[1], Zuzanna Stamirowska[3]

Abstract

Network bargaining is a natural extension of the classical, 2-player *Nash bargaining* solution to the network setting. Here one is given an exchange network G connecting a set of players V in which edges correspond to potential contracts between their endpoints. In the standard model, a player may engage in at most one contract, and feasible outcomes therefore correspond to matchings in the underlying graph. Kleinberg and Tardos [STOC'08] recently proposed this model, and introduced the concepts of *stability* and *balance* for feasible outcomes. The authors characterized the class of instances that admit such solutions, and presented a polynomial-time algorithm to compute them.

In this paper, we generalize the work of Kleinberg and Tardos by allowing agents to engage into more complex contracts that span more than two agents. We provide suitable generalizations of the above stability and balance notions, and show that many of the previously known results for the matching case extend to our new setting. In particular, we can show that a given instance admits a stable outcome only if it also admits a balanced one. Like Bateni et al. [ICALP'10] we exploit connections to cooperative games. We fully characterize the core of these games, and show that checking its non-emptiness is NP-complete. On the other hand, we provide efficient algorithms to compute core elements for several special cases of the problem, making use of compact linear programming formulations.

Keywords: bargaining, exchange networks, cooperative games

1. Introduction

The study of bargaining has been a central theme in economics and sociology, since it constitutes a basic activity in any human society. The most basic bargaining model is that of two agents A and B that negotiate how to divide a good of a certain value (say, 1) amongst themselves, while at the same time each has an *outside option* of value α and β respectively. The famous Nash bargaining solution [1] postulates that in an *equitable* outcome, each player should receive her outside option, and that the surplus $s = 1 - \alpha - \beta$ is to be split evenly between A and B.

☆This work was initiated in the International Problem Solving Workshop, held in July 16-20, 2012. and organized as part of the MITACS International Focus Period "Advances in Network Analysis and its Applications". We would like to thank MITACS for this great opportunity. Research partially supported by NSERC.

*Corresponding author

Email addresses: k2georgiou@math.uwaterloo.ca (Konstantinos Georgiou), karakos@mcmaster.ca (George Karakostas), jochen@math.uwaterloo.ca (Jochen Könemann), zuzanna.stamirowska@sciences-po.org (Zuzanna Stamirowska)

[1]Department of Combinatorics & Optimization, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada

[2]Department of Computing & Software, McMaster University, Hamilton, Ontario L8S 4K1, Canada

[3]École Polytechnique, Paris, France

More recently, Kleinberg and Tardos [2] proposed the following natural *network* extension of this game. Here, the set of players corresponds to the vertices of an undirected graph $G = (V, E)$; each edge $ij \in E$ represents a potential *contract* between players i and j of value $w_{ij} \geq 0$. In Kleinberg and Tardos' model, players are restricted to form contracts with at most one of their neighbours. Outcomes of the *network bargaining* game are therefore given by a matching $M \subseteq E$, and an *allocation* $x \in \mathbb{R}^V_+$ such that $x_i + x_j = w_{ij}$ for all $ij \in M$, and $x_i = 0$ if i is not incident to an edge of M.

Unlike in the non-network bargaining game, the outside option α_i of player is not a given parameter but rather implicitly determined by the network neighbourhood of i. Specifically, in an outcome (M, x), player i's outside option is defined as $\alpha_i = \max\{w_{ij} - x_j : ij \in \delta(i) \setminus M\}$, where $\delta(i)$ is the set of edges incident to i. An outcome (M, x) is then called *stable* if $x_i + x_j \geq w_{ij}$ for all edges $ij \in E$, and it is *balanced* if in addition, the value of the edges in M is split according to Nash's bargaining solution; i.e., for an edge ij, $x_i - \alpha_i = x_j - \alpha_j$. Kleinberg and Tardos provide a characterization of the class of graphs that admit balanced outcomes, and present a combinatorial algorithm that computes one if it exists.

Bateni et al. [3] recently exhibited a close link between the study of network bargaining and that of *matching games* in cooperative game theory. The authors showed that stable outcomes for an instance of network bargaining correspond to allocations in the *core* of the underlying matching game. Moreover, balanced outcomes correspond to *prekernel* allocations. As a corollary, this implies that an algorithm by Faigle et al. [4] gives an alternate method to obtain balanced outcomes in a network bargaining game. Bateni et al. also extended the work of [2] to bipartite graphs in which the agents of one side are allowed to engage in more than one contract.

Matching games have indeed been studied extensively in the game theory community since the early 70s, when Shapley and Shubik investigated the core of the class of bipartite matching games, so called *assignment games*, in their seminal paper [5]. Granot and Granot [6] also study the core of the assignment game; the authors show that it contains many points, some of which may not be desirable ways to share revenue. The authors propose to focus on the intersection of core and prekernel instead, and provide sufficient and necessary conditions for the former to be contained in the latter. Deng et al. [7] generalized the work of Shapley and Shubik to matchings in general graphs as well as to cooperative games of many other combinatorial optimization problems. We refer the reader also to the recent survey paper [8] and the excellent textbook [9].

In this paper we further generalize the work of [2] and [3] on network bargaining by allowing contracts to span more than two agents. Our study is motivated by bargaining settings where goods are complex composites of other goods that are under the control of autonomous agents. For example, in a computer network setting, two hosts A and B may wish to establish a connection between themselves. Any such connection may involve physical links from a number of smaller autonomous networks that are provisioned by individual players. In this setting, value generated by the connection between A and B cannot merely be shared by the two hosts, but must also be used to compensate those *facilitators* whose networks the connection uses.

1.1. Generalized network bargaining

We formalize the above ideas by defining the class of *generalized network bargaining* (GNB) games. In an instance of such a game, we are given a (directed or undirected) graph $G = (V, E)$ whose vertices correspond to players, and edges that correspond to atomic goods; the *value* of the good corresponding to e is given by $w_e \geq 0$. We assume that V is partitioned into *terminals* T, and *facilitators* R. Intuitively, the terminals are the *active* players that seek participation in contracts, while facilitators are *passive*, and may get involved in contracts, but do not seek involvement. We further let $\mathcal{C}$ be a family of *contracts* each of whom consists of a collection of atomic goods. We let $w(c)$ be the *value* of contract c which we simply define as the sum of values w_e of the edges $e \in c$. We note here that in the work of [2] and [3], $\mathcal{C}$ consists just of the singleton edges.

A set $C \subseteq \mathcal{C}$ of contracts is called *feasible* if each two contracts in C are vertex disjoint. An *outcome* of an instance of GNB is given by a feasible collection $C \subseteq \mathcal{C}$ as well as an allocation $x \in \mathbb{R}^V_+$ of the contract values to the players such that

$$x(c) := \sum_{v \in c} x_v = w(c).$$

Which outcomes are *desirable*? We propose the following natural extensions of the notions of *stability* and *balance* of [2]. Consider an outcome (C, x) of some instance of GNB. Then define the *outside option* α_i of player i as

$$\alpha_i := \max_{c \in \mathcal{C} : i \in c \notin C} \{w(c) - x(c)\} + x_i.$$

Intuitively, the outside option of i is given by the value she can earn by breaking her current contract, and participating in a contract that is not part of the current outcome. We will assume that each agent i is incident to a self-loop of value 0, and hence has the option of not collaborating with anyone else. In what follows $a(c) := \sum_{v \in c} a_v$ for a contract $c \in C$.

Having defined α_i, we can now introduce the notions of *stability* and *balance*. An outcome (C, x) is stable if $x_i \geq \alpha_i$ for all agents i: every agent earns at least her outside option. Again extending the concept of Nash bargaining solution in the most natural way, we say that an outcome is balanced if the surplus of each contract is shared evenly among the participating agents. Formally, for all $c \in C$, and for all $i \in c$ we require

$$x_i = \alpha_i + \frac{w(c) - \alpha(c)}{|c|}.$$

Equivalently, this means that $x_i - \alpha_i = x_j - \alpha_j$ for all $i, j \in c$, and for all $c \in C$.

1.2. Our results

Following Kleinberg and Tardos, we are interested in (a) characterizing the class of GNB instances that have stable and balanced outcomes, and (b) in computing such outcomes efficiently whenever they exist. Similar to [3], we first identify a natural *cooperative* game $\Gamma(I)$ associated with a given GNB instance I. $\Gamma(I)$ has player set V and the value function is defined by letting

$$v(S) = \max_{C \subseteq \mathcal{C}(S),\ C \text{ feasible}} \sum_{c \in C} w(c),$$

for all $S \subseteq V$, where $\mathcal{C}(S)$ is the set of contracts contained in the set S. We briefly introduce a few pertinent solution concepts for cooperative games, and refer the reader to [9] for a comprehensive introduction to the topic. The *core* $\mathbf{C}$ of $\Gamma(I)$ consists of all allocations $x \in \mathbb{R}^V_+$ that satisfy $x(S) \geq v(S)$ for all $S \subseteq V$, and this inequality is tight for $S = V$. The *power* of agent i over agent j is given by

$$s_{ij}(x) = \max_{S \subseteq V : i \in S, j \notin S} v(S) - x(S), \tag{1}$$

and the *prekernel* $\mathbf{P}$ consists of all allocations x for which $s_{ij} = s_{ji}$ for all agents i and j. In the following first result, let stable be the projection of the collection of all stable outcomes (C, x) onto the x-space. Similarly, $\mathsf{balance}$ is the projection of all balanced outcomes onto the x-space.

Theorem 1.1. *Let I be an instance of GNB, and $\Gamma(I)$ the corresponding cooperative game. Then $\boldsymbol{C} = \mathsf{stable}$, and $\boldsymbol{C} \cap \boldsymbol{P} \subseteq \mathsf{stable} \cap \mathsf{balance}$. There are instances of GNB where this inclusion is strict.*

It is well known (e.g., see [4]) that if the core of $\Gamma(I)$ is non-empty then so is the intersection of core and prekernel. We therefore obtain the following corollary.

Corollary 1.2. *Every GNB instance with a stable solution also admits a balanced one.*

But can one find stable and balanced solutions efficiently? As it turns out (see below) not always. However, given a point in the core, and an efficient oracle for the computation of powers (as specified in (1)), we can find a point in the prekernel of $\Gamma(I)$ via a result by Faigle, Kern and Kuipers [4] (see also [10]). We obtain the following corollary.

Corollary 1.3. *There is a polynomial-time algorithm to compute stable and balanced solutions for an instance of GNB if (a) we have a polynomial-time method to compute a point in the core, and (b)* (1) *can be computed efficiently.*

Unfortunately, computing s_{ij} in (1) may amount to solving an NP-hard problem; e.g., when $\mathcal{C}$ consists of all paths in the given graph, one easily sees that a poly-time oracle for computing powers would enable us to solve the NP-hard *longest path* problem. Nevertheless, there are many families of instances of interest where the conditions of Corollary 1.2 are satisfied, e.g. instances where $\mathcal{C}$ is explicitly given as part of the input, or whenever the family $\mathcal{C}$ induces an acyclic subgraph of the input graph.

In light of Corollary 1.2, in order to characterize instances of GNB that have stable and balanced solutions, we may characterize the set of instances I for which $\Gamma(I)$ has a non-empty core. We can show the following.

Theorem 1.4. *For a given GNB instance I, we can write a linear program* (P_1) *that has an integral optimal solution iff the core of $\Gamma(I)$ is non-empty.*

Hence (P_1) fully characterizes the class of GNB instances that admit stable and balanced solutions. We may, however, not be able to solve the LP.

Theorem 1.5. *Given an instance I of GNB, it is NP-complete to (a) check whether the core of $\Gamma(I)$ is non-empty, and (b) check whether a specific allocation $x \in \mathbb{R}^n_+$ is in the core.*

In this theorem, we assume that $\mathcal{C}$ is part of the input. We can show (a) by using a reduction from *exact-cover by* 3*-sets* following a previous result by Conitzer and Sandholm [11] closely. Part (b) employs a reduction from 3*-dimensional matching*, and is similar to a result for minimum-cost spanning tree games by Faigle et al. [12].

We note here that the results in Theorems 1.1, 1.4 and 1.5 do not rely on the specific type of underlying graph (i.e., *directed* or *undirected*). Departing from this, our next result focuses on GNB instances whose contract set is implicitly given as the set of all terminal-terminal paths in a *directed* graph. For such instances I, we present efficiently solvable linear programs (P_2) and (P_3) that are integral only if the core of $\Gamma(I)$ is non-empty.

Theorem 1.6. *Given an instance I of GNB where $\mathcal{C}$ is the set of all terminal-terminal paths in an underlying directed graph, we can find efficiently solvable LPs* (P_2) *and* (P_3) *that are integral only if the core of* $\Gamma(I)$ *is non-empty.*

Unfortunately, the latter two LPs do not fully characterize core non-emptiness of $\Gamma(I)$, and there are instances with non-empty core for which the two LPs are fractional. The two LPs are not equivalent, and there are instance of GNB where one of the two LPs is fractional and the other is not.

2. Computing balanced outcomes

The goal of this section is to provide a proof of Theorem 1.1. Let us fix an instance I of GNB with graph $G = (V, E)$, and weights w_e for all $e \in E$. Recall that the cooperative game $\Gamma(I)$ for I has player set V, and that the value $v(S)$ of a coalition $S \subseteq V$ is given by the maximum value of any feasible collection of contracts that are entirely contained in S. We first make the following observation.

Observation 2.1. *Computing $v(V)$ for Distant Bargaining Games G=(V,E), for which the set of feasible contracts is part of the input, is NP-hard.*

Proof. The reduction is from 3-dimensional matching (3DM) where all three vertex sets have the same size. Given an instance $H = (L \cup M \cup R, F)$ of this problem (where F contains hyperedges, each containing exactly one vertex from each L, M, R), we consider the following Distant Bargaining instance. The set of terminals is $L \cup R$, and the set of facilitators is M. For every hyperedge, we introduce two new edges connecting each of the two terminals to the facilitator, each of weight $1/2$, as well as we introduce the associated contract (of weight 1) containing exactly these two edges; we still allow only the "hyperedge contracts" to be formed by the new edges, and no other combinations. Finally, it is easy to see that $H = (L \cup M \cup R, F)$ admits a 3d-matching if and only if $v(L \cup M \cup R) = |L|$. □

We will now relate the core of $\Gamma(I)$ and the set of stable outcomes of I. In order to do this, we need the following lemma, and leave its straight-forward proof to the reader.

Lemma 2.2. *Let x be an allocation in the core. Then there is a feasible collection $C \subseteq \mathcal{C}$ of maximum value such that $\sum_i x_i = \sum_{c \in C} w(c)$.*

The following lemma shows that core and set of stable allocations coincide.

Lemma 2.3. ***C*** $=$ *stable*

Proof. Let $x \in \mathbf{C}$, and note that by definition, $x(c) \geq w(c)$ for all $c \in \mathcal{C}$. Again from the definition of the outside option, this means that $\alpha_i \geq 0$ for all agents i, and hence $x \in$ stable.

Conversely, let x be a stable allocation and let $C \subseteq \mathcal{C}$ be the associated feasible set of contracts. We clearly have $x(c) = w(c)$ for all $c \in C$, and $x(c') \geq w(c')$ for all $c' \notin C$. Consider a set S of players, and let $c'_1, \ldots, c'_k \in \mathcal{C}$ be a feasible collection of contracts that defines the value of S; i.e.,

$$v(S) = w(c'_1) + \ldots + w(c'_k).$$

From before we know that $x(c'_i) \geq w(c'_i)$ for all i, and feasibility therefore shows that $x(S) \geq \sum_i x(c'_i) \geq v(S)$. Finally, note that this argument shows that $x(V) \geq v(V)$, but clearly, this latter inequality must be tight from the definition of outcomes. □

We now show that solutions in the prekernel $\mathbf{P}$ are balanced.

Lemma 2.4. *$\mathbf{C} \cap \mathbf{P} \subseteq$ stable $\cap$ balance*

Proof. Let $x \in \mathbf{C} \cap \mathbf{P}$. By Lemma 2.3 we know that $x \in$ stable. Hence it remains to argue that x is also balanced. We first argue that for all agents i, j, whenever $x \in \mathbf{C}$, there must be a contract c containing i and not j such that $s_{ij} = v(c) - x(c)$. Indeed, suppose that

$$s_{ij} = v(S) - x(S),$$

for some $S \subseteq V$ for which $i \in S \not\ni j$. Then let $c_1, \ldots, c_t \in \mathcal{C}(S)$ be a feasible collection of contracts whose joint value equals $v(S)$. Without loss of generality, suppose that $i \in c_1$. Then for every contract c_r, $x \in \mathbf{C}$ implies that $w(c_r) - x(c_r) \leq 0$, and hence, as claimed,

$$v(S) - x(S) = \sum_{r=1}^{t} (w(c_r) - x(c_r)) \leq w(c_1) - x(c_1).$$

Since $x \in \mathbf{P}$ we know that $s_{ij}(x) = s_{ji}(x)$, for all $i, j \in V$. From Lemma 2.2 we also know that x corresponds to some maximum value set of feasible contracts, say, C. Fix a contract $c \in C$ and two agents $i, j \in C$. In what follows we argue that $\alpha_i - x_i = \alpha_j - x_j$, which directly implies that $x \in$ balance.

For the sake of simplicity, we denote $\arg\max \alpha_i$ and $\arg\max s_{ij}(x)$ by c_i and q_{ij} respectively. Then we note that if $i \notin c_j$ then $q_{ji} = c_j$ and hence $s_{ji}(x) = \alpha_j - x_j$. Also, if $i \in c_j$, then $\alpha_j - x_j \geq s_{ji}(x)$, since the set c_j is not considered when we maximize over subsets in order to find $s_{ij}(x)$.

With these observations at hand, we can now examine three cases. First, suppose that $i \notin c_j$ and $j \notin c_i$. Then $s_{ij}(x) = s_{ji}(x)$ implies that $\alpha_i - x_i = \alpha_j - x_j$. In the second case, if $i \in c_j$ and $j \in c_i$, then $c_i = c_j$, so again $\alpha_i - x_i = \alpha_j - x_j$. Finally, in the third case we assume that $i \notin c_j$ (and therefore $q_{ji} = c_j$, that is $s_{ji}(x) = \alpha_j - x_j$) and that $j \in c_i$ (and so, $\alpha_i - x_i \geq s_{ij}(x)$). It follows that if $s_{ij}(x) = s_{ji}(x)$, then $\alpha_i - x_i \geq \alpha_j - x_j$. Also, since $j \in c_i$ we conclude that $\alpha_j - x_j \geq \alpha_i - x_i$. Overall, this implies again that $\alpha_j - x_j = \alpha_i - x_i$, as we wanted. $\square$

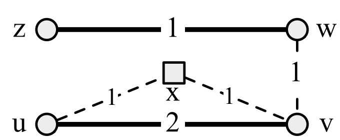

Figure 1: Counter-example for the reverse inclusion in Lemma 2.4

Figure 1 shows an instance of GNB with terminals $\{u, v, w, z\}$ and contracts $\mathcal{C} = \{uv, vw, wz, uz, uxv\}$. Consider feasible contracts $C = \{uv, wz\}$ of total value 3. The allocation $\chi_u = \chi_v = 1$, $\chi_w = \chi_z = 1/2$, and $\chi_x = 0$ is easily checked to be stable and balanced. However, since $s_{uv} = 0 - (1/2 + 1) = -3/2$, and $s_{vu} = 1 - (1/2 + 1) = -1/2$, χ is not in the prekernel. Together with Lemmata 2.3 and 2.4 we obtain a proof of Theorem 1.1.

3. Characterizing the core

As we have seen in Lemma 2.3, the set of stable allocations for a GNB instance I equals the core of the cooperative game $\Gamma(I)$. In this section, our goal will be to characterize instances I where $\Gamma(I)$ has a non-empty core. Further, if the core of $\Gamma(I)$ is non-empty then we will investigate the computational complexity of finding such a point.

3.1. The core via linear programming

We start this section by presenting a linear programming formulation that is integral iff the core of $\Gamma(I)$ is non-empty. The LP has a variable z_c for each contract $c \in \mathcal{C}$, and maximizes the total value of chosen contracts subject to feasibility. The LP is shown on the left below.

$$
\begin{aligned}
\max \quad & \sum_{c \in \mathcal{C}} w(c)\, z_c && && (\mathrm{P}_1)\\
\text{s.t.} \quad & \sum_{c : i \in c} z_c \le 1, && \forall i \in V\\
& z \ge \nvdash
\end{aligned}
\qquad\qquad
\begin{aligned}
\min \quad & \sum_{i \in V} y_i && && (\mathrm{D}_1)\\
\text{s.t.} \quad & \sum_{i \in c} y_i \ge w(c), && \forall c \in \mathcal{C}\\
& y \ge \nvdash.
\end{aligned}
$$

The LP on the right is the linear programming dual of (P_1). It has a variable y_i for each agent $i \in V$, and a constraint for every $c \in \mathcal{C}$. We will now present a proof of Theorem 1.4, and show that the core of $\Gamma(I)$ is non-empty iff (P_1) has an integral optimal solution.

Proof of Theorem 1.4. Recall that by Lemma 2.3, the core of $\Gamma(I)$ equals the set stable of stable allocations in I. Also recall that an outcome (C, x) is stable iff for all $c' \notin C$, $x(c') \ge w(c')$ and $w(c) = x(c)$ for all $c \in C$.

Now suppose that (P_1) has an integral optimal solution z, and let y be the corresponding optimal dual solution. Clearly, $\nvdash \le z \le \nVdash$, and hence we may define the set $C \subseteq \mathcal{C}$ of contracts c with $z_c = 1$. We now claim that (C, y) is a stable outcome. Indeed, all stability conditions are provided by the dual constraints, and by complementary slackness, they are tight when $z_c > 0$.

For the other direction, consider a stable outcome (C, x). It is easy to see that $z_P = 1$ for $P \in C$ and 0 otherwise, and $y = x$ are primal and dual feasible solutions respectively. Complementary slackness is implied exactly by the definition of outcomes that require that the sum of agent earnings in each contract matches the contract surplus. $\square$

We do not know how to solve (P_1) efficiently. Worse than that, even if we are able to solve the LP, we may not be able to decide whether there is an integral optimal solution. The proof of the following result is implicit in [11], and given here for completeness.

Lemma 3.1. *Given an instance I of GNB it is NP-complete to decide whether the core of $\Gamma(I)$ is non-empty.*

Proof. We first show that the problem is in NP. For this, we non-deterministically guess a feasible collection $C \subseteq \mathcal{C}$ of contracts. We then solve the linear system

$$x(c) \;=\; w(c) \quad \forall c \in C \tag{2}$$
$$x(c) \;\ge\; w(c) \quad \forall c \in \mathcal{C} \setminus C. \tag{3}$$

in order to find $x \in \mathbb{R}^V_+$. This can be done in polynomial time (e.g., via linear programming) as $\mathcal{C}$ is part of the input. It is easy to check that the system has a feasible solution if x is in the core.

To show hardness, we reduce from an instance of *exact cover by* 3*-sets* (X3C), where we are given a ground-set S of size $3m$ and subsets $\{S_1, \ldots, S_q\}$ of S each of which has size 3. The question is whether there are m pairs whose union is S.

Here is how we encode this problem as an instance of GNB. We create a graph G with vertex set $S \cup \{x, y\}$, where x and y are two new dummy vertices. For each $S_i = \{a, b, c\}$ in the X3C instance, we add distinct edges ab and bc each of cost $3/2$ (middle vertex is chosen, say, lexicographically), and we add abc to the list of allowed contracts $\mathcal{C}$. We also add trees T_x and T_y spanning $S \cup \{x\}$, and $S \cup \{y\}$, respectively. Once again, the edge sets of T_x and T_y are disjoint, and distinct from the other edges added previously. We distribute weight $6m$ over the edges of T_x, and similarly over the edges of T_y in some arbitrary way, and add $E(T_x)$ and $E(T_y)$ to the set of allowed contracts. Finally we add xy to the graph and contract set, and assign a weight of $6m$ to this edge. We claim that the core of the game $\Gamma(I)$ of the above instance is non-empty iff the given X3C instance is a 'yes' instance.

Assume first that $S_1, \ldots, S_m$ is an exact 3-cover. In this case, note that the corresponding contracts together with xy are feasible, and have joint value $9m$. One can now verify that χ with $\chi_i = 1$, for all $i \in S$, and $\chi_x = \chi_y = 3m$ is in the core.

Conversely, if no exact 3-cover exists, then the value $v(S)$ is less than $3m$, and the value of the grand coalition is less than $9m$. Consider any vector $\chi \in \mathbb{R}^{S \cup \{x,y\}}_+$ such that $\chi(V) = v(V) < 9m$. It is not difficult to see that there are two distinct sets

$$U, W \in \{S, \{x\}, \{y\}\},$$

such that $\chi(U) + \chi(W) < 6m$. But $U \cup W$ is a coalition of value $6m$ by our definition, and χ is therefore not in the core. □

So, even if (P$_1$) can be solved efficiently, we may not be able to check efficiently for an integral optimal solution. We now show that it is also hard to check whether a certain allocation is in the core, which in combination with Lemma 3.1 conclude Theorem 1.5.

Lemma 3.2. *It is NP-complete to check whether an allocation $x \in \mathbb{R}^V_+$ is in the core of the cooperative game of a GNB instance I.*

Proof. The problem is certainly contained in NP. To see this, we first non-deterministically guess a feasible collection $C \subseteq \mathcal{C}$ and then check that (2) and (3) hold.

To prove hardness, we once again reduce from the 3-dimensional matching problem. Given an instance of 3DM, we create an instance of GNB by creating a graph with terminal vertices $L \cup M \cup R$. For each $(l, m, r) \in F$, we add edges lm and mr of value $1/2$ each, and add contract $\{lm, mr\}$ to the set $\mathcal{C}$ of allowed contracts.

Consider the vector χ with $\chi_v = 1$ if $v \in M$, and $\chi_v = 0$ otherwise. We claim that χ is in the core iff the given 3DM instance is a 'yes' instance.

If the given 3DM instance is a 'no' instance, then $v(V) < |M| = \chi(V)$, and hence χ is not in the core. Conversely suppose that the 3DM instance is a 'yes' instance. In this case, $\chi(V) = |M| = v(V)$, and clearly $\chi(c) = 1 = w(c)$ for every contract $c \in C$. □

3.2. Implicitly given contracts

In this section, we focus on GNB instances where $\mathcal{C}$ is implicitly given as the set of *all terminal-terminal paths* in an underlying directed graph D with node-set V, and arcs A. The internal nodes of each of these paths are assumed to be facilitators. Thus, $\mathcal{C}$ is not part of the input, and LP (P_1) may have an exponential number of variables. We do not know how to efficiently solve this LP in this case. In the following, we present two LPs for a given instance of GNB that (a) have integral optimal solutions only if the core of $\Gamma(I)$ is non-empty, and (b) are poly-time solvable.

In the following, let us fix an instance I of GNB with graph $D = (V, A)$, and weights w_{uv} for all arcs $(u, v) \in A$. The two LPs to be presented are *flow* formulations.

3.2.1. Cycle-free Flow Formulation

Observe that a set C of arcs in D corresponds to a feasible set of contracts iff (a) every terminal agent has at most one incident arc, (b) every facilitator agent has at most one outgoing arc, (c) every facilitator has equally many incoming and outgoing arcs, and (d) for every set of facilitators S, there is at least one outgoing arc in C if there is an arc in C that has both endpoints in S. Therefore, the following LP is a relaxation for computing the value of the grand coalition (recall that contracts are all terminal-terminal paths). For a set S of nodes, we let $\delta^+(S)$ be the set of arcs with tail in and head outside S. Furthermore, we let $\gamma(S)$ be the set of arcs with both ends in S.

$$
\begin{aligned}
\max \quad & \sum_{a \in A} w_a x_a && && (\mathrm{P}_2)\\
\text{s.t.} \quad & x\left(\delta^-(v)\right) + x\left(\delta^+(v)\right) \le 1 && \forall v \in T &&\\
& x(\delta^+(v)) \le 1 && \forall v \in R &&\\
& x\left(\delta^-(v)\right) - x\left(\delta^+(v)\right) = 0 && \forall v \in R &&\\
& x\left(\delta^+(S)\right) \ge x_a && \forall S \subseteq R,\ \forall a \in \gamma(S) && (4)\\
& x \ge 0 && &&
\end{aligned}
$$

Note that (P_2) can be solved in polynomial time via the Ellipsoid method [13]: given a candidate solution x, it can be efficiently checked whether one of the polynomially many constraints of one of the first three types is violated. Separating the constraints of type (4) can be reduced to a polynomial number of minimum-cut computations in suitable auxiliary graphs. We leave the details to the reader.

Lemma 3.3. *If LP (P_2) for GNB instance I has an integral optimal solution, then the core of $\Gamma(I)$ is non-empty.*

Proof. We need to find an outcome χ such that for every $S \subseteq V$, $\chi(S) - v(S) \ge 0$, as well as $\chi(V) = v(V)$. We claim that χ can be determined by considering the dual of (P_2). For this we introduce the dual variables $\alpha_v, y_v, z_v, \beta_{S,a}$, corresponding to the constraints in the order they appear in (P_2). Then the dual reads as follows.

$$\begin{aligned}
\min \quad & \sum_{v\in T}\alpha_v + \sum_{v\in R} y_v && && \text{(D}_2\text{)}\\
\text{s.t.} \quad & y_u + z_u - z_v - \sum_{S,a:(u,v)\in\delta^+(S)} \beta_{S,a} + \sum_{S:(u,v)\in\gamma(S)} \beta_{S,(u,v)} \geq w_{uv} && \forall (u,v)\in\gamma(R)\\
& \alpha_v - z_v - y_u - \sum_{S,a:(u,v)\in\delta^+(S)} \beta_{S,a} \geq w_{uv} && \forall (u,v)\in A, u\in R, v\in T\\
& \alpha_u + z_v \geq w_{uv} && \forall (u,v)\in A, u\in T, v\in R\\
& \alpha_v + \alpha_u \geq w_{uv} && \forall (u,v)\in\gamma(T)\\
& \alpha, y, \beta \geq 0
\end{aligned}$$

We set $\chi_v = \alpha_v$ for all $v \in T$ and $\chi_v = y_v$ for all $v \in R$, and we claim that if (P$_2$) has integrality gap 1, in which case there is a matching dual solution, then $\chi \in \mathbf{C}$. To see this, consider $S \subseteq V$. We need to show that $\chi(S) - v(S) \geq 0$, and that this constraint is tight for the grand coalition $S = V$. One quickly sees that it suffices to show that for any path-contract P, $x(P) - w(P) \geq 0$. This is because

$$v(S) = w(P_1) + \ldots + w(P_k)$$

for some feasible collection of terminal-terminal paths $P_1, \ldots, P_k$.

So, fix a contract path $P = v_0, \ldots, v_p$, where $v_0, v_p \in T$, and the rest of the agents are facilitators in R. We sum all dual constraints that correspond to the arcs $(v_0, v_1), \ldots, (v_{p-1}, v_p)$, deriving

$$\alpha_{v_0} + \alpha_{v_p} + \sum_{i=1}^{p-1} y_{v_i} + \sum_{i=1}^{p-2}\left(-\sum_{S,a:(v_iv_{i+1})\in\delta^+(S)} \beta_{S,a} + \sum_{S:(v_i,v_{i+1})\in\gamma(S)} \beta_{S,(v_i,v_{i+1})}\right) - \sum_{S,a:v_{p-1}v_p\in\delta^+(S)} \beta_{S,a} \geq \sum_{i=0}^{p-1} w_{v_iv_{i+1}}$$

or in other words that

$$\alpha_{v_0} + \alpha_{v_p} + \sum_{i=1}^{p-1} y_{v_i} + \sum_{S,a} \beta_{S,a}\left(-\left|\delta^+(S)\cap P\right| + \xi(a\in P)\right) \geq w(P), \tag{5}$$

where $\xi(a \in P)$ is 1 if $a \in P$ and 0 otherwise. We note then that if $a \in P$, then since P is a path contract, there is at least one edge leaving S (if S is non empty), since all cuts S considered are subsets of the facilitators R. It follows that $-|\delta^+(S)\cap P| + \xi(a \in P) \leq 0$, and by our definition of χ, inequality (5) implies that $\chi(P) \geq w(P)$.

Finally, consider the grand coalition $S = V$. Clearly, $\chi(V)$ is the value of the dual (D$_2$) which agrees with the optimal value of the (P$_2$). The latter is exactly the value of the grand coalition $v(V)$, since we assumed that the primal has integrality gap 1. □

Note that Lemma 3.3 is a direct implication of Theorem 1.4. The reason is that any solution feasible to (P$_1$) can be converted to a feasible solution to (P$_2$) (of equal objective value) as follows; for every $uv \in A$ set

$\chi_{uv} = \frac{1}{2}\sum_{P\in\mathcal{P}:\, uv\in P} z_P$. It follows that the optimal value of (P$_1$) is sandwiched between the value of the cooperative game $\Gamma(I)$, and that of (P$_2$). Taking into consideration that (P$_2$) restricted to integral values is an exact formulation of our problem, if (P$_2$) has integrality gap 1, so does (P$_1$). Nevertheless, the important observation is that unlike (P$_1$), we know how to efficiently solve relaxation (P$_2$). Furthermore, the proof of Lemma 3.3 is constructive, and gives rise to an efficient algorithm to compute a core allocation.

Unfortunately, we will later see that there are example instances of GNB with non-empty core for which (P$_2$) has no integral optimal solution (see Lemma 3.6). There are, however, many natural instance classes of GNB for which we are able to find core allocations if these exist via our LP. An example is a class of multi-layered graphs where the nodes are partitioned in k layers $L_1, \ldots, L_k$, with $(L_1 \cup L_2) = T, (L_2 \cup L_3 \cup \ldots \cup L_{k-1}) = R$, and arcs existing only between nodes in consecutive layers (i.e., if $(u, v) \in E$ then $u \in L_i, v \in L_{i+1}$ for some $1 \leq i \leq k-1$). Note that the only contracts allowed are paths from terminals in L_1 to terminals in L_k. Each feasible solution we get if we relax (P$_2$) by removing constraints (4) can be mapped to a single-commodity flow on the network we get if we connect all nodes in L_1 to a source s and all nodes in L_k to a sink t (with arcs of weight 0), and we give capacity 1 to all nodes. Each such flow can also be mapped to a feasible solution of the relaxed (P$_2$). Since the flow polytope is integral, the optimal solution of the relaxed (P$_2$) is also integral; it also satisfies constraints (4), and, therefore, it is also an integral optimal solution for (P$_2$).

3.2.2. Subtour Formulation

We now present yet another polynomial-time solvable relaxation for GNB. Once again, we will see that the existence of an integral optimal solution implies non-emptiness of core for $\Gamma(I)$, and that the reverse of this statement is false. However, as we will see in Section 3.2.3, (P$_2$) and this new LP are incomparable, and one may have integer optimal solution when the other does not.

$$
\begin{aligned}
\max \quad & \sum_{a\in A} w_a x_a && && \text{(P}_3\text{)}\\
\text{s.t.} \quad & x(\delta^+(v)) + x(\delta^-(v)) \leq 1 && \forall v \in T\\
& x(\delta^+(v)) \leq 1 && \forall v \in R\\
& x(\delta^-(v)) - x(\delta^+(v)) = 0 && \forall v \in R\\
& x(\gamma(S)) \leq |S| - 1 && \forall S \subseteq R\\
& x \geq \nvdash
\end{aligned}
$$

It can be easily seen that (P$_3$) restricted on integral values models exactly the problem of computing the value of the grand coalition in the associated coalition game (recall that contracts are all terminal-terminal paths). Once again it is easily shown that (P$_3$) is polynomial-time solvable, and once again we utilize the Ellipsoid method. We observe that the function $(|S| - 1) - x(\gamma(S))$ is submodular. Separating the constraints of type (4) then reduces to submodular function minimization, for which there are polynomial-time algorithms (e.g., see [14]).

Similarly to the previous section, we show that (P$_3$) can be used as a certificate that the core is non empty, for some, but not all instances.

Lemma 3.4. *If LP (P$_3$) for GNB instance I has an integral optimal solution, then the core of $\Gamma(I)$ is non-empty.*

Proof. We will use the dual of (P3), which has variables α_v, y_v, z_v, and β_S corresponding to the constraints in (P3).

$$
\begin{aligned}
\min \quad & \sum_{u \in T} \alpha_u + \sum_{u \in R} y_u + \sum_{S \subseteq R} (|S| - 1)\beta_S && && \text{(D}_3\text{)}\\
\text{s.t.} \quad & y_v + z_u - z_v + \sum_{S:(u,v) \in \gamma(S)} \beta_S \geq w_{uv} && \forall (u,v) \in \gamma(R) \\
& \alpha_u + y_v - z_v \geq w_{uv} && \forall (u,v) : u \in T, v \in R \\
& \alpha_v + z_u \geq w_{uv} && \forall (u,v) : u \in R, v \in T \\
& \alpha_v + \alpha_u \geq w_{uv} && \forall (u,v) \in \gamma(T) \\
& \alpha, y, \beta \geq \nvdash
\end{aligned}
$$

Suppose that x is an optimal integral solution to (P3) and α, β, y, z the corresponding dual solution of (D3). Let $\mathcal{P}$ be the collection of simple (directed) paths from T to T corresponding to x. Note that it follows from complementary slackness that $\beta_S > 0$ only if the corresponding constraint in (P3) is tight. Hence, we obtain the following immediate claim.

Claim 3.5. *A dual variable β_S is non-zero only if S is the set of facilitator vertices of a contiguous subpath of a path in $\mathcal{P}$.*

We will now show that the following assignment χ is in the core:

$$
\chi_u := \begin{cases} \alpha_u, \text{ if } u \in T \\ y_u + \sum_{S:u \in S} \frac{|S|-1}{|S|} \beta_S, \text{ if } u \in R \end{cases}
$$

To see this, let p be a directed path contract (not necessarily in $\mathcal{P}$) with vertex set $V(p) = \{v_0, \ldots, v_q\}$, where v_0 and v_q are terminals, and all other v_i are facilitator vertices. Just like in the proof of Lemma 3.3, we sum the dual constraints corresponding to arcs on p, and obtain

$$
\alpha_{v_0} + y_{v_1} - z_{v_1} + \sum_{i=1}^{q-2} \left(y_{v_{i+1}} + z_{v_i} - z_{v_{i+1}} + \sum_{S:(v_i, v_{i+1}) \in \gamma(S)} \beta_S \right) + \alpha_{v_q} + z_{v_{q-1}} \geq w(p).
$$

Collecting terms, and rearranging gives

$$
\alpha_{v_0} + \alpha_{v_q} + \sum_{i=1}^{q-1} y_{v_i} + \sum_S \beta_S |\gamma(S) \cap p| \geq w(p), \tag{6}
$$

where $\gamma(S) \cap p$ denotes the set of (facilitator,facilitator) arcs of p that lie entirely in S. Now note that

$$
\sum_{i=0}^{q} \chi_{v_i} = \alpha_{v_0} + \alpha_{v_q} + \sum_{i=1}^{q-1} y_{v_i} + \sum_S |S \cap \{v_1, \ldots, v_{q-1}\}| \frac{|S| - 1}{|S|} \beta_S. \tag{7}
$$

It is now easy to lower-bound the coefficient of β_S:

$$|S \cap \{v_1, \ldots, v_{q-1}\}| \frac{|S|-1}{|S|} \geq |S \cap \{v_1, \ldots, v_{q-1}\}| - 1 \geq |\gamma(S) \cap p|,$$

and hence, with (6) and (7), it follows that $\chi(p) \geq w(p)$ as wanted. Finally, note that

$$\chi(V) = \sum_{v \in T} \alpha_v + \sum_{v \in R} y_v + \sum_{S} |S| \frac{|S|-1}{|S|} \beta_S,$$

which is exactly the dual objective value, and hence $\chi(V) = v(V)$, and χ is in the core. □

Similarly to (P$_2$), Lemma 3.4 is a direct consequence of Theorem 1.4. The proof we provided is constructive and gives rise to an efficient algorithm for computing core allocations. Finally note that Lemmata 3.3 and 3.4 prove Theorem 1.6.

Lemma 3.6. *There are instances for which the **C** is non empty, still the integrality gap of both* (P$_2$) *and* (P$_3$) *is bigger than 1.*

Proof. Consider terminals t_1, t_2 and facilitators f_1, f_2, f_3, f_4, with edges connecting them (along with weights) as seen in Figure 2. One of the optimal solutions is the path t_1f_2, f_2f_4, f_4t_2 of value 11. A

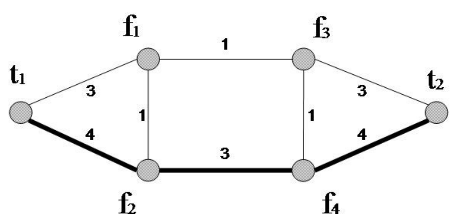

Figure 2: The optimal contract.

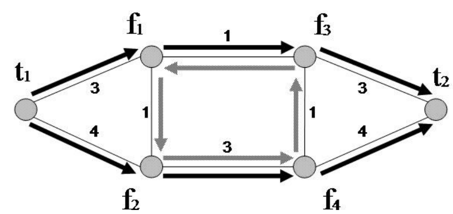

Figure 3: The fractional LP solution.

core assignment would give $x_{t_1} = x_{t_2} = \frac{11}{2}$, and 0 to all facilitators. This can be seen to be in the core since contracts are always paths connecting t_1, t_2, and none of them has cost more than what both terminals earn together.

Finally, we argue how to fool both (P$_2$) and (P$_3$). For this we invent three flows; the path t_1f_2, f_2f_4, f_4t_2, the path t_1f_1, f_1f_3, f_3t_2 and the cycle $f_1f_2, f_2f_4, f_4f_3, f_3f_1$ (depicted in Figure 3) all with value 1/2. A claim that can be easily checked is that the proposed values satisfy both LPs, while the objective value in both cases is 12, which is strictly bigger than the integral optimal. □

3.2.3. The two flow formulations are incomparable

In this section we study the performance of (P$_2$) and (P$_3$) on two different instances of GNB with non-empty core, for which we show that exactly one of the relaxations has integrality gap 1. This shows that the two formulations are incomparable.

Lemma 3.7. *There are instances for which the **C** is non empty, still the integrality gap of* (P$_2$) *is bigger than 1.*

Proof. We consider an instance on terminals t_1, t_2, t_3 and facilitators f_1, f_2, f_3, as depicted in Figure 4. Edges between facilitators have weight 2, while edges between facilitators and terminals have weight 1. The optimal contract has weight 6, connecting two of the terminals and utilizing all three facilitators (depicted with the solid edges in Figure 4).

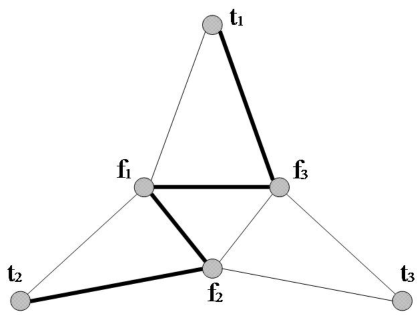

Figure 4: The optimal contract.

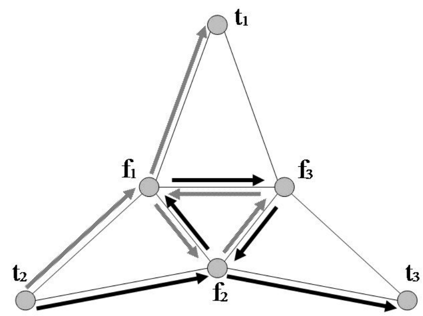

Figure 5: Part of the fractional LP solution.

Now we define an outcome in the **C** by setting $x_{t_i} = 0$ and $x_{f_i} = 2$, $i = 1, 2, 3$. Note that the sum of all earnings is 6, exactly as the value of the optimal contract. Any other valid contract will use either one or two facilitators. If only one is used, then the optimal contract will have value 2, exactly as much as the facilitator earns. If the contract has two facilitators, then the contract has value 4, which is again as much as the participating agents earn. Overall, this shows that $x \in \mathbf{C}$.

Our next claim is that the (P$_2$) has value strictly larger than 6, introducing an integrality gap strictly more than 1. For this we define the following fractional flow: from every terminal we send two equal flows of value $\frac{1}{6}$ to the other two terminals by utilizing all three facilitators, once with a clockwise and once with a counterclockwise flow. The two flows leaving terminal t_2 are shown in Figure 5. Next, we set the value of each edge to be its accumulated flow we defined above, and we claim that this is a feasible to (P$_2$).

Indeed, from every terminal we have a total of $\frac{2}{6}$ incoming and $\frac{2}{6}$ outgoing flow, and hence the first constraint is satisfied. Next, we observe that for every facilitator we have a total outgoing flow of $\frac{6}{6}$, and so the second constraint is satisfied as well. The third constraint requires flow conservation which is obvious from our construction. Finally, for the third constraint we need to consider subsets of facilitators of size either 2 or 3. In both cases we have total outgoing flow $\frac{2}{6}$ which is no less than the total flow on every internal edge. This concludes feasibility. The lemma then follows by observing that the value of (P$_2$) is 8. □

Lemma 3.8. *There are instances for which the **C** is non empty, still the integrality gap of* (P$_3$) *is bigger than 1.*

Proof. Consider two terminals t_1, t_2 and facilitators f_1, f_2, f_3, as seen in Figure 6. Edges between the facilitators have weight 1, terminal-facilitator edges have weight 0, and finally the unique terminal-terminal edge has weight 2. The optimal solution (which is not unique) is the contract connecting directly t_1 with t_2 (and is depicted with the solid edge in Figure 6, along with the weight on all edges.)

Next we set $x_{t_2} = x_{t_1} = 1$, and $x_{f_1} = x_{f_2} = x_{f_3} = 0$, and we claim that $x \in \mathbf{C}$, something that can be easily verified. To prove the lemma, we just need to show a feasible LP solution that exceeds the value 2.

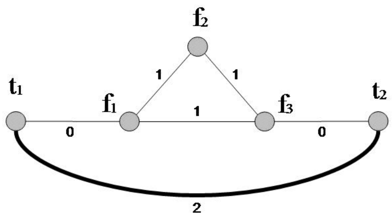

Figure 6: The optimal contract.

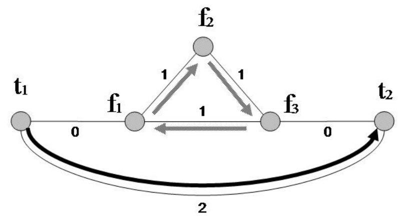

Figure 7: The fractional LP solution.

To that end, we propose an LP solution that consists of two flows, as shown in Figure 7. The direct flow connecting the terminals has value 1, while the cyclical flow within the facilitators gives value $2/3$ to each directed edge. The latter assignment satisfies the subtour constraint tightly, as well as all other LP constraints. Still the value of the objective is 4. □

One may observe that the instance of Figure 6 fooling (P_3) does not elude from the last constraints of (P_2), which together with our similar observation about the instance of Figure 4 and (P_3), shows that the relaxations (P_2) and (P_3) are actually incomparable. Since both relaxations are defined over the same set of variables, the possibility of characterizing non empty **C** with the intersection of the polytopes cannot be precluded in principle. Still Lemma 3.6 shows that this is indeed the case.

4. Conclusion

In this paper, we introduce the class of *generalized bargaining* games as a natural extension of *network bargaining*. We show that many of the known results for network bargaining extend to the new setting. For example, we show that an instance I of GNB has a balanced outcome whenever it has a stable one. We define a cooperative game $\Gamma(I)$ for every GNB instance I and present an LP (P_1) that has an integral optimal solution iff the core of $\Gamma(I)$ is non-empty.

Several interesting open questions remain: (1) In the case where the set of contracts is implicitly given as all terminal-terminal paths in the underlying graph, is it hard to solve (P_1) efficiently? (2) In the same setting, can we give a good characterization of the class of graphs (possibly via excluded minors) that have stable solutions?

References

[1] J. Nash, The bargaining problem, Econometrica 18 (1950) 155–162.

[2] J. M. Kleinberg, É. Tardos, Balanced outcomes in social exchange networks, in: Proceedings, ACM Symposium on Theory of Computing, 2008, pp. 295–304.

[3] M. Bateni, M. Hajiaghayi, N. Immorlica, H. Mahini, The cooperative game theory foundations of network bargaining games, in: Proceedings, Int. Colloq. on Automata, Languages and Processing, 2010, pp. 67–78.

[4] U. Faigle, W. Kern, J. Kuipers, An efficient algorithm for nucleolus and prekernel computation in some classes of TU-games, Technical Report 1464, U. of Twente, 1998.

[5] L. S. Shapley, M. Shubik, The assignment game : the core, International Journal of Game Theory 1 (1971) 111–130.

[6] D. Granot, F. Granot, On some network flow games, Math. Oper. Res. 17 (1992) 792–841.

[7] X. Deng, T. Ibaraki, H. Nagamochi, Algorithmic aspects of the core of combinatorial optimization games, Math. Oper. Res. 24 (1999) 751–766.

[8] X. Deng, Q. Fang, Algorithmic cooperative game theory, in: Pareto Optimality, Game Theory And Equilibria, Springer Optimization and Its Applications, Volume 17, Springer, 2008, pp. 159–185.

[9] G. Chalkiadakis, E. Elkind, M. Wooldridge, Computational Aspects of Cooperative Game Theory, Morgan & Claypool Publishers, 2011.

[10] H. Meinhardt, An lp approach to compute the pre-kernel for cooperative games, Computers & OR 33 (2006) 535–557.

[11] V. Conitzer, T. Sandholm, Complexity of constructing solutions in the core based on synergies among coalitions, Artif. Intell. 170 (2006) 607–619.

[12] U. Faigle, W. Kern, S. P. Fekete, W. Hochstättler, On the complexity of testing membership in the core of min-cost spanning tree games, Int. J. Game Theory 26 (1997) 361–366.

[13] M. Grötschel, L. Lovász, A. Schrijver, The ellipsoid method and its consequences in combinatorial optimization, Combinatorica 1 (1981) 169–197.

[14] A. Schrijver, Combinatorial optimization, Springer, New York, 2003.