Mathematical Programming (2020) 183:555–581
https://doi.org/10.1007/s10107-020-01483-4

FULL LENGTH PAPER

Series B

Computing the nucleolus of weighted cooperative matching games in polynomial time

Jochen Könemann[1] · Kanstantsin Pashkovich[2] · Justin Toth[1]

Received: 4 June 2019 / Accepted: 14 February 2020 / Published online: 28 February 2020

Abstract
We provide an efficient algorithm for computing the nucleolus for an instance of a weighted cooperative matching game. This resolves a long-standing open question posed in Faigle (Math Programm, 83: 555–569, 1998).

Keywords Combinatorial Optimization · Algorithmic Game Theory · Matchings

Mathematics Subject Classification 90C27 · 91A12

1 Introduction

Imagine a network of players that form partnerships to generate value. For example, a tennis league pairing players to play exhibition matches [3], or people making trades in an exchange network [46]. These are examples of what are called *matching games*. In a (weighted) matching game, we are given a graph $G = (V, E)$, weights $w : E \to \mathbb{R}_{\geq 0}$, the player set is the set V of nodes of G, and $w(uv)$ denotes the value earned when

This work was done in part while the second author was visiting the Simons Institute for the Theory of Computing. Supported by DIMACS/Simons Collaboration on Bridging Continuous and Discrete Optimization through NSF grant #CCF-1740425.
We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC). Cette recherche a été financée par le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG).

✉ Justin Toth
wjtoth@uwaterloo.ca

Jochen Könemann
jochen@uwaterloo.ca

Kanstantsin Pashkovich
kpashkov@uottawa.ca

1 University of Waterloo, Waterloo, ON N2L 3G1, Canada

2 University of Ottawa, Ottawa, ON K1N 6N5, Canada

u and v collaborate. Each coalition $S \subseteq V$ is assigned a *value* $\nu(S)$ so that $\nu(S)$ is equal to the value of a maximum weight matching in the induced subgraph $G[S]$. The special case of matching games where $w = \mathbb{1}$ is the all-ones vector, and G is bipartite is called an *assignment game*. Assignment games were introduced in a classical paper by Shapley and Shubik [46], and were later generalized to matching games on general graphs by Denget al. [11].

We are interested in what a fair redistribution of the total value $\nu(V)$ to the players in the network looks like. The field of *cooperative game theory* gives us the language to make this question formal. A vector $x \in \mathbb{R}^V$ is called an efficient *allocation* if $x(V) = \nu(V)$ (where we use $x(V)$ as a short-hand for $\sum_{i \in V} x(i)$ as usual). An allocation is called *individually rational* if $x(i) \geq \nu(\{i\})$ for all $i \in V$. Given such an allocation, we let $x(S) - \nu(S)$ be the *excess* of coalition $S \subseteq V$. This quantity can be thought of as a measure of the satisfaction of coalition S. A fair allocation should be individually rational and maximize the bottleneck excess, i.e. maximize the minimum excess, and this can be accomplished by an LP:

$$\begin{aligned} \max \quad & \varepsilon && && (P)\\ \text{s.t.} \quad & x(S) \geq \nu(S) + \varepsilon && \text{for all} \quad S \subseteq V \\ & x(V) = \nu(V) \\ & x(i) \geq \nu(\{i\}) && \text{for all} \quad i \in V. \end{aligned}$$

Let ε^* be the optimum value of (P), and define $P(\varepsilon^*)$ to be the set of allocations x such that (x, ε^*) is feasible for (P). The set $P(\varepsilon^*)$ is known as the *leastcore* [38] of the given cooperative game, and the special case when $\varepsilon^* = 0$, $P(0)$ is the well-known *core* [25] of (V, ν). Intuitively, allocations in the core describe payoffs in which no coalition of players could profitably deviate from the *grand coalition* V.

Why stop at maximizing the bottleneck excess? Consider an allocation which, subject to maximizing the smallest excess, maximizes the second smallest excess, and subject to that maximizes the third smallest excess, and so on. This process of successively optimizing the excess of the worst-off coalitions yields our primary object of interest, the *nucleolus*. For an allocation $x \in \mathbb{R}^V$, let $\theta(x) \in \mathbb{R}^{2^V - 2}$ be the vector obtained by sorting the list of excess values $x(S) - \nu(S)$ for any $\varnothing \neq S \subset V$ in non-decreasing order.[1] The *nucleolus*, denoted $\eta(V, \nu)$ and defined by Schmeidler [44], is the unique allocation that lexicographically maximizes $\theta(x)$:

$$\eta(V, \nu) := \arg \operatorname{lex} \max\{\theta(x) : x \in P(\varepsilon^*)\}.$$

We refer the reader to Fig. 1 for an example instance of the weighted matching game with its nucleolus. We now have sufficient terminology to state our main result:

[1] It is common within the literature, for instance in [30], to exclude the coalitions for $S = \varnothing$ and $S = V$ in the definition of the nucleolus. On the other hand, one could also consider the definition of the nucleolus with all possible coalitions, including $S = \varnothing$ and $S = V$. We note that the two definitions of the nucleolus are equivalent in all instances of matching games except for the trivial instance of a graph consisting of two nodes joined by a single edge.

Theorem 1 *Given a graph $G = (V, E)$ and weights $w : E \to \mathbb{R}$, the nucleolus $\eta(V, \nu)$ of the corresponding weighted matching game can be computed in polynomial time.*

Despite its intricate definition the concept of the nucleolus is surprisingly ancient. Its history can be traced back to a discussion on bankruptcy division in the Babylonian Talmud [1]. Modern research interest in the nucleolus stems not only from its geometric beauty [38], or several practical applications (e.g., see [5,37]), but from the strange way problems of computing the nucleolus fall in the complexity landscape, seeming to straddle the NP vs P boundary.

Beyond being one of the most fundamental problems in combinatorial optimization, starting with the founding work of Kuhn on the Hungarian method for the assignment problem [34], matching problems have historically teetered on the cusp of hardness. For example, prior to Edmonds' celebrated Blossom Algorithm [13,14] it was not clear whether Maximum Matching belonged to P. For another example, until Rothvoß' landmark result [43] it was thought that the matching polytope could potentially have sub-exponential extension complexity. In cooperative game theory, matchings live up to their historical pedigree of representing a challenging problem class. The long standing open problem in this area was whether the nucleolus of a weighted matching game instance can be computed in polynomial time. The concept of the nucleolus has been known since [44], and the question was posed as an important open problem in multiple papers. In Faigle et al. [17] mention the problem in their work on the nucleon, a multiplicative-error analog to the nucleolus which they show is polynomial time computable. Kern and Paulusma state the question of computing the nucleolus for general matching games as an important open problem in [30]. In Deng and Fang [9] conjectured this problem to be NP-hard, and in Biró et al. [4] reaffirmed this problem as an interesting open question. Theorem 1 settles the question, providing a polynomial-time algorithm to compute the nucleolus of a general instance of a weighted cooperative matching game.

Our approach to proving Theorem 1 is to provide a compact description of each feasible region polytope in a hierarchical sequence of linear programs described in the well-known *Maschler scheme* [33,38] for computing the nucleolus. While there are a linear number of LPs in the sequence, their naive implementation requires an exponential number of constraints. The base LP for this approach is the least core. It is known how to separate over the least core, but the challenge lies in solving all successive linear programs in the sequence. Previous results in [30,41] made use of the *unweighted* nature of node-weighted instances of matching games, and were able to employ the *Edmonds-Gallai* structure theorem to derive compact formulations for the LPs in Maschler's hierarchy. We do not know how to extend this line of work beyond node-weighted instances.

In our work, we identify a minimal family of tight excess constraints of (P) corresponding to coalitions whose vertices are saturated by so called *universal matchings*. A matching is universal if it saturates a coalition of vertices that is tight for all allocations in the least core. As we will show, universal matchings are the optimal matchings for carefully chosen cost functions derived from so called *universal allocations* (see Sect. 2.3). From here now, we rely on well-known characterizations of extreme points of matching polyhedra, and the fact that these are defined by laminar families of blossom constraints. Ultimately, the above allows us to obtain a decomposition of the input graph into the edges on either the inside or outside of blossoms. The structure of the associated optimum face of the matching polytope (e.g., see Schrijver [45]) elucidates the structure of excess over all coalitions in the matching game. Our proof uses a critical insight into the symmetric nature of exchange on the nodes of a blossom as we move between leastcore allocations (see Lemma 6). We present the details of the leastcore LP in Sect. 2. We show how this formulation can be used in Maschler's framework to compute the nucleolus in Sect. 3.

Prior to our work, the nucleolus was known to be polynomial-time computable only in structured instances of the matching game. Solymosi and Raghavan [47] showed how to compute the nucleolus in an (unweighted) assignment game instance in polynomial time. Kern and Paulusma [30] later provided an efficient algorithm to compute the nucleolus in general unweighted matching game instances. Paulusma [41] extended the work in [30] and gave an efficient algorithm to compute the nucleolus in matching games where edge weights are induced by node potentials. Farczadi [22] finally extended Paulusma's framework further using the concept of *extendible allocations*. We note also that it is easy to compute the nucleolus in weighted instances of the matching game with non-empty core. For such instances, the leastcore has a simple compact description that does not include constraints for coalitions of size greater than 2. Thus it is relatively straightforward to adapt the iterative algorithm of Maschler [38] to a polynomial-time algorithm for computing the nucleolus (e.g., see [22, Chapter 2.3] for the details, Sect. 1.3 for an overview). Such an algorithm for computing the nucleolus would rely on the ellipsoid method. Circumventing this, Biró, Kern, and Paulusma [3] gave a combinatorial algorithm for computing the nucleolus of matching games with non-empty core.

1.1 Related work

In a manner analogous to how we have defined matching games, a wide variety *combinatorial optimization games* can be defined [11]. In such games, the value of a coalition S of players is succinctly given as the optimal solution to an underlying combinatorial optimization problem. It is natural to conjecture that the complexity of computing the nucleolus in an instance of such a game would fall in lock-step with the complexity of the underlying problem. Surprisingly this is not the case. For instance, computing the nucleolus is known to be NP-hard for network flow games [10], weighted threshold games [15], and spanning tree games [18,21]. On the other hand, polynomial time algorithms are known for finding the nucleolus of special cases of flow games, certain classes of matching games, fractional matching games, and convex games [2,6,10,19,22,26,27,30,35,39,41,42,47].

One application of cooperative matching games is to network bargaining [12,49], where one considers a population of players that are interconnected by an underlying social network. Players engage in profitable *partnerships* – each player with at most one of its neighbours. The profit generated through a partnership then needs to be shared between the participating players in an equitable way. Cook and Yamagishi [7] first proposed an elegant profit-sharing model that not only generalizes Nash's famous 2-player bargaining solution [40], but also validates empirical findings from the lab setting. Cook and Yamagishi's model defines a so called *outside option* for each player v that essentially denotes the largest share that v could demand in a partnership with any of its neighbours. An outcome is called *stable* if each player receives a profit share that is at least her outside option, and an outcome is *balanced* if value generated in excess of the sum of the outside options of the involved players is shared evenly. Kleinberg and Tardos [31] showed that an instance of network bargaining has a balanced outcome if and only if it has a stable outcome. Farczadi, Georgiou, and Könemann [23] extended this result to network bargaining games with general vertex capacities. Bateni et al. [2] further noted that stable outcomes for network bargaining correspond to elements of the core of the underlying cooperative matching game, and that balanced outcomes correspond to elements in the intersection of core and prekernel.

It is well-known that the prekernel of a cooperative game may be non-convex and that it may even be disconnected [33,48]. Despite this, Faigle, Kern and Kuipers [19] showed how to compute a point in the intersection of prekernel and leastcore in polynomial time under the reasonable assumption that the game has a polynomial time oracle to compute the minimum excess coalition for a given allocation. Later the same authors [20] refined their result to computing a point in the intersection of the core and lexicographic kernel, a set which is also known to contain the nucleolus. Bateni et al. posed as an open question the existence of an efficiently computable, balanced and *unique* way of sharing the profit. It is well-known that the nucleolus lies in the intersection of core and prekernel (if these are non-empty) and is always unique [44]. Theorem 1 therefore resolves the latter open question left in [2].

Connections have also been made between matching games and stable matchings. Allocations in the core of the corresponding matching game have been found to be in 1-to-1 correspondence with stable matchings with payments, a variant of the Gale and Shapley's [24] famous stable marriage problem. See Koopmans and Beckmann [32], Shapley and Shubik [46], Eriksson and Karlander [16], and Biró, Kern, and Paulusma [3] for details.

1.2 Leastcore and core of matching games

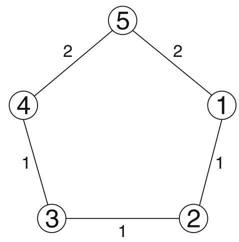
It is straightforward to see that (P) can be rewritten equivalently as

$$\begin{aligned} \max\ & \varepsilon && && (P_1)\\ \text{s.t. } & x(M) \geq w(M) + \varepsilon && \text{for all} \quad M \in \mathcal{M} \\ & x(V) = \nu(G) \\ & x \geq \mathbb{0}\,, \end{aligned}$$

where $\mathcal{M}$ is the set of all matchings M on G, and $x(M)$ is a shorthand for $x(V(M))$.

The separation problem for the linear program (P_1) can be reduced to finding a maximum weight matching in the graph G with edge weights $w(uv) - x(uv)$, $uv \in E$ (where we use $x(uv)$ as a shorthand for $x(u) + x(v)$). Since the maximum weight

Fig. 1 Matching game with empty core



matching can be found in polynomial time [13], we know that the linear program (P_1) can be solved in polynomial time as well [29].

We use ε_1 to denote the optimal value of (P_1) and $P_1(\varepsilon_1)$ for the set of allocations x such that (x, ε_1) is feasible for the leastcore linear program (P_1). In general, for a value ε and a linear program (Q) on variables in $\mathbb{R}^V \times \mathbb{R}$ we denote by $Q(\varepsilon)$ the set $\{x \in \mathbb{R}^V : (x, \varepsilon) \text{ is feasible for } (Q)\}$.

Note that $\varepsilon_1 \leq 0$. Indeed, $\varepsilon \leq 0$ in any feasible solution (x, ε) to (P_1) as otherwise $x(M)$ would need to exceed $w(M)$ for all matchings M. In particular this would also hold for a maximum weight matching in G, implying that $x(V) > \nu(G)$. If $\varepsilon_1 = 0$ then the core of the cooperative matching game is non-empty. One can see that $\varepsilon_1 = 0$ if and only if the value of a maximum weight matching in G with weights w equals the value of a maximum weight fractional matching. This follows since $x \in P_1(\varepsilon_1)$ is a fractional weighted node cover of value $\nu(G)$ when $\varepsilon_1 = 0$. When $\varepsilon_1 < 0$, the cooperative matching game has an empty core.

Example of a Matching Game With Empty Core Consider the example in Fig. 1. This graph $G = (V, E)$ is a 5-cycle with two adjacent edges 15 and 45 of weight 2, and the remaining three edges of weight 1. Since the maximum weight matching value is $\nu(G) = 3$, but the maximum weight fractional matching value is $7/2$, the core of this game is empty. The allocation x^* defined by

$$x^*(1) = x^*(2) = x^*(3) = x^*(4) = \frac{2}{5} \quad \text{and} \quad x^*(5) = \frac{7}{5}$$

lies in the leastcore. Each edge has the same excess, $-1/5$, and any coalition of four vertices yields a minimum excess coalition with excess $-2/5$. Hence the leastcore value of this game is $\varepsilon_1 = -2/5$.

In fact, we can see that x^* is the nucleolus of this game. To certify this we can use the result of Schmeidler [44] that the nucleolus lies in the intersection of the leastcore and the prekernel. For this example, the prekernel condition is that for all $i \neq j \in V$,

$$\max_{S \subseteq V \setminus \{j\}} x(S \cup \{i\}) - \nu(S \cup \{i\}) = \max_{S \subseteq V \setminus \{i\}} x(S \cup \{j\}) - \nu(S \cup \{j\}).$$

This condition reduces to the condition that the excess values of non-adjacent edges are equal. Since G is an odd cycle, this implies that all edges have equal excess, i.e.

$$\text{excess}(x, 12) = \text{excess}(x, 23) = \text{excess}(x, 34) = \text{excess}(x, 45) = \text{excess}(x, 15).$$

Combining the four equations above with the leastcore condition that $x(V) = \nu(G)$ we obtain a system of equations with the unique solution x^*. Hence the intersection of the leastcore and prekernel is precisely $\{x^*\}$, and so by Schmeidler, x^* is the nucleolus.

In this paper, we assume that the cooperative matching game (G, w) has an empty core, as computing the nucleolus is otherwise well-known to be solveable in polynomial time [41].

1.3 Maschler's scheme

As discussed, our approach to proving Theorem 1 relies on Maschler's scheme. The scheme requires us to solve a linear number of LPs: $\{(P_j)\}_{j\geq 1}$ that we now define. (P_1) is the leastcore LP that we have already seen in Sect. 1.2. LPs (P_j) for $j \geq 2$ are defined inductively. Crucial in their definition is the notion of *fixed* coalitions that we introduce first. For a polyhedron $Q \subseteq \mathbb{R}^V$ we denote by $\mathrm{Fix}(Q)$ the collection of sets $S \subseteq V$ such that $x(S)$ is constant over the polyhedron Q, i.e.

$$\mathrm{Fix}(Q) := \{S \subseteq V \ : \quad x(S) = x'(S) \quad \text{for all} \quad x, x' \in Q\}\,.$$

With this we are now ready to state LP (P_j) for $j \geq 2$:

$$\begin{aligned} \max \quad & \varepsilon & & \qquad (P_j) \\ \text{s.t.} \quad & x(S) - \nu(S) \geq \varepsilon & & \text{for all} \quad S \subset V,\ S \notin \mathrm{Fix}(P_{j-1}(\varepsilon_{j-1})) \\ & \qquad\quad x \in P_{j-1}(\varepsilon_{j-1}), & & \end{aligned}$$

where ε_j is the optimal value of the linear program (P_j). Let j^* be the minimum number j such that $P_j(\varepsilon_j)$ contains a single point. This point is the nucleolus of the game [8]. It is well-known [38] that $P_{j-1}(\varepsilon_{j-1}) \subset P_j(\varepsilon_j)$ and $\varepsilon_{j-1} < \varepsilon_j$ for all j. Since the dimension of the polytope describing feasible solutions of (P_j) decreases in every round until the dimension becomes zero, we have $j^* \leq |V|$ [38] [41, Pages 20-24].

Therefore, in order to find the nucleolus of the cooperative matching game efficiently it suffices to solve each linear program (P_j), $j = 1, \ldots, j^*$ in polynomial time. We accomplish this by providing polynomial-size formulations for (P_j) for all $j \geq 1$.

In Sect. 2 we introduce the concept of *universal matchings* which are fundamental to our approach, and give a compact formulation for the first linear program in Maschler's Scheme, the leastcore. We also present our main technical lemma, Lemma 6, which provides a crucial symmetry condition on the values allocations can take over the vertices of blossoms in the graph decomposition we use to describe the compact formulation. In Sect. 3 we describe the successive linear programs in Maschler's Scheme and provide a compact formulation for each one in a matching game. In Sect. 4 we describe a leastcore separation oracle for b-matching games.

2 Leastcore formulation

In this section we provide a polynomial-size description of (P_1). It will be useful to define a notation for *excess*: for any $x \in P_1(\varepsilon_1)$ and $M \in \mathcal{M}$ let $\text{excess}(x, M) := x(M) - w(M)$.

2.1 Universal matchings, universal allocations

For each $x \in P_1(\varepsilon_1)$ we say that a matching $M \in \mathcal{M}$ is an x-tight matching whenever $\text{excess}(x, M) = \varepsilon_1$. We denote by $\mathcal{M}^x$ the set of x-tight matchings.

A *universal matching* $M \in \mathcal{M}$ is a matching which is x-tight for all $x \in P_1(\varepsilon_1)$. We denote the set of universal matchings in G by $\mathcal{M}_{uni}$. A *universal allocation* $x^* \in P_1(\varepsilon_1)$ is a leastcore point whose x^*-tight matchings are precisely the set of universal matchings, i.e. $\mathcal{M}^{x^*} = \mathcal{M}_{uni}$.

Lemma 1 *There exists a universal allocation $x^* \in P_1(\varepsilon_1)$.*

Proof Indeed, it is straightforward to show that every x^* in the relative interior (see [50, Lemma 2.9(ii)]) of $P_1(\varepsilon_1)$ is a universal allocation. If the relative interior is empty then $P_1(\varepsilon_1)$ is a single point, which trivially contains a universal allocation. For the sake of completeness we now provide a combinatorial proof.

Let $\mathcal{M}^x$ denote the set of x-tight matchings for some $x \in P_1(\varepsilon_1)$. That is $\mathcal{M}^x := \{M \in \mathcal{M} : \text{excess}(x, M) = \varepsilon_1\}$. We claim that there exists $x^* \in P_1(\varepsilon_1)$ such that

$$\mathcal{M}^{x^*} = \bigcap_{x \in P_1(\varepsilon_1)} \mathcal{M}^x .$$

Let $x^* \in P_1(\varepsilon_1)$ be chosen to minimize $|\mathcal{M}^{x^*}|$. Suppose for a contradiction that x^* is not universal. Then there exists $x \in P_1(\varepsilon_1)$ and $M \in \mathcal{M}^{x^*} \setminus \mathcal{M}^x$. Let $\bar{x} := \frac{1}{2}(x^* + x)$. Since $P_1(\varepsilon_1)$ is a convex set, $\bar{x} \in P_1(\varepsilon_1)$. Furthermore, $\mathcal{M}^{\bar{x}} = \mathcal{M}^{x^*} \cap \mathcal{M}^x$, thus $\mathcal{M}^{\bar{x}} \subseteq \mathcal{M}^{x^*} \setminus \{M\}$ contradicting the minimality of $|\mathcal{M}^{x^*}|$. □

Lemma 2 *A universal allocation $x^* \in P_1(\varepsilon_1)$ can be computed in polynomial time.*

Proof A point x^* in the relative interior of $P_1(\varepsilon_1)$ can be found in polynomial time using the ellipsoid method (Theorem 6.5.5 [28]). Since any allocation x^* from the relative interior of $P_1(\varepsilon_1)$ is a universal allocation, this implies the statement of the lemma. In Sect. 2.2 we provide a self-contained proof of this Lemma. □

2.2 Computing a universal allocation

For the sake of completeness, we prove Lemma 2 by providing an algorithm for computing a universal allocation x^*. Consider the following algorithm:

1: Compute $x \in P_1(\varepsilon_1)$.
2: $F \leftarrow$ face of matching polytope maximizing function defined by $w(uv) - x(u) - x(v)$, $uv \in E$.
3: Compute $M \in \mathcal{M}$, such that $\chi(M)$ is a vertex of F.
4: Initialize $K \leftarrow \varnothing$.
5: **loop**
6: $\quad y \leftarrow \arg\max\{y(M) : y \in P_1(\varepsilon_1)\}$.
7: $\quad$ **if** $y(M) > x(M)$ **then**
8: $\qquad x \leftarrow \frac{1}{2}(x + y)$
9: $\qquad$ Let F be the face of the matching polytope maximizing the linear function defined by weights $w(uv) - x(u) - x(v)$, $uv \in E$.
10: $\quad$ **else** $\{M$ is a universal matching$\}$
11: $\qquad K \leftarrow K \cup \{\chi(M)\}$
12: $\qquad$ Compute M, such that $\chi(M) \notin \operatorname{span}(K)$ is a vertex of F. If no such M exists, break Loop.
13: $\quad$ **end if**
14: **end loop**
15: **return** x.

Theorem 2 *The allocation x^* returned in Step 15 of the above algorithm is a universal allocation.*

Proof Suppose that x^* is not a universal allocation, then there exists a non-universal x^*-tight matching M; i.e., there exists $y \in P_1(\varepsilon_1)$ such that M is not y-tight.

Since $\chi(M) \in \operatorname{span}(K)$, we know that

$$\chi(M) = \sum_{j=1}^{p} \alpha_j \chi(M_j),$$

for some $\alpha_j \in \mathbb{R}$, $j \in [p]$, where $K = \{\chi(M_1), \ldots, \chi(M_p)\}$. Recall, that all matchings $M_1, \ldots, M_p$ are universal matchings, hence

$$x^*(M_j) = y(M_j) = \varepsilon_1 + w(M_j)$$

for all $j \in [p]$. It follows that

$$\varepsilon_1 + w(M) = x^*(M) = \sum_{j=1}^{p} \alpha_j x^*(M_j) = \sum_{j=1}^{p} \alpha_j y(M_j) = y(M),$$

contradicting the fact that M is x^*-tight but not y-tight. □

Theorem 3 *The algorithm terminates in polynomial time.*

Proof Indeed, if $y(M) > x(M)$ in Step 7 of the algorithm then Step 9 leads to a dimension reduction for face F. This dimension reduction follows since the vertices of the old F correspond to the set of x-tight matchings, whereas the vertices of the new F correspond to the set of $\frac{1}{2}(x+y)$-tight matchings. Since $y(M) > x(M)$, the $\frac{1}{2}(x+y)$-tight matchings are a strict subset of the x-tight matchings, and hence the new F is a strict subface of the old F.

If on the other hand $y(M) \leq x(M)$ then the dimension of K as defined in Step 11 increases because of our choice of M in Step 12. The upper bound on the dimensions of the linear space span(K) and the face F is $|E|$, while the lower bound on their dimensions is 0.

Since we can separate over $P_1(\varepsilon_1)$ in polynomial time we can solve the optimization problem in Step 3(a) in polynomial time. Moreover, since we can separate over F in polynomial time, we can decide if F has a vertex outside the linear space span(K) in polynomial time by solving two optimization problems for each of the (at most $|E|$) equations defining span(K): maximizing and minimizing the corresponding (normal to span(K)) vector over F, and checking if both optimal values are zero. Since each step of the algorithm runs in polynomial time, this implies that the algorithm terminates in polynomial time. Hence, the statement of the theorem follows. $\square$

Given a non-universal allocation x and a universal allocation x^*, we observe that $\mathcal{M}^{x^*} \subset \mathcal{M}^x$ and so $\theta(x^*)$ is strictly lexicographically greater than $\theta(x)$. Thus the nucleolus is a universal allocation. We emphasize that $\mathcal{M}^{x^*} = \mathcal{M}_{uni}$ is invariant under the (not necessarily unique) choice of universal allocations x^*. Henceforth we fix a universal allocation $x^* \in P_1(\varepsilon_1)$.

2.3 Description for convex hull of universal matchings.

By the definition of universal allocation x^*, a matching M is universal if and only if it is x^*-tight. Thus, M is a universal matching if and only if its characteristic vector lies in the optimal face of the matching polytope corresponding to (the maximization of) the linear objective function assigning weight $-\operatorname{excess}(x^*, uv) = w(uv) - x^*(u) - x^*(v)$ to each edge $uv \in E$. Let $\mathcal{O}$ be the set of node sets $S \subseteq V$ such that $|S| \geq 3$, $|S|$ is odd. Edmonds [13] gave a linear description of the matching polytope of G as the set of $y \in \mathbb{R}^E$ satisfying:

$$\begin{aligned} y(\delta(v)) &\leq 1 && \text{for all} \quad v \in V \\ y(E(S)) &\leq (|S|-1)/2 && \text{for all} \quad S \in \mathcal{O} \\ y &\geq 0\,. \end{aligned}$$

Thus, a matching $M \in \mathcal{M}$ is universal if and only if it satisfies the constraints

$$\begin{aligned} M \cap \delta(v) &= 1 && \text{for all} \quad v \in W \\ M \cap E(S) &= (|S|-1)/2 && \text{for all} \quad S \in \mathcal{L} \\ M \cap \{e\} &= 0 && \text{for all} \quad e \in F\,, \end{aligned} \tag{1}$$

where W is some subset of V, $\mathcal{L}$ is a subset of $\mathcal{O}$, and F is a subset of E. Using an uncrossing argument, as in [36, Pages 141-150], we may assume that the collection of sets $\mathcal{L}$ is a laminar family of node sets; i.e., for any two distinct sets $S, T \in \mathcal{L}$, either

$S \cap T = \varnothing$ or $S \subseteq T$ or $T \subseteq S$. For completeness we include the most important lemma behind the uncrossing here.

Lemma 3 *Let $S, T \in \mathcal{O}$ such that $S \cap T \neq \varnothing$. Consider the face of the matching polytope of G defined by* $\operatorname{conv}\{\chi(M) : M \in \mathcal{M}_{uni}\}$. *Let y be a point in such a face. If*

$$y(E(S)) = \frac{|S|-1}{2} \quad \textit{and} \quad y(E(T)) = \frac{|T|-1}{2}$$

which is to say that S and T are tight constraints of the matching polytope with respect to y, then either

(i) the sets $S \cap T$, $S \cup T$ are tights sets in $\mathcal{O}$ and

$$\chi(E(S)) + \chi(E(T)) = \chi(E(S \cap T)) + \chi(E(S \cup T)),$$

(ii) or the sets $S \backslash T$, $T \backslash S$ are tight sets in $\mathcal{O}$, the degree constraints for each vertex in $S \cap T$ are tight, and

$$\chi(E(S)) + \chi(E(T)) = \chi(S \backslash T) + \chi(T \backslash S) + \sum_{v \in S \cap T} \chi(\delta(v)).$$

Proof First suppose that $|S \cap T|$ is odd. In this case we will show (i). We have that $|S \cup T|$ is odd, since $|S| + |T| = |S \cap T| + |S \cup T|$. Thus we have

$$\begin{aligned}
\frac{|S|-1}{2} + \frac{|T|-1}{2} &= y(E(S)) + y(E(T)) \\
&\leq y(E(S \cap T)) + y(E(S \cup T)) \qquad \text{(by supermodularity)} \\
&\leq \frac{|S \cap T|-1}{2} + \frac{|S \cup T|-1}{2} \\
&= \frac{|S|-1}{2} + \frac{|T|-1}{2}.
\end{aligned}$$

Therefore each inequality above holds with equality. By the second inequality holding at equality both $S \cap T$ and $S \cup T$ are tight. The first inequality holding at equality implies there are no edges between $S \backslash T$ and $T \backslash S$. Thus

$$\chi(S) + \chi(T) = \chi(S \cap T) + \chi(S \cup T).$$

Now suppose $|S \cap T|$ is even. In this case will show (ii). We have that $|S \backslash T|$ is odd and $|T \backslash S|$ is odd since $|S|$ and $|T|$ are odd. Thus we have

$$\begin{aligned}
\frac{|S|-1}{2} + \frac{|T|-1}{2} &= y(E(S)) + y(E(T)) \\
&\leq y(E(S \backslash T)) + y(E(T \backslash S)) + \sum_{v \in S \cap T} y(\delta(v))
\end{aligned}$$

$$\leq \frac{|S\backslash T|-1}{2}+\frac{|T\backslash S|-1}{2}+|S\cap T|$$
(by the degree and odd set constraints)
$$=\frac{|S|-1}{2}+\frac{|T|-1}{2}.$$

The second inequality holding at equality implies that $S\backslash T$, $T\backslash S$, and $\delta(v)$ are tight for each $v \in S\cap T$. The first inequality holding at equality implies there are no edges between $S\cap T$ and $V\backslash(S\cup T)$. Therefore

$$\chi(S)+\chi(T)=\chi(S\backslash T)+\chi(T\backslash S)+\sum_{v\in S\cap T}\chi(\delta(v)).$$

□

We will now show a crucial lemma demonstrating that no node is covered by every universal matching.

Lemma 4 *For every node $v \in V$ there exists $M \in \mathcal{M}_{uni}$ such that v is exposed by M. Hence, $W = \varnothing$.*

Proof Assume for a contradiction that there exists a node $v \in V$ such that $v \in W$.

First, note that there always exists a non-universal matching $M \in \mathcal{M}\backslash\mathcal{M}_{uni}$ since otherwise the empty matching would be universal, and thus

$$0 = x^*(\varnothing) = w(\varnothing) + \varepsilon_1,$$

implying that the core of the given matching game instance is non-empty.

Suppose first that there exists a node $u \in V$ exposed by some matching $M' \in \mathcal{M}_{uni}$ such that $x^*_u > 0$. Our strategy is to shift a small amount of allocation from u to v while preserving feasibility. Shifting by any sufficiently small positive δ will work. In what follows we explicitly define a value for δ for concreteness. We define

$$\delta_0 := \min\{\text{excess}(x^*, M) - \varepsilon_1 \;:\; \quad M \in \mathcal{M}\backslash\mathcal{M}_{uni}\}.$$

Recall that $\mathcal{M}_{uni}$ is the set of maximum weight matchings in G with respect to the node weights $w(uv) - x^*(uv)$, $uv \in E$, i.e. $\mathcal{M}_{uni}$ is the set of x^*-tight matchings. Moreover, recall that $\text{excess}(x^*, M) = \varepsilon_1$ for $M \in \mathcal{M}_{uni}$. Thus, we have $\delta_0 > 0$.

We define $\delta := \min\{\delta_0, x^*_u\} > 0$ and a new allocation x' as follows:

$$x'_r := \begin{cases} x^*_r + \delta, & \text{if } r = v \\ x^*_r - \delta, & \text{if } r = u \\ x^*_r, & \text{otherwise.} \end{cases}$$

Since all universal matchings contain v, the excess with respect to x' of any universal matching is no smaller than their excess with respect to x^*. Therefore, by our choice of δ, (x', ε_1) is a feasible, and hence optimal, solution for (P_1). But M' is not x'-tight, since M' covers v and exposes u. This contradicts that M' is a universal matching.

Now consider the other case: for all $u \in V$ if u is exposed by a universal matching then $x^*_u = 0$. Then, for every universal matching $M \in \mathcal{M}_{uni}$ we have

$$\varepsilon_1 = \text{excess}(x^*, M) = x^*(V) - w(M) = \nu(G) - w(M).$$

Since $\nu(G)$ is the maximum weight of a matching in G with respect to the weights w, we get that $\varepsilon_1 \geq 0$. Thus x^* is in the core, contradicting our assumption that the core is empty. □

2.4 Description of leastcore

We denote inclusion-wise maximal sets in the family $\mathcal{L}$ as $S^*_1, S^*_2, \ldots, S^*_k$. We define the edge set E^+ to be the set of edges in G such that at most one of its nodes is in S^*_i for every $i \in [k] := \{1, \ldots, k\}$, i.e.

$$E^+ := E \Big\backslash \left(\bigcup_{i=1}^{k} E(S^*_i) \right).$$

Lemma 5 *For every choice of $v_i \in S^*_i$, $i \in [k]$, there exists a universal matching $M \in \mathcal{M}_{uni}$ such that the node set covered by M is as follows*

$$\bigcup_{i=1}^{k} S^*_i \Big\backslash \{v_i\}.$$

Proof By Lemma 4, we know that for every $i \in [k]$ there exists a universal matching $M_{v_i} \in \mathcal{M}_{uni}$ such that v_i is exposed by M_{v_i}. Now, for every $i \in [k]$, let us define

$$M_i := E(S^*_i) \cap M_{v_i}.$$

Since M_i satisfies all laminar family constraints in $\mathcal{L}$ for subsets of S^*_i we have that

$$\bigcup_{i=1}^{k} M_i$$

is a matching satisfying all the constraints (1), and hence is a universal matching covering the desired nodes. □

Representative Universal Matching For each $i \in [k]$ fix a unique *representative* node $v^*_i \in S^*_i$. By Lemma 5, there exists a universal matching M^* covering precisely $\bigcup_{i \in [k]} S^*_i \backslash \{v^*_i\}$. For any $x \in P_1(\varepsilon_1)$ and $S \subseteq V$ we use diff(x, S) to denote

$$\text{diff}(x, S) := x(S) - x^*(S).$$

For single nodes we use the shorthand $\mathrm{diff}(x, v) = \mathrm{diff}(x, \{v\})$. We now prove the following crucial structure result on allocations in the leastcore.

Lemma 6 *For every leastcore allocation x, i.e. for every $x \in P_1(\varepsilon_1)$, we have that*

(i) for all $i \in [k]$, for all $u \in S_i^*$: $\quad \mathrm{diff}(x, u) = \mathrm{diff}(x, v_i^*)$,
(ii) for all $e \in E^+$: $\quad \mathrm{excess}(x, e) \geq 0$.

Proof Consider $u \in S_i^*$, and note that we may use Lemma 5 to choose a universal matching M_u covering precisely

$$S_i^* \Big\backslash \{u\} \cup \bigcup_{j \neq i} S_j^* \Big\backslash \{v_j^*\}.$$

Hence we have $V(M_u) \cup \{u\} = V(M^*) \cup \{v_i^*\}$, and since M^* and M_u are universal, $x(M^*) = x^*(M^*)$ and $x(M_u) = x^*(M_u)$. Using these observations we see that

$$\begin{aligned} \mathrm{diff}(x, u) &= x(u) - x(M_u) - (x^*(u) - x^*(M_u)) \\ &= x(v_i^*) - x(M^*) - (x^*(v_i^*) - x^*(M^*)) = \mathrm{diff}(x, v_i^*). \end{aligned}$$

showing (i).

Now we prove (ii). Consider $e \in E^+$ where $e = \{u, v\}$. Since $e \notin E(S_i^*)$ for all $i \in [k]$, we can choose a universal matching M exposing u and v by Lemma 5. Thus $M \cup \{e\}$ is also a matching. Notice M is x-tight, and so we have

$$\mathrm{excess}(x, e) = \underbrace{\mathrm{excess}(x, M \cup \{e\})}_{\geq \varepsilon_1} - \underbrace{\mathrm{excess}(x, M)}_{= \varepsilon_1} \geq 0$$

as desired. □

2.5 Optimal matchings of restricted cardinality

It is well-known [45], that for any $t \in \mathbb{N}$ and for any graph H, $\mathrm{conv}\{\chi(M) : M \in \mathcal{M}(H), |M| = t\}$ has the following linear description:

$$\begin{aligned} P_t(H) := \{x \in \mathbb{R}^E : & \\ & x(\delta(v)) \leq 1 && \text{for all} \quad v \in V(H) \\ & x(E(U)) \leq \frac{|U| - 1}{2} && \text{for all} \quad U \in \mathcal{O}(H) \\ & x(E(H)) = t \\ & x \geq 0\}. \end{aligned}$$

For any given $c \in R^{E(H)}$ we denote by $P_t^c(H)$ the set of vertices x of the above polytope maximizing $c^T x$, i.e. the optimal solutions to the linear program $\max\{c^T x : x \in P_t(H)\}$. The dual to this linear optimization problem is to minimize

$$\sum_{v\in V(H)} y_v + \sum_{U\in\mathcal{O}(H)} \frac{|U|-1}{2} z_U + t\gamma$$

where (y, z, γ) is in $D_{t,c}(H)$ defined as follows:

$$\begin{aligned} D_{t,c}(H) := \Big\{&(y, z, \gamma) \in \mathbb{R}^{V(H)} \times \mathbb{R}^{\mathcal{O}(H)} \times \mathbb{R} : \\ &y_u + y_v + \gamma + \sum_{U\in\mathcal{O}: uv\subseteq U} z_U \geq c(uv) \quad \text{for all} \quad uv \in E(H) \\ &y, z \geq 0\Big\}. \end{aligned}$$

Let $D_t^c(H)$ denote the set of optimal solutions to

$$\min\left\{\sum_{v\in V(H)} y_v + \sum_{U\in\mathcal{O}(H)} \frac{|U|-1}{2} z_U + t\gamma : (y, z, \gamma) \in D_{t,c}(H)\right\}.$$

Lemma 7 *Suppose* $2 \leq t \leq \left\lfloor \frac{|V(H)|}{2} \right\rfloor$. *Let* $x \in P_t^c(H)$ *and* $(y, z, \gamma) \in D_t^c(H)$. *If the support of* z *is laminar and* $y = 0$ *then there exists* $e \in \operatorname{supp}(x)$ *such that*

$$x - \chi(e) \in P_{t-1}^c(H)$$

and there exists $(0, z', \gamma') \in D_{t-1}^c(H)$ *with the support of* z' *laminar.*

Proof Let $\mathcal{L} = \operatorname{supp}(z)$ be the laminar family defined by the support of z. Let $S_1, \ldots, S_\ell \in \mathcal{L}$ be the top level sets of $\mathcal{L}$ (i.e. containment maximal sets), ordered so that

$$0 < z_{S_1} \leq z_{S_2} \leq \cdots \leq z_{S_\ell}.$$

Since $y = 0$, if there exists $e \in \operatorname{supp}(x) \cap (E(H) \setminus \bigcup_{i\in[\ell]} E(S_i))$ then by complementary slackness, $x - \chi(e) \in P_{t-1}^c(H)$ and $(0, z, \gamma) \in D_{t-1}^c(H)$. Thus we may assume that $x(uv) = 0$ for all $uv \in E(H) \setminus \bigcup_{i\in[\ell]} E(S_i)$. Let $uv \in E(S_1)$ such that $x_{uv} = 1$ and S_1 is a minimal set in $\mathcal{L}$ containing uv. Complementary slackness assures us that such uv exists. Let $x' = x - \chi(uv)$. Define $z' \in \mathbb{R}^{\mathcal{O}(H)}$ as follows

$$z'_U = \begin{cases} z_U - z_{S_1}, & \text{if } U = S_i \text{ for some } i \in [\ell] \\ z_U, & \text{otherwise.} \end{cases}$$

Define $\gamma' = \gamma + z_{S_1}$. We will show $x' \in P_{t-1}^c(H)$ and $(0, z', \gamma') \in D_{t-1}^c(H)$. First we verify feasibility. Indeed, since x is a matching with t edges, x' is a matching with $t-1$ edges and primal feasibility is satisfied. For dual feasibility, observe by our choice of S_1 that $z' \geq 0$, and for any edge $uv \in \bigcup_{i\in[\ell]} E(S_i)$ the net effect on the left hand side of the dual constraint associated with e is 0. For $uv \in E(H)\setminus\bigcup_{i\in[\ell]} E(S_i)$, the left hand side of the dual constraint associated with e increased by z_{S_1}.

Clearly since $\mathrm{supp}(z)$ is laminar, and $\mathrm{supp}(z') \subseteq \mathrm{supp}(z)\setminus\{S_1\}$, we have that $\mathrm{supp}(z')$ is laminar. It remains to verify optimality, which we will show via complementary slackness. Consider some $uv \in E(H)$ for which $x'_{uv} > 0$. Then $uv \in E(S_i)$ for exactly one i. Hence

$$\gamma' + \sum_{U\in\mathcal{O}(G):uv\subseteq U} z'_U = \gamma + z_{S_1} + (-z_{S_1}) + \sum_{U\in\mathcal{O}(G):uv\subseteq U} z_U = c(uv)$$

where the last equality follows from complementary slackness for x and $(0, z, \gamma)$. Lastly, let $U \in \mathcal{L}$ such that $z'_U > 0$. By our choice of z', $U \neq S_1$ and so

$$x'(E(U)) = x(E(U)) = \frac{|U|-1}{2}$$

where the last equality follows from complementary slackness for x and $(0, z, \gamma)$. □

Lemma 8 *Let $x \in P_1(\varepsilon_1)$ and let $M \in \mathcal{M}$ be a matching such that $M \subseteq \bigcup_{i\in[k]} E(S^*_i)$. Then there exists $M' \subseteq M^*$ such that* $\mathrm{excess}(x, M') \leq \mathrm{excess}(x, M)$ *and for all $i \in [k]$, $|M' \cap E(S^*_i)| = |M \cap E(S^*_i)|$.*

Proof Let $i \in [k]$ and let $H = G[S^*_i]$. Let $t = |M^* \cap E(H)|$, and let $c \in \mathbb{R}^{E(H)}$ be defined by

$$c(uv) := w(uv) - x(uv) \quad \text{for all } uv \in E(H).$$

Let $(y, z, \gamma) \in D^c_t(H)$. Since $c^T\chi(M) = -\mathrm{excess}(x, M)$ for all $M \in \mathcal{M}(H)$, by Lemma 4 and complementary slackness, we have $y = 0$. Now we use standard uncrossing techniques (see for ex. [36, Pages 141-150]) to obtain $(0, z, \gamma) \in D^c_t(H)$ with $\mathrm{supp}(z)$ laminar.

Note that $\chi(M^* \cap E(H))$ is in $P^c_t(H)$, otherwise we can replace $M^* \cap E(H)$, within H, with an optimal matching in $P^c_t(H)$ to obtain a matching with lower excess than M^*, contradicting the universal tightness of M^*.

Apply Lemma 7 inductively to obtain $M'_i \subseteq M^* \cap E(H)$ such that $\chi(M'_i) \in P^c_{|M'\cap E(H)|}(H)$. Since all S^*_i are disjoint, the matching $M' = \bigcup_{i\in[k]} M'_i$ is the desired matching. □

2.6 Compact formulation

Recall that x^* is a fixed universal allocation in $P_1(\varepsilon_1)$. Let $E^* \subseteq E$ denote the union of universal matchings, i.e. $E^* = \cup_{M\in\mathcal{M}_{uni}} M$. Since a matching is universal if any only if it is x^*-tight it is not hard to compute E^*. To test if an edge uv is in E^*, one simply computes a maximum weight matching on the graph obtained by deleting u and v from G, $G - u - v$, with respect to the weights $-\mathrm{excess}(x^*, e)$ for all $e \in E(G - u - v)$. Then $uv \in E^*$ if and only if the value of this maximum weight matching is $-\varepsilon_1 + \mathrm{excess}(x^*, uv)$.

We now define linear program ($\overline{P}_1$).

$$\max \quad \varepsilon \tag{$\overline{P}_1$}$$

$$\begin{aligned}
\text{s.t.} \quad & \operatorname{diff}(x,u) = \operatorname{diff}(x, v_i^*) && \text{for all} \quad u \in S_i^*,\ i \in [k] \\
& \operatorname{excess}(x,e) \le 0 && \text{for all} \quad e \in E^* \\
& \operatorname{excess}(x,e) \ge 0 && \text{for all} \quad e \in E^+ \\
& \operatorname{excess}(x, M^*) = \varepsilon && \\
& x(V) = \nu(G) && \\
& x \ge 0\,. &&
\end{aligned} \tag{2}$$

Let $\overline{\varepsilon}_1$ be the optimal value of the linear program ($\overline{P}_1$). We now show that $\overline{P_1}(\overline{\varepsilon}_1)$ is indeed a compact description of the leastcore $P_1(\varepsilon_1)$.

Theorem 4 *We have $\varepsilon_1 = \overline{\varepsilon}_1$ and $P_1(\varepsilon_1) = \overline{P}_1(\overline{\varepsilon}_1)$.*

Proof First, we show that $P_1(\varepsilon_1) \subseteq \overline{P}_1(\varepsilon_1)$. Consider $x \in P_1(\varepsilon_1)$. By Lemma 6(i) we have

$$\operatorname{diff}(x,u) = \operatorname{diff}(x, v_i^*) \qquad \text{for all} \quad u \in S_i^*,\ i \in [k].$$

Lemma 6(ii) shows that $\operatorname{excess}(x,e) \ge 0$ for all $e \in E^+$, and $\operatorname{excess}(x, M^*) = \varepsilon_1$ holds by the universality of M^*. It remains to show that

$$\operatorname{excess}(x,e) \le 0 \quad \text{for all} \quad e \in E^*.$$

Suppose for contradiction there exists $e \in E^*$ such that $\operatorname{excess}(x,e) > 0$. By the definition of E^*, there exists a universal matching M' containing e. Since M' is universal, $\operatorname{excess}(x, M') = \varepsilon_1$. But by our choice of e,

$$\operatorname{excess}(x, M' \backslash \{e\}) < \operatorname{excess}(x, M') = \varepsilon_1$$

contradicting that x is in $P_1(\varepsilon_1)$. Thus we showed that (x, ε_1) is feasible for ($\overline{P}_1$), i.e. we showed that $P_1(\varepsilon_1) \subseteq \overline{P}_1(\varepsilon_1)$.

To complete the proof we show that $\overline{P}_1(\overline{\varepsilon}_1) \subseteq P_1(\overline{\varepsilon}_1)$. Let x be an allocation in $\overline{P}_1(\overline{\varepsilon}_1)$. Due to the description of the linear program (P_1), it is enough to show that for every matching $M \in \mathcal{M}$ we have

$$\operatorname{excess}(x, M) \ge \overline{\varepsilon}_1\,.$$

Since $\operatorname{excess}(x,e) \ge 0$ for all $e \in E^+$, it suffices to consider only the matchings M, which are unions of matchings in the graphs $G[S_i^*]$, $i \in [k]$. Let $t_i := |M \cap E(S_i^*)|$. By Lemma 8 applied to x^* there exists $M' \subseteq M^*$ such that $\operatorname{excess}(x^*, M) \ge \operatorname{excess}(x^*, M')$ and $|M' \cap E(S_i^*)| = t_i$, for all $i \in [k]$. Then due to constraints (2) in ($\overline{P}_1$) we have

$$\text{excess}(x, M) = \underbrace{\sum_{i=1}^{k} 2t_i \,\text{diff}(x, v_i^*)}_{=\text{diff}(x,M')=\text{diff}(x,M)} + \underbrace{\text{excess}(x^*, M)}_{\geq \text{excess}(x^*,M')}$$
$$\geq \text{excess}(x, M') \geq \text{excess}(x, M^*) = \overline{\varepsilon}_1 \,,$$

where the last inequality follows since $M' \subseteq M^*$ and $\text{excess}(x, e) \leq 0$ for all $e \in E^*$.

Thus, we showed that $P_1(\varepsilon_1) \subseteq \overline{P}_1(\varepsilon_1)$ and $\overline{P}_1(\overline{\varepsilon}_1) \subseteq P_1(\overline{\varepsilon}_1)$. Recall, that ε_1 and $\overline{\varepsilon}_1$ are the optimal values of the linear programs (P_1) and ($\overline{P}_1$) respectively. Thus, we have $\varepsilon_1 = \overline{\varepsilon}_1$ and $P_1(\varepsilon_1) = \overline{P}_1(\overline{\varepsilon}_1)$. □

3 Computing the nucleolus

The last section presented a polynomial-size formulation for the leastcore LP (P_1). In this section we complete our polynomial-time implementation of Maschler's scheme by showing that (P_j) has the following compact reformulation:

$$\begin{array}{llll} \max & \varepsilon & & (\overline{P}_j) \\ \text{s.t.} & \text{excess}(x, e) \geq \varepsilon - \varepsilon_1 & \text{for all} & e \in E^+,\ e \notin \text{Fix}(\overline{P}_{j-1}(\overline{\varepsilon}_{j-1})) \\ & x(v) \geq \varepsilon - \varepsilon_1 & \text{for all} & v \in V,\ v \notin \text{Fix}(\overline{P}_{j-1}(\overline{\varepsilon}_{j-1})) \\ & \text{excess}(x, e) \leq \varepsilon_1 - \varepsilon & \text{for all} & e \in E^*,\ e \notin \text{Fix}(\overline{P}_{j-1}(\overline{\varepsilon}_{j-1})) \\ & x \in \overline{P}_{j-1}(\overline{\varepsilon}_{j-1})\,, & & \end{array}$$

where $\overline{\varepsilon}_j$ is the optimal value of the linear program ($\overline{P}_j$)

Theorem 5 *For all $j = 1, \ldots, j^*$, we have $\varepsilon_j = \overline{\varepsilon}_j$ and $P_j(\varepsilon_j) = \overline{P}_j(\overline{\varepsilon}_j)$.*

Proof We proceed by induction. For $j = 1$ the statement holds due to Theorem 4. Let us show that the statement holds for each $j = 2, \ldots, j^*$, assuming that the statement holds for $j - 1$. By induction, $P_{j-1}(\varepsilon_{j-1}) = \overline{P}_{j-1}(\overline{\varepsilon}_{j-1})$. We let $\mathcal{F} := \text{Fix}(P_{j-1}(\varepsilon_{j-1})) = \text{Fix}(\overline{P}_{j-1}(\overline{\varepsilon}_{j-1}))$ to ease presentation.

First we show $P_j(\varepsilon_j) \subseteq \overline{P}_j(\varepsilon_j)$. Let $x \in P_j(\varepsilon_j)$. First consider an edge $e \in E^+ \backslash \mathcal{F}$. By Lemma 5 there exists a universal matching $M \in \mathcal{M}_{uni}$ exposing the endpoints of e. Moreover, since $V(M) \in \text{Fix}(P_1(\varepsilon_1)) \subseteq \mathcal{F}$ we have $V(M \cup \{e\}) \notin \mathcal{F}$. Thus we see that

$$\text{excess}(x, e) = \underbrace{\text{excess}(x, M \cup \{e\})}_{\geq \varepsilon_j} - \underbrace{\text{excess}(x, M)}_{=\varepsilon_1} \geq \varepsilon_j - \varepsilon_1 .$$

Now consider $v \in V \backslash \mathcal{F}$. Similar to the previous argument, but via Lemma 4, we can find a universal matching M exposing v. Thus, since M is universal, $V(M) \cup \{v\} \notin \mathcal{F}$. We also have $\nu(V(M) \cup \{v\}) \geq w(M)$, so

$$x(v) = x(M \cup \{v\}) - w(M) - \text{excess}(x, M) \geq \varepsilon_j - \varepsilon_1 .$$

Finally consider $e = \{u, v\} \in E^* \backslash \mathcal{F}$. Since $e \in E^*$ there exists a universal matching $M \in \mathcal{M}_{uni}$ such that e is in M. Since $V(M) \in \mathrm{Fix}(P_1(\varepsilon_1)) \subseteq \mathcal{F}$ and $\{u, v\} \notin \mathcal{F}$, we see that $M \backslash \{e\} \notin \mathcal{F}$. Thus we have

$$\text{excess}(x, e) = \text{excess}(x, M) - \text{excess}(x, M \backslash \{e\}) \leq \varepsilon_1 - \varepsilon_j$$

as desired.

So we have shown $P_j(\varepsilon_j) \subseteq \overline{P}_j(\varepsilon_j)$, and thus it remains to show $\overline{P}_j(\overline{\varepsilon}_j) \subseteq P_j(\overline{\varepsilon}_j)$. Let $x \in \overline{P}_j(\overline{\varepsilon}_j)$ and let $S \subset V$ such that $S \notin \mathcal{F}$. We need to show

$$x(S) - \nu(S) \geq \overline{\varepsilon}_j.$$

Since $S \notin \mathcal{F}$, there exists $v \in S$ such that $\{v\} \notin \mathcal{F}$. Let M be a maximum w-weight matching in $G[S]$, i.e. $w(M) = \nu(S)$. Either $v \in S \backslash V(M)$ or $v \in V(M)$. We proceed by case distinction.

Case 1: $v \in S \backslash V(M)$. Since $x \geq 0$ we have $x(S) \geq x(V(M) \cup \{v\})$, and it suffices to prove that the right-hand side exceeds the weight of M by at least $\overline{\varepsilon}_j$. By assumption, $x \in \overline{P}_j(\overline{\varepsilon}_j) \subseteq \overline{P}_1(\overline{\varepsilon}_1) = P_1(\varepsilon_1)$, and hence

$$\text{excess}(x, M) \geq \varepsilon_1. \tag{3}$$

Together with $x(v) \geq \overline{\varepsilon}_j - \varepsilon_1$ this yields the desired inequality.

Case 2: $v \in V(M)$. The same argument as before applies if there is $u \in S \backslash V(M)$ with $\{u\} \notin \mathcal{F}$. Therefore, we focus on the case where $\{u\} \in \mathcal{F}$ for all $u \in S \backslash V(M)$. Then M contains an edge f such that $f \notin \mathcal{F}$. Again using $x \geq 0$, it suffices to show that $x(V(M)) \geq \overline{\varepsilon}_j$. We further distinguish cases: either $f \in M \cap E^+$ or $f \in M \backslash E^+$.

Case 2a: $f \in M \cap E^+$. Here, the desired inequality follows from $\text{excess}(x, M \backslash \{f\}) \geq \varepsilon_1$ due to $x \in P_1(\varepsilon_1)$ and from $\text{excess}(x, f) \geq \overline{\varepsilon}_j - \varepsilon_1$.

Case 2b: $f \in M \backslash E^+$. If $M \cap E^+$ has an edge that is not in $\mathcal{F}$ then we use the same argument as in Case 2a. So we may assume that all of the edges in $M \cap E^+$ are in $\mathcal{F}$. Now recall that $x \in P_1(\varepsilon_1)$, and hence $\text{excess}(x, e) \geq 0$ for all $e \in E^+$ by Lemma 6.(ii). Thus, we may assume that $V(M) \notin \mathcal{F}$ and $M \cap E^+ = \varnothing$.

By Lemma 8, applied to x, there exists $M' \subseteq M^*$ such that $|M' \cap E(S_i^*)| = |M \cap E(S_i^*)|$ for all $i \in [k]$ and $\text{excess}(x, M) \geq \text{excess}(x, M')$. Observe that

$$\begin{aligned} x(M') &= \sum_{i=1}^{k} x(M' \cap E(S_i^*)) \\ &\quad \text{(since } M' \subseteq M^*\text{)} \\ &= \sum_{i=1}^{k} x(M' \cap E(S_i^*)) + x^*(M' \cap E(S_i^*)) - x^*(M' \cap E(S_i^*)) \end{aligned}$$

$$
\begin{aligned}
&= x^*(M') + \sum_{i=1}^{k} |V(M') \cap S_i^*| \cdot \mathrm{diff}(x, v_i^*) \\
&\quad \text{(by Lemma 6)} \\
&= x^*(M') + \sum_{i=1}^{k} |V(M) \cap S_i^*| \cdot \mathrm{diff}(x, v_i^*) \\
&\quad (\text{since } |M' \cap E(S_i^*)| = |M \cap E(S_i^*)|) \\
&= x^*(M') + x(M) - x^*(M).
\end{aligned}
$$

That is, since $x^*(M') - x^*(M)$ is constant, $x(M')$ is a constant difference from $x(M)$. Hence $V(M) \in \mathcal{F}$ if and only if $V(M') \in \mathcal{F}$. Since $V(M) \notin \mathcal{F}$ we have $V(M') \notin \mathcal{F}$. Thus $V(M^* \backslash M') \notin \mathcal{F}$ and hence there exists $e \in M^* \backslash M'$ such that $e \notin \mathcal{F}$. Observe that e is in E^*. Hence,

$$
\begin{aligned}
\mathrm{excess}(x, M) &\geq \mathrm{excess}(x, M') \\
&= \mathrm{excess}(x, M^*) - \mathrm{excess}(x, e) - \mathrm{excess}(x, M^* \backslash (M' \cup \{e\})) \\
&\geq \varepsilon_1 - (\varepsilon_1 - \overline{\varepsilon}_j) \\
&= \overline{\varepsilon}_j,
\end{aligned}
$$

where the last inequality follows since $\mathrm{excess}(x, M^* \backslash (M' \cup \{e\})) \leq 0$ and $x \in \overline{P}_j(\overline{\varepsilon}_j)$, $e \in E^* \backslash \mathcal{F}$. □

Supposing we have a separation oracle for $P_j(\varepsilon_j)$, we can decide, for any $S \subseteq V$, if $S \in \mathrm{Fix}(P_j(\varepsilon_j))$ in polynomial time. To do so, using equivalence of separation and optimization we solve two linear programs: $\max\{x(S) : x \in P_j(\varepsilon_j)\}$ and $\min\{x(S) : x \in P_j(\varepsilon_j)\}$, and test if their optimal values are equal. Note that their optimal values are equal if and only if $S \in \mathrm{Fix}(P_j(\varepsilon_j))$.

With Theorem 5 we can replace each linear program (P_j) with ($\overline{P}_j$) in Maschler's Scheme. We already know how to execute the first round of Maschler's Scheme in polynomial time via Sect. 1.2. Therefore consider the j-th round of Maschler's Scheme, for $j > 1$, and suppose inductively that we can optimize over $\overline{P}_{j-1}(\overline{\varepsilon}_{j-1})$. Since the universal allocation x^*, the node sets S_i^*, $i \in [k]$, the edge set E^+, and the edge set E^* can all be computed in polynomial time, and we can test if a coalition is fixed over $\overline{P}_{j-1}(\overline{\varepsilon}_{j-1})$ in polynomial time, we have shown that the j-th round of Maschler's Scheme can be executed in polynomial time for cooperative matching games with empty core. Hence we can compute the nucleolus of such games in polyomial time. Therefore we have shown Theorem 1.

4 Leastcore separation for simple *b*-matching games

Up to this point in the paper, we have exclusively considered matching games. In this Section we would like to consider the natural generalization to b-matchings. For $b \in \mathbb{Z}^V_{\geq 0}$, a b-matching is a vector $x \in \mathbb{Z}^E_{\geq 0}$ such that

$$x(\delta(v)) \leq b(v) \quad \forall v \in V.$$

Given a graph $G = (V, E)$ with weights $w : E \to \mathbb{R}_{\geq 0}$, edge capacities $c \in \mathbb{Z}^E_{\geq 0}$, and node capacities $b \in \mathbb{Z}^V_{\geq 0}$ the corresponding c-capacitated b-matching game (V, ν) is the cooperative game where $\nu(S)$ is equal to the maximum weight of a b-matching in $G[S]$ using each edge e at most $c(e)$ times. If $c(e) = 1$ for all $e \in E$ then we refer to this special case as *simple* b-matching.

In this Section we describe a procedure for solving the separation problem for the leastcore of b-matching games where $b(v) \leq 2$ for all $v \in V$. This generalizes the matching games we considered throughout the paper, and to the authors' knowledge is the first polynomial time algorithm for this problem. The leastcore linear program for simple b-matching games can be stated as

$$\begin{aligned} \max\ & \varepsilon && && (P_1^b) \\ \text{s.t. } & x(M) \geq w(M) + \varepsilon && \text{for all} \quad M \in \mathcal{M}_b \\ & x(V) = \nu(G) \\ & x \geq \mathbb{0}, \end{aligned}$$

where $\mathcal{M}_b$ is the set of simple b-matchings in G, and $\nu(G)$ denotes the maximum weight of a simple b-matching in G.

It has been shown that core separation, and hence leastcore separation, is NP-hard [4] for simple b-matching games when we allow $b(v) \geq 3$ for arbitrary nodes $v \in V$. Thus we restrict our attention to b vectors such that $b(v) \leq 2$ for all $v \in V$, and consider simple b-matching games in this setting.

4.1 Max-weight min-cost matching

To separate the leastcore for a given point $(p, \varepsilon) \in \mathbb{R}^V \times \mathbb{R}$ it suffices to be able to solve

$$\begin{aligned} \max\ & w(M) - p(V(M)) && (P_2^b) \\ \text{s.t. } & M \in \mathcal{M}_b \end{aligned}$$

and verify that this value is at most $-\varepsilon$.

The challenge with adapting the leastcore separation algorithm for matching games to b-matching games is deciding what the weight of edges adjacent to b-value 2 nodes should be. If we simply take weights $\bar{w}(uv) = w(uv) - p(u) - p(v)$, for all $uv \in E$, as we would for matching games, then the cost $p(u)$ for nodes u with $b(u) = 2$ would be double counted in b-matchings which cover u with 2 edges. On the other hand if we only subtracted $\frac{1}{2}p(u)$ from weights of edges incident with b-value 2 node u, the cost $p(u)$ would only be half accounted for in b-matchings which only use one edge incident to u. To put it succinctly, we do not know a priori if the optimal matching will use one edge or two edges incident to any given b-value 2 node and hence do not known what the appropriate reduced weights should be.

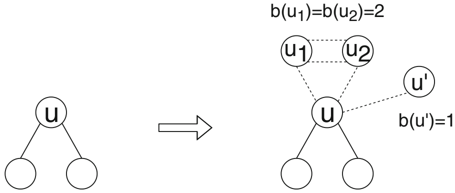


Fig. 2 The transformation of a b-value 2 vertex in G to the corresponding gadget in $\bar{G}$

To overcome the aforementioned difficulty, we will construct an auxiliary graph instance on which we can control the number of edges incident to b-value 2 nodes and still solve (P_2^b). Construct graph $\bar{G} = (\bar{V}, \bar{E})$ by starting with graph G. Now, for each $v \in V$ such that $b(v) = 2$ add nodes v_1, v_2 to $\bar{G}$ and create a cycle vv_1v_2. Let T be the set of all such cycles. Additionally for each $v \in V$ such that $b(v) = 2$, add node v' to $\bar{G}$ and create edge vv'. Now $\bar{G}$ consists of graph G, triangles T, and edges vv' for each $v \in V$ such that $b(v) = 2$ (Fig. 2).

For each $v \in V$ set $\bar{b}(v) = b(v)$. Set $\bar{b}(v_1) = \bar{b}(v_2) = 2$ for each $vv_1v_2 \in T$. Set $\bar{b}(v') = 1$ for each $v \in V$ such that $b(v) = 2$. For each $e \in \bar{E}$ set

$$\bar{c}(e) = \begin{cases} 2, & \text{if } e = v_1v_2 \text{ for some } vv_1v_2 \in T \\ 1, & \text{otherwise.} \end{cases}$$

Finally for each $uv \in \bar{E}$ set

$$\bar{w}(uv) = \begin{cases} w(uv) - \frac{p(u)}{b(u)} - \frac{p(v)}{b(v)}, & \text{if } uv \in E \\ -\frac{p(u)}{2} & \text{if } b(v) = 2 \text{ and } u = v' \\ 0, & \text{otherwise.} \end{cases}$$

The problem of finding a maximum $\bar{w}$-weight, $\bar{c}$-capacitated, $\bar{b}$-matching in $\bar{G}$ which fully matches each $\bar{b}$-value 2 node can be solved in polynomial time by computing a vertex solution to the following linear program [45]:

$$\begin{aligned} & \max\ \bar{w}^T x && \text{(Q)} \\ & \text{s.t. } x(\delta(v)) = \bar{b}(v) && \forall v \in \bar{V} : \bar{b}(v) = 2 \\ & x(\delta(v)) \le \bar{b}(v) && \forall v \in \bar{V} : \bar{b}(v) = 1 \\ & x(E(U)) + x(F) \le \frac{\bar{b}(U) + \bar{c}(F) - 1}{2} && \forall U \subseteq \bar{V}, F \subseteq \delta(U) : \bar{b}(U) + \bar{c}(F) \text{ is odd} \\ & 0 \le x \le \bar{c}. \end{aligned}$$

Note that the vertices of the feasible region of (Q) are all integral. We will show the optimal value of (Q) is equal to the value of an optimal solution to (P_2^b).

Lemma 9 *For each x feasible for (Q), for each $vv_1v_2 \in T$ either $x(v_1v_2) = 2$ or*

$$x(vv_1) = x(vv_2) = x(v_1v_2) = 1.$$

Proof First observe that $x(v_1v_2) \geq 1$, for otherwise

$$x(\delta(v_1)) \leq x(vv_1) \leq \bar{c}(vv_1) = 1 < 2 = \bar{b}(v_1)$$

and so x is not feasible for (Q). Now suppose for a contradiction that the claim is false. Then $x(v_1v_2) = 1$ and, without loss of generality, $x(vv_1) = 0$. But then

$$x(\delta(v_1)) = x(vv_1) + x(v_1v_2) = 1 < 2 = \bar{b}(v_1),$$

contradicting that x is a feasible for (Q). □

For an edge set $F \subseteq \bar{E}$, and a node $v \in \bar{V}$, define

$$\delta_F(v) := \{e \in F : v \in e\},$$

and define $d_F(v) := |\delta_F(v)|$.

Lemma 10 *Let $\bar{M}$ be the $\bar{c}$-capacitated $\bar{b}$-matching defined by the support of a feasible solution to (Q). Let M be the restriction of $\bar{M}$ to edges of G. Let $v \in \{v \in V : b(v) = 2\}$. The following hold:*

1. *If $d_M(v) = 0$ then vv_1, vv_2, and v_1v_2 are edges of $\bar{M}$.*
2. *If $d_M(v) = 1$ then vv' is an edge of $\bar{M}$.*
3. *If $d_M(v) = 2$ then v_1v_2 appears twice in the multiset $\bar{M}$.*

Proof The third claim is immediate from Lemma 9. The first claim follows from Lemma 9 and the necessity that v has two incidence edges in $\bar{M}$. The second claim follows from the same reasoning as the first claim. □

Theorem 6 *The optimal values of (P_2^b) and (Q) are equal.*

Proof Let $M \in \mathcal{M}_b$ be an optimal solution to (P_2^b). Then M is a node-disjoint union of a family of cycles, which we will denote by $\mathcal{C}$, and a family of paths which we will denote by $\mathcal{P}$. Let

$$B := \{v \in V : v \text{ is an endpoint of some path } P \in \mathcal{P} \text{ and } b(v) = 2\}.$$

Then $\bar{w}(M)$ overcounts $w(M) - p(V(M))$ by precisely $p(B)/2$. Formally,

$$\begin{aligned}\bar{w}(M) &= \sum_{P \in \mathcal{P}} \bar{w}(P) + \sum_{C \in \mathcal{C}} \bar{w}(C) \\ &= \frac{p(B)}{2} + \sum_{P \in \mathcal{P}} (w(P) - p(V(P))) + \sum_{C \in \mathcal{C}} (w(C) - p(C)) \\ &= w(M) - p(V(M)) + \frac{p(B)}{2}.\end{aligned}$$

Let $E_B = \{vv' : v \in B\}$. Then $\bar{w}(E_B) = -p(B)/2$. So we have that

$$\bar{w}(M \cup E_B) = \bar{w}(M) + \bar{w}(E_B) = w(M) - p(V(M)).$$

So if we let $\bar{M}_0 = M \cup E_B$ then $\bar{M}$ is a $\bar{c}$-capacitated, $\bar{b}$-matching in $\bar{G}$ with $\bar{w}$-weight equal to the optimal value of (P_2^b). It remains to add edges to $\bar{M}_0$ to create a feasible solution to (Q) decreasing its $\bar{w}$ value.

Observe that every node that is covered by $\bar{M}_0$ is covered by precisely its $\bar{b}$-value number of edges. Further observe that each pair of dummy nodes v_1, v_2 for each triangle $vv_1v_2 \in T$ is $\bar{M}_0$ exposed. Construct $\bar{M}$ from $\bar{M}_0$ by adding the following edges for each $v \in V$ such that $b(v) = 2$. Add edges $\{vv_1, vv_2, v_1v_2\}$ if v is $\bar{M}_0$ exposed; if v is $\bar{M}_0$ covered then add edge v_1v_2 twice to $\bar{M}$ (recalling that $\bar{M}$ is a multiset). It is not hard to see the resulting multiset $\bar{M}$ is feasible for (Q). The edges added to $\bar{M}_0$ to obtain $\bar{M}$ all have cost 0 and hence $\bar{M}$ is a solution to (Q) of weight exactly that of an optimal solution to (Q). Thus the value of (P_2^b) is at most that of (Q).

Now suppose that x is an optimal extreme point solution to (Q). Let $\bar{M}$ be the $\bar{c}$-capacitated, $\bar{b}$-matching in $\bar{G}$ obtained from the support of x. Let M be the restriction of $\bar{M}$ to the edges of G. Let $M' := \bar{M} \backslash E(T)$ be obtained from $\bar{M}$ by deleting all edges vv_1, vv_2, v_1v_2 such that $v \in V(\bar{M})$. Observe that $\bar{w}(\bar{M}) = \bar{w}(M')$, and the restriction of M' to edges of G is equal to M. Now we have,

$$\bar{w}(M) = w(M) - p(V(M)) + \sum_{v \in V(M): b(v)=2, d_M(v)=1} \frac{1}{2} p(v)$$

by our choice of weights $\bar{w}$. But by Lemma 10, each node $v \in V$ such that $b(v) = 2$ and $d_M(v) = 1$ is also matched to their corresponding node v' in M' in addition to their partner in M. Further, such vv' edges are the only edges in $M' \backslash M$. Hence by the choice of weight $\bar{w}$,

$$\bar{w}(M') = \bar{w}(M) - \sum_{v \in V(M): b(v)=2, d_M(v)=1} \frac{1}{2} p(v) = w(M) - p(V(M)).$$

Hence the value of (Q) is at most the value of (P_2^b) and thus the result holds. □

Since we can solve (Q) in polynomial time, this Theorem implies we can separate over the leastcore of simple b-matching games where $b(v) \leq 2$ for all $v \in V$, and hence we can compute a leastcore allocation of such games. Consequently, via Faigle, Kern, and Kuipers algorithm [19] we can compute points in the intersection of leastcore and prekernel in polynomial time for b-matching games with $b \leq 2$. If the leastcore intersect prekernel is unique for such an instance we can compute the nucleolus.

5 Conclusion

In Sect. 4 we gave a leastcore separation oracle for simple b-matching games where $b(v) \leq 2$ for all $v \in V$. The complexity of computing the nucleolus for such games remains open. We also mentioned in Sect. 4 that core separation for simple b-matching games with b unrestricted is NP-hard [4]. While this does not rule out computing the nucleolus in such a setting as NP-hard, it does rule out the technique for computing the nucleolus of matching games used in this paper of implementing Maschler's scheme efficiently. Settling the complexity of computing the nucleolus for such b-matching games would be an interesting problem.

Our algorithm relies heavily on the ellipsoid method. When the core is non-empty, there is a combinatorial algorithm for finding the nucleolus [3]. It would be interesting to develop a combinatorial algorithm for nucleolus computation of matching games in general.

Acknowledgements The authors thank Umang Bhaskar, Daniel Dadush, and Linda Farczadi for stimulating and insightful discussions related to this paper.

References

1. Aumann, R.J., Maschler, M.: Game theoretic analysis of a bankruptcy problem from the talmud. J. Econ. Theory **36**(2), 195–213 (1985)
2. Bateni, M., Hajiaghayi, M., Immorlica, N., Mahini, H.: The cooperative game theory foundations of network bargaining games. In: International Colloquium on Automata, Languages, and Programming, pp. 67–78. Springer (2010)
3. Biró, P., Kern, W., Paulusma, D.: Computing solutions for matching games. Int. J. Game Theory **41**, 75–90 (2012)
4. Biró, P., Kern, W., Paulusma, D., Wojuteczky, P.: The stable fixtures problem with payments. Games Econ. Behav. **11**(9), 24–241 (2017)
5. Brânzei, R., Solymosi, T., Tijs, S.: Strongly essential coalitions and the nucleolus of peer group games. Int. J. Game Theory **33**(3), 447–460 (2005)
6. Chen, N., Lu, P., Zhang, H.: Computing the nucleolus of matching, cover and clique games. In: AAAI (2012)
7. Cook, K.S., Yamagishi, T.: Power in exchange networks: a power-dependence formulation. Soc. Netw. **14**(3–4), 245–265 (1992)
8. Davis, M., Maschler, M.: The kernel of a cooperative game. Naval Res. Logist. Q. **12**(3), 223–259 (1965)
9. Deng, X., Fang, Q.: Algorithmic cooperative game theory. In: Pareto Optimality, Game Theory And Equilibria, pp. 159–185. Springer (2008)
10. Deng, X., Fang, Q., Sun, X.: Finding nucleolus of flow game. J. Comb. Optim. **18**(1), 64–86 (2009)
11. Deng, X., Ibaraki, T., Nagamochi, H.: Algorithmic aspects of the core of combinatorial optimization games. Math. Oper. Res. **24**(3), 751–766 (1999)
12. Easley, D., Kleinberg, J.: Networks, Crowds, and Markets: Reasoning About a Highly Connected World. Cambridge University Press, Cambridge (2010)
13. Edmonds, J.: Maximum matching and a polyhedron with 0, 1-vertices. J. Res. Natl. Bureau Stand. B **69**(125–130), 55–56 (1965)
14. Edmonds, J.: Paths, trees, and flowers. Can. J. Math. **17**(3), 449–467 (1965)
15. Elkind, E., Goldberg, L.A., Goldberg, P., Wooldridge, M.: Computational complexity of weighted threshold games. In: Proceedings of the National Conference on Artificial Intelligence, p. 718 (2007)
16. Eriksson, K., Karlander, J.: Stable outcomes of the roommate game with transferable utility. Int. J. Game Theory **29**(4), 555–569 (2001)

17. Faigle, U., Kern, W., Fekete, S.P., Hochstättler, W.: The nucleon of cooperative games and an algorithm for matching games. Math. Program. **83**(1–3), 195–211 (1998)
18. Faigle, U., Kern, W., Kuipers, J.: Note computing the nucleolus of min-cost spanning tree games is np-hard. Int. J. Game Theory **27**(3), 443–450 (1998)
19. Faigle, U., Kern, W., Kuipers, J.: On the computation of the nucleolus of a cooperative game. Int. J. Game Theory **30**(1), 79–98 (2001)
20. Faigle, U., Kern, W., Kuipers, J.: Computing an element in the lexicographic kernel of a game. Math. Methods Oper. Res. **63**(3), 427–433 (2006)
21. Faigle, U., Kern, W., Paulusma, D.: Note on the computational complexity of least core concepts for min-cost spanning tree games. Math. Methods Oper. Res. **52**(1), 23–38 (2000)
22. Farczadi, L.: Matchings and Games on Networks. University of Waterloo, Waterloo (2015)
23. Farczadi, L., Georgiou, K., Könemann, J.: Network bargaining with general capacities. In: European Symposium on Algorithms, pp. 433–444. Springer (2013)
24. Gale, D., Shapley, L.S.: College admissions and the stability of marriage. Am. Math. Monthly **69**(1), 9–15 (1962)
25. Gillies, D.B.: Solutions to general non-zero-sum games. Contrib. Theory Games **4**(40), 47–85 (1959)
26. Granot, D., Granot, F., Zhu, W.R.: Characterization sets for the nucleolus. Int. J. Game Theory **27**(3), 359–374 (1998)
27. Granot, D., Maschler, M., Owen, G., Zhu, W.R.: The kernel/nucleolus of a standard tree game. Int. J. Game Theory **25**(2), 219–244 (1996)
28. Grötschel, M., Lovász, L., Schrijver, A.: Geometric Algorithms and Combinatorial Optimization, Algorithms and Combinatorics: Study and Research Texts, vol. 2. Springer, Berlin (1988)
29. Grötschel, M., Lovász, L., Schrijver, A.: Geometric Algorithms and Combinatorial Optimization, vol. 2. Springer, Berlin (2012)
30. Kern, W., Paulusma, D.: Matching games: the least core and the nucleolus. Math. Oper. Res. **28**(2), 294–308 (2003)
31. Kleinberg, J., Tardos, E.: Balanced outcomes in social exchange networks. In: Proceedings of the fortieth annual ACM symposium on Theory of computing-STOC 08, p. 295. New York, New York, USA (2008)
32. Koopmans, T.C., Beckmann, M.: Assignment problems and the location of economic activities. In: Econometrica: journal of the Econometric Society, pp. 53–76 (1957)
33. Kopelowitz, A.: Computation of the kernels of simple games and the nucleolus of n-person games. Tech. rep., Hebrew University Jerusalem (Israel) Department of Mathematics (1967)
34. Kuhn, H.W.: The hungarian method for the assignment problem. Naval Res. Logist. Q. **2**(1–2), 83–97 (1955)
35. Kuipers, J., Solymosi, T., Aarts, H.: Computing the nucleolus of some combinatorially-structured games. Math. Program. **88**(3), 541–563 (2000)
36. Lau, L.C., Ravi, R., Singh, M.: Iterative Methods in Combinatorial Optimization, vol. 46. Cambridge University Press, Cambridge (2011)
37. Lemaire, J.: An application of game theory: cost allocation. ASTIN Bull. J. IAA **14**(1), 61–81 (1984)
38. Maschler, M., Peleg, B., Shapley, L.S.: Geometric properties of the kernel, nucleolus, and related solution concepts. Math. Oper. Res. **4**(4), 303–338 (1979)
39. Megiddo, N.: Computational complexity of the game theory approach to cost allocation for a tree. Math. Oper. Res. **3**(3), 189–196 (1978)
40. Nash Jr., J.F.: The bargaining problem. Econ. J. Econ. Soc. **10**, 155–162 (1950)
41. Paulusma, D.: Complexity Aspects of Cooperative Games. Twente University Press, New York (2001)
42. Potters, J., Reijnierse, H., Biswas, A.: The nucleolus of balanced simple flow networks. Games Econ. Behav. **54**(1), 205–225 (2006)
43. Rothvoß, T.: The matching polytope has exponential extension complexity. J. ACM **64**(6), 41 (2017)
44. Schmeidler, D.: The nucleolus of a characteristic function game. SIAM J. Appl. Math. **17**(6), 1163–1170 (1969)
45. Schrijver, A.: Combinatorial Optimization: Polyhedra and Efficiency, vol. 24. Springer, Berlin (2002)
46. Shapley, L.S., Shubik, M.: The assignment game I: the core. Int. J. Game Theory **1**(1), 111–130 (1971)
47. Solymosi, T., Raghavan, T.E.: An algorithm for finding the nucleolus of assignment games. Int. J. Game Theory **23**(2), 119–143 (1994)
48. Stearns, R.E.: Convergent transfer schemes for n-person games. Trans. Am. Math. Soc. **134**(3), 449–459 (1968)

49. Willer, D.: Network Exchange Theory. Greenwood Publishing Group, Greenwood (1999)
50. Ziegler, G.M.: Lectures on Polytopes, vol. 152. Springer, Berlin (2012)