The Stable Fixtures Problem with Payments

Péter Biró[1,2] [*], Walter Kern[3], Daniël Paulusma[4] [**], and Péter Wojuteczky[1]

[1] Institute of Economics, Hungarian Academy of Sciences; H-1112, Budaörsi út 45, Budapest, Hungary, peter.biro@krtk.mta.hu, peter.wojuteczky@gmail.com
[2] Department of Operations Research and Actuarial Sciences, Corvinus University of Budapest
[3] Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, P.O.Box 217, NL-7500 AE Enschede
w.kern@math.utwente.nl
[4] School of Engineering and Computing Sciences, Durham University, Science Laboratories, South Road, Durham DH1 3LE, UK,
daniel.paulusma@durham.ac.uk

Abstract. We generalize two well-known game-theoretic models by introducing multiple partners matching games, defined by a graph $G = (N, E)$, with an integer vertex capacity function b and an edge weighting w. The set N consists of a number of players that are to form a set $M \subseteq E$ of 2-player coalitions ij with value $w(ij)$, such that each player i is in at most $b(i)$ coalitions. A payoff is a mapping $p : N \times N \to \mathbb{R}$ with $p(i, j) + p(j, i) = w(ij)$ if $ij \in M$ and $p(i, j) = p(j, i) = 0$ if $ij \notin M$. The pair (M, p) is called a solution. A pair of players i, j with $ij \in E \setminus M$ blocks a solution (M, p) if i, j can form, possibly only after withdrawing from one of their existing 2-player coalitions, a new 2-player coalition in which they are mutually better off. A solution is stable if it has no blocking pairs. We give a polynomial-time algorithm that either finds that no stable solution exists, or obtains a stable solution. Previously this result was only known for multiple partners assignment games, which correspond to the case where G is bipartite (Sotomayor, 1992) and for the case where $b \equiv 1$ (Biro et al., 2012). We also characterize the set of stable solutions of a multiple partners matching game in two different ways and initiate a study on the core of the corresponding cooperative game, where coalitions of any size may be formed.

1 Introduction

Consider a group of soccer teams participating in a series of friendly games with each other off-season. Suppose each team has some specific target number of games it wants to play. For logistic reasons, not every two teams can play against each other. Each game brings in some revenue, which is to be shared by

the two teams involved. The revenue of a game may depend on several factors, such as the popularity of the two teams involved or the soccer stadium in which the game is played. In particular, at the time when the schedule for these games is prepared, the expected gain may well depend on future outcomes in the current season (which are in general difficult to predict [17]). In this paper, we assume for simplicity that the revenues are known. Is it possible to construct a *stable* fixture of games, that is, a schedule such that there exist no two unmatched teams that are better off by playing against each other? Note that this could require them to first cancel one of their others games in order not to exceed their targets.

The above example describes the problem introduced in this paper. We model it as follows. A *multiple partners matching game* is a triple (G, b, w), where $G = (N, E)$ is a finite undirected graph on n vertices and m edges with no loops and no multiple edges, $b : N \to \mathbb{Z}_+$ is a nonnegative integer function called a *vertex capacity function*, and $w : E \to \mathbb{R}_+$ is a nonnegative edge weighting. The set N is called the *player set*. There exists an edge $ij \in E$ if and only if players i, j can form a 2-player coalition. A set $M \subseteq E$ is a *b-matching* if every player i is incident to at most $b(i)$ edges of M. So, a b-matching is a set of 2-player coalitions, in which no player is involved in more 2-player coalitions than described by her capacity. If $ij \in M$ then i and j are *matched* by M. With M we associate a binary vector $x^M : E \to \{0, 1\}$ called the *characteristic function* of M, which is defined by $x^M(ij) = 1$ for all $ij \in M$ and $x^M(ij) = 0$ for all $ij \in E \setminus M$. Then we can write $\sum_{j:ij \in E} x^M(ij) \leq b(i)$ for all $i \in N$ as an alternative way to state the capacity condition. The *value* of a 2-player coalition i, j with $ij \in E$ is given by $w(ij)$.

A nonnegative function $p : N \times N \to \mathbb{R}_+$ is a *payoff* with respect to M if $p(i, j) + p(j, i) = w(ij)$ for $ij \in M$ and $p(i, j) = p(j, i) = 0$ for $ij \notin M$; we also say that M and p are *compatible*. Note that p prescribes how the value $w(ij)$ of a 2-player coalition $\{i, j\}$ is distributed amongst i and j, ensuring that non-coalitions between two players yield a zero payoff. A pair (M, p), where M is a b-matching and p is a payoff compatible with M, is a *solution* for (G, b, w). If M is a 1-matching (i.e, a *matching*) then, for each $i \in N$, we have $p(i, j) > 0$ for at most one player $j \neq i$, which must be matched to i. Hence, if $b \equiv 1$, we assume, with slight abuse of notation, that p is a nonnegative function defined on N.

Let (M, p) be a solution. Two players i, j with $ij \in E \setminus M$ may decide to form a new 2-player coalition if they are "better off", even if one or both of them must first leave an existing 2-player coalition in M (in order not to exceed their individual capacity). To describe this formally we define a *utility function* $u_p : N \to \mathbb{R}_+$, related to payoff p. If i is *saturated* by M, that is, if i is incident with $b(i)$ edges in M, then we let $u_p(i) = \min\{p(i, j) \mid x^M(ij) = 1\}$ be the worst payoff $p(i, j)$ of any 2-player coalition i is involved in. Otherwise, i is *unsaturated* by M and we define $u_p(i) = 0$. A pair i, j with $ij \in E \setminus M$ *blocks* (M, p) if $u_p(i) + u_p(j) < w(ij)$. We say that (M, p) is *stable* if it has no blocking pairs, or equivalently, if every edge $ij \in E \setminus M$ satisfies the *stability condition* $u_p(i) + u_p(j) \geq w(ij)$. We can now define our problem formally:

STABLE FIXTURE WITH PAYMENTS (SFP)
Instance: A multiple partners matching game (G, b, w)
Question: Does (G, b, w) have a stable solution?

So far, we modelled only situations in which 2-player coalitions can be formed. Allowing coalitions of any size is a natural and well-studied setting in the area of Cooperative Game Theory. Moreover, as we will discuss, there exist close relationships between stable solutions and their counterpart in the second setting, the so-called core allocations, which we define below.

A *cooperative game with transferable utilities* (TU-game) is a pair (N, v), where N is a set of n *players* and a *value function* $v : 2^N \to \mathbb{R}_+$ with $v(\emptyset) = 0$. It is usually assumed that the *grand coalition* N is formed. Then the central problem is how to allocate the *total value* $v(N)$ to the individual players in N. In this context, a *payoff* (or *allocation*) is a vector $p \in \mathbb{R}^N$ with $p(N) = v(N)$, where we write $p(S) = \sum_{i \in S} p(i)$ for $S \subseteq N$. The *core* of a TU-game consists of all allocations $p \in \mathbb{R}^N$ satisfying

$$
\begin{aligned}
p(S) &\geq v(S), \quad \emptyset \neq S \subseteq N \\
p(N) &= v(N)
\end{aligned}
\tag{1}
$$

A core allocation is seen as reasonable, because it offers no incentive for a subset of players to leave the grand coalition and form a coalition on their own. However, a TU-game may have an empty core. Hence, the most interesting computational complexity questions (given an input game) are:

1. Is the core nonempty?
2. Can we exhibit a vector in the core – provided there is any?
3. Does a given vector $p \in \mathbb{R}^N$ belong to the core?

In the literature both polynomial-time and NP-hardness results are known for each of these three questions (see e.g. [8]). An efficient algorithm for answering question 3 implies that questions 1-2 can be solved in polynomial time as well. This follows from the work of Grötschel, Lovász and Schrijver [13, 14] who proved, by refining the ellipsoid method of Khachiyan [18], that an efficient algorithm for solving the separation problem for a polyhedron P implies a polynomial-time algorithm that either finds that P is empty, or obtains a vector of P.

We define the TU-game (N, v) that corresponds with a multiple partners matching game (G, b, w) by setting, for every $S \subseteq N$,

$$
v(S) = w(M_S) = \sum_{e \in M_S} w(e),
$$

where M_S is a maximum weight b-matching in the subgraph of G induced by S (we define $v(S) = 0$ if S induces an edgeless graph). We say that (N, v) is *defined on* (G, b, w) but, unless confusion is possible, we may also call (N, v) a multiple partners matching game. If we say that the payoff vector p of a stable solution is a core allocation, we mean that the *total payoff vector* $p^t \in \mathbb{R}^N$ defined by $p^t(i) = \sum_{ij \in E} p(i, j)$ for all $i \in N$ is a core allocation.

Example. Let G be the 4-vertex cycle $v_1v_2v_3v_4v_1$. We define a vertex capacity function b by $b(v_1) = b(v_2) = 1$ and $b(v_3) = b(v_4) = 2$, and an edge weighting w by $w(v_1v_2) = 3$ and $w(v_2v_3) = w(v_3v_4) = w(v_4v_1) = 1$. The pair (M, p) with $M = \{v_1v_2, v_3v_4\}$ and $p(v_1, v_2) = p(v_2, v_1) = \frac{3}{2}$, $p(v_3, v_4) = p(v_4, v_3) = \frac{1}{2}$ and $p(v_2, v_3) = p(v_3, v_2) = p(v_4, v_1) = p(v_1, v_4) = 0$ is a solution for the multiple partners matching game (G, b, w). Note that $u_p(v_1) = u_p(v_2) = \frac{3}{2}$ and $u_p(v_3) = u_p(v_4) = 0$. We find that (M, p) is even a stable solution, because $u_p(v_2) + u_p(v_3) = \frac{3}{2} \geq 1 = w(v_2v_3)$ and $u_p(v_4) + u_p(v_1) = \frac{3}{2} \geq 1 = w(v_4v_1)$ (note that we only need to verify the stability condition for edges outside the matching). Moreover, the total payoff vector p^t given by $p^t(v_1) = p^t(v_2) = \frac{3}{2}$ and $p^t(v_3) = p^t(v_4) = \frac{1}{2}$ is readily seen to be a core allocation of the corresponding TU-game. In Section 3 we will give an example of a multiple partners matching game with no stable solutions for which the corresponding TU-game has a nonempty core.

Before stating our results for multiple partners matching games in both settings we first discuss some existing work. As we will see, our model in both settings generalizes (or relaxes) several well-known models.

Known Results. The first model that we discuss is related to the famous *stable marriage problem* (SM), defined by Gale and Shapley [11] as follows. Given sets I and J of men and women, respectively, let each player have a strict preference ordering over the opposite set of players. A set of marriages is a matching in the underlying bipartite graph with partition classes I and J. Such a matching is stable if there is no unmarried pair, who would prefer to marry each other instead of a possible other partner. Gale and Shapley [11] proved that every instance of this problem has a stable matching and gave a linear-time algorithm that finds one. The main assumptions in this model are

 (i) monogamy: each player is matched to at most one other player (1-matching);
 (ii) opposite-sex: every match is between players from I and J (bipartiteness);
(iii) no dowry: only cardinal preferences are considered (no payments).

Dropping one or more of these three conditions leads to seven new models, one of which corresponds to our model of multiple partner matching games, namely the one, in which *none* of the three conditions (i)–(iii) is imposed. Below we briefly survey the other six models (see also Table 1).

		opposite-sex	different-sex allowed
monogamy	no dowry	SM (stable marriage)	SR (stable roommates)
	dowry	SMP & assignment game	SRP & matching game
bigamy allowed	no dowry	many-to-many stable matching	SF (stable fixtures)
	dowry	MPA & mp assignment game	**SFP & mp matching game**

Table 1. The eight different models; "mp" stands for "multiple partners", and the highlighted case is the new model introduced and considered in this paper.

In the first three models that we discuss, payments are not allowed. Hence, the notion of a (core) allocation is meaningless for these three models.

Not (i). If we allow bigamy, that is, if we allow general b-matchings instead of only 1-matchings, we obtain the *many-to-many stable matching problem*, which generalizes the stable marriage problem. The problem variant, in which we demand that $b(i) = 1$ for each player $i \in I$, is called the *college admission problem* [11], which is also known as the *many-to-one stable matching problem* [25] and as the *hospital/residents problem* [23]. Gale and Shapley [11] proved that every instance of the college admission problem has a stable matching and gave a linear-time algorithm that finds one. Baïou and Balinski [1] proved these two results for the (more general) many-to-many stable matching problem.

Not (ii). If we allow same-sex marriages, so the underlying graph may be non-bipartite, then we get the *stable roommates problem* (SR), also defined by Gale and Shapley [11]. They proved that, unlike the previously discussed models, in this model a stable matching does not always exist. Irving [15] gave a linear-time algorithm for finding a stable matching (if there exists one).

Not (i) & (ii). Allowing bigamy *and* same-sex marriages leads to the *stable fixtures problem* (SF), which generalizes the stable roommates problem. Hence, a stable matching does not always exist. Irving and Scott [16] gave a linear-time algorithm for finding a stable matching (if there exists one). Cechlárová and Fleiner [5] defined the more general *multiple activities problem*, in which the underlying graph may have multiple edges. They proved that even in this setting a stable matching can be found in polynomial time (if there exists one). Moreover, they also showed that this problem can be reduced to SR by a polynomial size graph construction.

In the remaining three models we allow payments to individual players.

Not (iii). If we allow dowry then we obtain an *assignment game*, which is a multiple partners matching game (G, b, w) where G is bipartite and $b \equiv 1$. In this case the SFP problem is known as the *stable marriage problem with payments* (SMP). Koopmans and Beckmann [21] proved that every instance of SMP has a stable solution. Shapley and Shubik [27] proved that every core allocation is a payoff vector in a stable solution and vice versa. Consequently, every assignment game has a nonempty core. It is possible to obtain a stable solution in polynomial time and also to give affirmative answers to questions 1-3 about the core of an assignment game; in the next paragraph we explain that this holds even if we allow same-sex marriages.

Not (ii) & (iii). If we allow dowry and same-sex marriages then we obtain a *matching game*, which is a multiple partners matching game (G, b, w) where $b \equiv 1$. In this case the SFP problem is called the *stable roommates problem with payments* (SRP). The following two observations are well-known [4, 10] and easy to verify. First, a payoff p is a core allocation of a matching game if and only if there exists a matching M, such that (M, p) is a stable solution. Second, for matching games, the coalitions in the system of inequalities (1) may be restricted to 2-player coalitions. The latter means that question 3, on core membership,

can be answered in linear time. We also obtain polynomial-time algorithms for solving questions 1-2, about core nonemptiness, and finding a core allocation, and thus finding a stable solution; the restriction to 2-player coalitions even allows one to use the ellipsoid method of Khachiyan [18] directly. In a previous paper [4], we circumvented the ellipsoid method and presented an $O(nm + n^2 \log n)$-time algorithm that either finds that the core is empty, or obtains a core allocation.

Not (i) & (iii). If we allow dowry and bigamy then we obtain a *multiple partners assignment game*, which is a multiple partners matching game (G, b, w) where G is bipartite. In this case the SFP problem is called the *multiple partners assignment problem* (MPA). Just as matching games, multiple partners assignment games generalize assignment games. Sotomayor proved the following, which answers questions 1-2 positively (the answer to question 3 is still open).

Theorem 1 ([28]). *Every multiple partners assignment game has at least one stable solution, which can be found in polynomial time. Moreover, for every stable solution (M, p) it holds that M has maximum weight, p is a core allocation and every other maximum weight b-matching is compatible with p.*

Our Results. In Section 2 we will prove that SFP is polynomial-time solvable. This generalizes the aforementioned corresponding results for SRP and MPA, respectively. Our proof technique is based on a reduction to MPA. Moreover, we characterize the set of stable solutions for a given instance of SFP via a reduction to SRP. We do this via linear programming techniques that show a close relationship between optimal solutions in the dual LP for SFP and stable solutions in the reduced instance of SRP.

In Section 3 we first prove that also for multiple partners matching games, the payoff vectors in stable solutions are always core allocations. We then prove that core membership, which corresponds to question 3, is polynomial-time solvable for multiple partner matching games defined on a triple (G, b, w) with $b \leq 2$, that is, with $b(i) \leq 2$ for all $i \in N$. Due to the aforementioned result of Grötschel, Lovász and Schrijver [13, 14] we also obtain efficient answers to questions 1–2 (for $b \leq 2$). In our proof, we make a connection to the tramp steamer problem [19].

Finally, in Section 4, we give some directions for future work.

2   Stable Fixtures with Payments

In order to prove our results we will use two known results as lemmas.

Lemma 1 ([3]). *If (M, p) is a stable solution of an instance $(G, 1, w)$ of SRP then M has maximum weight, and every maximum weight matching M' of G is compatible with p.*

Lemma 2 ([22]). *Let G be a graph with vertex capacity function b and edge weighting w. Then it is possible to find a maximum-weight b-matching of G in $O(n^2 m \log(n^2/m))$ time.*

2.1 Characterizing Stable Solutions of SFP

In this subsection we show a correspondence of stable solutions of an instance (G, b, w) of SFP with stable solutions of a corresponding instance $(G', 1, w')$ of SRP, where G' is a graph of size $O(n^3)$, and with integral optimal solutions of an LP relaxation.

We first explain how to construct $(G', 1, w')$; see also Figure 1. Our construction is based on a well-known construction, which was introduced by Tutte [31] for nonbipartite graphs with no edge weights. For each player $i \in N$ with capacity $b(i)$ we create $b(i)$ copies, $i^1, i^2, \ldots i^{b(i)}$ in N'. For each edge $ij \in E$ we create four players, $\overline{i_j}, i_j, j_i, \overline{j_i}$, with edges $i^s \overline{i_j}$ for $s = 1 \ldots b(i)$, $\overline{i_j} i_j$, $i_j j_i$, $j_i \overline{j_i}$ and $\overline{j_i} j^t$ for $t = 1 \ldots b(j)$, each with weight $w(ij)$. This completes our construction. We write $G' = (N', E')$. Note that G' is bipartite if and only if G is bipartite. Hence, our construction also reduces an instance of MPA to an instance of SMP.
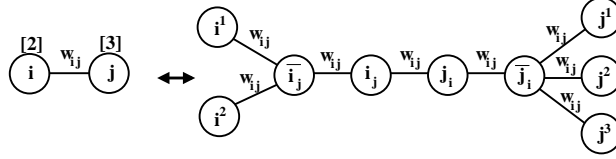
Fig. 1. The construction of an SPR instance $(G', 1, w')$ from a SFP instance (G, b, w).

Given an instance (G, b, w) of SFP, we formulate the corresponding primal LP, denoted by Primal-(G, b, w), as follows.

$$\max \sum_{ij \in E} w(ij) x(ij) \qquad \text{(P-obj)}$$

$$\text{subject to} \quad \sum_{j:ij \in E} x(ij) \leq b(i) \text{ for each } i \in N \qquad \text{(a)}$$

$$0 \leq x(ij) \leq 1 \text{ for each } ij \in E. \qquad \text{(b)}$$

The integral solutions of this LP are the b-matchings of instance (G, b, w). We now formulate the dual LP, denoted by Dual-(G, b, w).

$$\min \sum_{i \in N} b(i) y(i) + \sum_{ij \in E} d(ij) \qquad \text{(D-obj)}$$

$$\text{subject to} \quad y(i) + y(j) + d(ij) \geq w(ij) \text{ for each } ij \in E, \qquad \text{(a')}$$

$$\text{where } 0 \leq y(i) \text{ for all } i \in N, \text{ and } 0 \leq d(ij) \text{ for all } ij \in E.$$

Note that for an optimal dual solution (y, d), it holds that $d(ij) = [w(ij) - y(i) - y(j)]_+$ (where the latter notation means $max\{w(ij) - y(i) - y(j), 0\}$).

We are now ready to prove our main result.

Theorem 2. *Let (G, b, w) be an instance of* SFP. *The following statements are equivalent.*

1. *(G, b, w) has a stable solution.*
2. *$(G', 1, w')$ has a stable solution.*
3. *Primal-(G, b, w) has an integral optimal solution.*

Proof. We prove in three separate statements that 1 implies 2, 2 implies 3 and 3 implies 1.

1$\Rightarrow$2. Given a stable solution (M, p) of (G, b, w), we define (M', p') of $(G', 1, w')$ as follows. Recall that we define p' as a function on N, as $b \equiv 1$. The payments of the copies will be the same as the minimum payments of the original players, that is, for each $i \in N$, let $p'(i^s) = u_p(i)$ for every $s = 1 \ldots b(i)$. For each $ij \in M$, if j is i's s-th partner for some $s \in \{1 \ldots b(i)\}$ and i is j's t-th partner for some $t \in \{1 \ldots b(j)\}$ then let $i^s \overline{i_j} \in M'$, $i_j j_i \in M'$ and $\overline{j_i} j^t \in M'$ with the following payments: $p'(i_j) = p(i, j)$ and $p'(\overline{i_j}) = w(ij) - u_p(i)$, and similarly $p'(j_i) = p(j, i)$ and $p'(\overline{j_i}) = w(ij) - u_p(j)$. For each $ij \in E \setminus M$, let $\overline{i_j} i_j$, $j_i \overline{j_i} \in M'$ with $p'(i_j) = min\{u_p(i), w(ij)\}$ and $p'(\overline{i_j}) = w(ij) - min\{u_p(i), w(ij)\}$, and similarly $p'(\overline{j_i}) = w(ij) - min\{u_p(j), w(ij)\}$ and $p'(j_i) = min\{u_p(j), w(ij)\}$. We observe that (M', p') is a solution for $(G', 1, w')$. In order to prove that (M', p') is a stable solution we have to check the stability condition for each edge not in M'. As $u_{p'}(i) = p'(i)$ for any $i \in N'$, this comes down to checking whether $p'(i) + p'(j) \geq w(ij)$ for all $i, j \in N'$. For each edge $i^s \overline{i_j}$ not in M' there are two cases. In the first case, when $p'(\overline{i_j}) = w(ij) - u_p(i)$, the condition is satisfied by equality, because $p'(i^s) = u_p(i)$. In the other case, we have $p'(\overline{i_j}) = w(ij) - min\{u_p(i), w(ij)\}$, so $p'(i^s) + p'(\overline{i_j}) = u_p(i) + w(ij) - min\{u_p(i), w(ij)\} \geq w(ij)$. For each edge $\overline{i_j} i_j$ not in M', $p'(\overline{i_j}) = w(ij) - u_p(i)$ and $p'(i_j) = p(i, j)$, so $p'(\overline{i_j}) + p'(i_j) \geq w(ij)$ as $u_p(i) \leq p(i, j)$. Finally, if $i_j j_i$ is not in M' then in the first case, when any payoff of the middle players is $w(ij)$, the requirement is trivially true, and in the second case when the payoffs of both players differ from $w(ij)$ the stability of (M, p) implies that $p'(i_j) + p'(j_i) = u_p(i) + u_p(j) \geq w(ij)$.

2$\Rightarrow$3 Suppose that (M', p') be a stable solution for $(G', 1, w')$. We will first prove that (M', p') can be transformed into a stable solution (M'', p') of $(G', 1, w')$, where M'' is a matching that we may obtain by the above reduction from (G, b, w) to $(G', 1, w')$. From Lemma 1 we know that M' is a maximum weight matching in $(G', 1, w')$. For a player set X let $E(X)$ denote the set of edges incident to any player in X. Let $E_{ij} = E(\{\overline{i_j}, i_j, j_i, \overline{j_i}\})$. Considering the edge set E_{ij}, either three or two edges should be contained in a maximum weight matching M'. If $|E_{ij} \cap M'| = 3$ then M' should contain $i_j j_i$, $i^s \overline{i_j}$ for some $s \in \{1, \ldots, b(i)\}$ and $\overline{j_i} j^t$ for some $t \in \{1, \ldots, b(j)\}$. Let M'' contain the same edges in this case. If $|E_{ij} \cap M'| = 2$ then let $\{\overline{i_j} i_j, j_i \overline{j_i}\} \subseteq M''$. Therefore M'' is a maximum weight matching, and from Lemma 1, (M'', p') is a stable solution in $(G', 1, w')$, which can be obtained from a reduction from (G, b, w) to $(G', 1, w')$, as described in the first part of our proof. Now we note that $p'(i^s) = p'(i^t)$ for any indices $s, t \in \{1 \ldots b(i)\}$. This is because $p'(i^s) < p'(i^t)$ would imply that $p'(i^s) > 0$, so i^s must be covered by M', say, $i^s \overline{i_j} \in M'$, where

$p'(i^s) + p'(\bar{i_j}) = w(ij)$, and hence $i^r\bar{i_j}$ would be blocking, a contradiction. Therefore we can set $y(i) = p'(i^s)$ for every copy i^s of $i \in N$, as this is well-defined. Together with $d(ij) = [w(ij) - y(i) - y(j)]_+$ for any $ij \in E$ we get a feasible solution (y,d) of Dual-(G,b,w). Now we define an integral feasible solution x of Primal-(G,b,w) as follows. Let $x(ij) = 1$ if $i_j j_i \in M''$ and $x(ij) = 0$ otherwise. We prove that (y,d) and x are both optimal solutions as they satisfy the complementary slackness conditions.

- If $\sum_{j:ij \in E} x(ij) < b(i)$ then some copy of $i \in N$, say i^s, is unmatched in M'', therefore $y(i) = p'(i^s) = 0$.
- If $x(ij) < 1$ for some $ij \in E$ then $i_j j_i \notin M''$ and therefore $y(i) + y(j) \geq p'(i_j) + p'(j_i) \geq w(ij)$, where the first inequality is implied by the stability condition for $i^s\bar{i_j} \notin M''$ and $j^t\bar{j_i} \notin M''$, and the second inequality is implied by the stability condition for $i_j j_i \notin M''$. As a consequence $d(ij) = [w(ij) - y(i) - y(j)]_+ = 0$ should hold.
- If $x(ij) > 0$ then $i_j j_i \in M''$ and therefore $y(i) + y(j) \leq p'(i_j) + p'(j_i) = w(ij)$, where the inequality is implied by the stability condition for $i_j\bar{i_j} \notin M''$ and $j_i\bar{j_i} \notin M''$. As a consequence $d(ij) = w(ij) - y(i) - y(j)$ and the dual condition (a') is binding.

Hence x is an optimal integral solution for Primal-(G,b,w), as required.

3⇒1 From an optimal solution x of Primal-(G,b,w) and optimal dual solution (y,d) of Dual-(G,b,w) we create a stable solution (M,p) for (G,b,w) as follows. Let M be the b-matching defined by the characteristic function x. For each $ij \in M$ we choose $\xi(i,j) \geq 0$ and $\xi(j,i) \geq 0$ with $\xi(i,j) + \xi(j,i) = d(ij)$ and define $p(i,j) = y(i) + \xi(i,j)$, and otherwise we define $p(i,j) = 0$. These are valid payments, as $x(ij) = 1$ implies $p(i,j) + p(j,i) = y(i) + \xi(i,j) + y(j) + \xi(j,i) = y(i) + y(j) + d(ij) = w(ij)$, where the last equality is coming from the fact that condition (a') is binding. Now we show that $u_p(i) \geq y(i)$ for every $i \in N$. If i is unsaturated by M then $u_p(i) = y(i) = 0$ again by the complementary slackness condition for $y(i)$. If i is saturated then for every $ij \in M$ we have $p(i,j) \geq y(i)$ by our setting of $p(i,j)$, and therefore $u_p(i) \geq y(i)$ by the definition of $u_p(i)$. Finally, if $ij \notin M$ then $x(ij) < 1$ implies $d(ij) = 0$. Consequently, $u_p(i) + u_p(j) \geq y(i) + y(j) \geq w(ij)$, where the last equality is due to the fact that (a') is tight. This completes our proof. □

Remark 1. From the proof of Theorem 2 we note that there is a one-to-one correspondence between the stable payments in an SFP instance (G,b,w) and the utilities of some of the players in its reduced SPR instance $(G',1,w')$, namely a pair (M,p) is a stable solution for (G,b,w) if and only if there exists a stable solution (M',p') for $(G',1,w')$ with $p'(i_j) = p(i,j)$ for every $ij \in M$.

Remark 2. Assume (G,b,w) is a solvable instance of SFP. Then there is a one-to-one correspondence between the dual variables y of an optimal solution (y,d) of Dual-(G,b,w) and the stable payoffs of the players' copies in the reduction G'.

2.2 Solving SFP efficiently

In order to solve SFP on an instance (G, b, w), we construct the instance $(G', 1, w')$ of SRP. This takes $O(n^3)$ time as we may assume without loss of generality that $b(i) \leq n$ for all $i \in N$ and thus $|V(G')| = \sum_{i \in N} b(i) + 4m = O(n^2)$ and $|E(G')| \leq \sum_{i \in N} b(i)n + 3m \leq n^3 + 3m = O(n^3)$. We then use the aforementioned algorithm of Biró et al. [4] to compute in $O(n'm' + n'^2 \log n') = O(n^5)$ time a stable solution for $(G', 1, w')$ or else conclude that $(G', 1, w')$ has no stable solution. In the first case we can modify the stable solution into a stable solution of (G, b, w) in $O(n^3)$ time, as described in the proof of Theorem 2. In the second case, Theorem 2 tells us that (G, b, w) has no stable solution. The total running time is $O(n^5)$. Below we present an algorithm that solves SFP in $O(n^2 m \log(n^2/m))$ time.

A *half-b-matching* in a graph $G = (N, E)$ with an integer vertex capacity function b and an edge weighting w is an edge mapping f that maps each edge e to a value in $\{0, \frac{1}{2}, 1\}$, such that $\sum_{e:v \in e} f(e) \leq b(v)$ for each $v \in N$. The weight of f is $w(f) = \sum_{e \in E} w(e)f(e)$.

Let (G, b, w) be an instance of SFP. We define its *duplicated* instance $(\hat{G}, \hat{b}, \hat{w})$ of MPA as follows. We replace each player i of G by two players i' and i'' in $\hat{G}$ with the same capacities, that is, we set $\hat{b}(i') = \hat{b}(i'') = b(i)$. Moreover, we replace each edge ij by two edges $i'j''$ and $i''j'$ with half-weights, that is, we set $\hat{w}(i'j'') = \hat{w}(i''j') = \frac{1}{2}w(ij)$.

In a previous work [4], three of us proved the following statement for instances of SRP only. In our next theorem we generalize this result for instances of SFP by using similar arguments (proof omitted due to space restrictions).

Theorem 3. *An instance (G, b, w) of SFP admits a stable solution if and only if the maximum weight of a b-matching in G is equal to the maximum weight of a half-b-matching in G.*

We note that the maximum weight of a b-matching can be computed in $O(n^2 m \log(n^2/m))$ time, as described in Lemma 2. The maximum weight of a half-b-matching can be computed in the same running time, since the maximum weight of a half-b-matching in (G, b, w) is the same as the maximum weight of a b-matching in a duplicated bipartite graph $(\hat{G}, \hat{b}, \hat{w})$, as explained in the proof of Theorem 3. This leads to the following result.

Theorem 4. SFP *can be solved in $O(n^2 m \log(n^2/m))$ time.*

3 Core Properties

We first present the following result, which is in line with corresponding results for the other models and thus shows that the notion of stability is well defined with respect to the core definition. We omit the proof due to page restrictions.

Proposition 1. *The payoff vector of every stable solution of a multiple partners matching game is a core allocation.*

In contrast to our results in Section 2, the analysis of the core cannot be reduced to the the case in which we have unit vertex capacities. We illustrate this by giving the following example that shows that the core of a multiple partners matching game can be nonempty whilst there exist no stable solutions.

Example 1. Take a *diamond*, that is, a cycle on three vertices s_1, s_2, s_3 to which we add a fourth vertex u with edges $s_2 u$ and $s_3 u$. We set $b(s_i) = 2$ for $i = 1, 2, 3$ and $b(u) = 1$, and $w \equiv 1$. Then Theorem 3 tells us that a stable solution does not exist, since the maximum weight of a b-matching is 3, whilst the maximum weight of a half-b-matching is $3\frac{1}{2}$ (for the latter, take $f(s_1 s_2) = f(s_1 s_3) = 1$ and $f(s_2 s_3) = f(s_2 u) = f(s_3 u) = \frac{1}{2}$). However, the total payoff vector p^t defined by $p^t(s_i) = 1$ for $i = 1, \ldots, 3$ and $p^t(u) = 0$, which corresponds to, say, the b-matching $M = \{s_1 s_2, s_1 s_3, s_2 s_3\}$ with payments $p(s_1, s_2) = 1$, $p(s_2, s_3) = 1$ and $p(s_3, s_1) = 1$ and zero payments for the other edges is in the core.

Moreover, the core of a multiple partners matching game may be empty. This is clear for $b \equiv 1$ (for instance, take a triangle on three vertices and let $w \equiv 1$) but may also be the case for $b \neq 1$. We illustrate the latter by presenting the following example, which shows that the core of a b-matching game may be empty even if $b \equiv 2$.

Example 2. Take a *net*, that is, a cycle on three vertices s_1, s_2 and s_3 to which we add a pendant (degree 1) vertex t_i to s_i for $i = 1, 2, 3$. We set $b \equiv 2$ and $w \equiv 1$. Then $v(N) = 4$. We may assume without loss of generality that $p(t_i) = 0$ for $i = 1, 2, 3$. Then, by symmetry, $p(s_i) = \frac{4}{3}$ for $i = 1, 2, 3$. Take the coalition $S = \{s_1, s_2, t_1, t_2\}$. It holds that $p(S) = 2 \times \frac{4}{3} < 3 = v(S)$ and thus the core of this game is empty.

In what follows, we analyze the case $b(i) \leq 2$ for $i = 1, ..., n$. Note that a player i with $b(i) = 0$ necessarily gets 0 payoff in any core allocation, so the problem reduces trivially to $G \backslash \{i\}$. For this reason we assume $b(i) \geq 1$ for all $i \in N$ in the following.

Our main result is that answering question 3, on testing core membership, can be done in polynomial time, hereby answering question 3 positively. Recall that this implies the existence of a polynomial-time algorithm that either finds that the core is empty, or else obtains a core allocation. As mentioned in Section 1, our algorithm uses an algorithm that solves the tramp steamer problem, which we formally define below, as a subproblem.

Let $G = (N, E)$ be a graph with an edge weighting $p : E \rightarrow \mathbb{R}_+$ called the *profit function* and an edge weighting $w : E \rightarrow \mathbb{R}_+$ called the *cost function*. Let $C = (N_C, E_C)$ be a (simple) cycle of G. The *profit-to-cost ratio* for a cycle C is $\frac{p(C)}{w(C)}$ where we write $p(C) = p(E_C)$ and $w(C) = w(E_C)$. The tramp steamer problem is that of finding a cycle C with maximum profit-to-cost ratio. This problem is well-known to be polynomial-time solvable both for directed and undirected graphs (see [19], or [24] for a treatment of more general "fractional optimization" problems).

Lemma 3. *The tramp steamer problem can be solved in polynomial time.*

We are now ready to prove the main result of this section.

Theorem 5. *It is possible to test in polynomial time if an allocation is in the core of a multiple partners matching game defined on a triple (G, b, w) with $b \leq 2$.*

Proof. Let (N, v) be a multiple partners matching game defined on a triple (G, b, w), where $b(i) \leq 2$ for all $i \in N$. Given $S \subseteq N$, a maximum weight b-matching in $G[S]$ is composed of cycles and paths. Hence the core can be alternatively described by the following (slightly smaller) set of constraints:

$$
\begin{aligned}
p(C) &\geq w(C), \text{ for all cycles } C \in \mathcal{C} \\
p(P) &\geq w(P), \text{ for all paths } \ P \in \mathcal{P} \\
p(N) &= v(N).
\end{aligned}
\tag{2}
$$

Here, $\mathcal{C}$ stands for the set of simple cycles $C \subseteq E$ in G with $b(i) = 2$ for all $i \in V(C)$. Similarly, $\mathcal{P}$ stands for the set of simple paths P with $b(i) = 2$ for all inner points on P.

Given $p \in \mathbb{R}^N$, we can of course easily check whether $p(N) = v(N)$ holds by computing a maximum weight b-matching in G, which can be done in polynomial time by Lemma 2. Thus we are left with the inequalities for cycles and paths in (2).

We deal with the cycles first. Let $N_2 := \{i \in N \mid b(i) = 2\}$ and $G_2 := G[N_2]$. In the induced graph $G_2 = (N_2, E_2)$, we "discharge" the given allocations $p(i)$ to the edges by setting $p(i, j) := (p(i) + p(j))/2$ for every edge ij in G_2. This defines an edge weighting $p : E_2 \to \mathbb{R}$ such that, obviously, the core constraints for cycles are equivalent to

$$
\max_{C \in \mathcal{C}} \frac{w(C)}{p(C)} \ \leq \ 1,
$$

where the maximum is taken over all cycles in G_2. Hence we obtained an instance of the tramp steamer problem, which is polynomial-time solvable by Lemma 3. Note that by solving the above minimization problem we either find that all cycle constraints in (2) are satisfied or we end up with a particular cycle C corresponding to a violated core inequality. (The latter is of particular importance if we intend to use the "membership oracle" as a subroutine for the ellipsoid method.)

In what follows, we thus assume that all cycle constraints in (2) are satisfied by the given vector $p \in \mathbb{R}^N$ and turn to the path constraints. We process these separately for all possible endpoints $i_0, j_0 \in N$ (with $i_0 \neq j_0$) and all possible lengths $k = 1, ..., n - 1$. Let $\mathcal{P}_k(i_0, j_0) \subseteq \mathcal{P}$ denote the set of (simple) $i_0 - j_0$-paths of length k in G. We construct a corresponding auxiliary graph $G_k(i_0, j_0)$, a subgraph of $G \times P_{k+1}$, the product of G with a path of length k. To this end, let $N_2^{(1)}, ..., N_2^{(k-1)}$ be $k - 1$ copies of N_2. The vertex set of $G_k(i_0, j_0)$ is then $\{i_0, j_0\} \cup N_1^{(1)} \cup ... \cup N_2^{(k-1)}$. We denote the copy of $i \in N_2$ in $N_2^{(r)}$ by $i^{(r)}$. The

edges of $G_k(i_0, j_0)$ and their weights $\bar{w}$ can then be defined as

$$
\begin{array}{lll}
(i_0, j^{(1)}) & \text{for } i_0 j \in E & \text{with weight } \bar{w}(i_0 j) := p(i_0) + p(j)/2 - w(i_0 j) \\
(i^{(r-1)}, j^{(r)}) & \text{for } ij \in E & \text{with weight } \bar{w}(ij) := (p(i) + p(j))/2 - w(ij) \\
(i^{(k-1)}, j_0) & \text{for } ij_0 \in E & \text{with weight } \bar{w}(ij_0) := p(i)/2 + p(j_0) - w(ij_0).
\end{array}
$$

We claim that $p(P) \geq w(P)$ holds for all $P \in \mathcal{P}$ if and only if the (w.r.t. $\bar{w}$) shortest i_0-j_0–path in $G_k(i_0, j_0)$ has weight ≥ 0 for all $i_0 \neq j_0$ and $k = 1, .., n-1$. Then what is left to do, in order to verify whether $p(P) \geq w(P)$ holds for all $P \in \mathcal{P}$, is to solve $O(n^3)$ instances of the shortest path problem, each of which have size $O(n^2)$ by using the well-known Bellman-Ford algorithm [2].

First suppose some $P \in \mathcal{P}$ has $p(P) < w(P)$. Let i_0 and j_0 denote its endpoints and let k denote its length. Then $P \in \mathcal{P}_k(i_0, j_0)$ corresponds to an $i_0 - j_0$-path $\bar{P}$ in $G_k(i_0, j_0)$ of weight $\bar{w}(\bar{P}) < 0$. Now we will show that $p(P) \geq w(P)$ for all $P \in \mathcal{P}$ implies $\bar{w}(\bar{P}) \geq 0$ for all $i_0 - j_0$-paths $\bar{P}$ in any $G_k(i_0, j_0)$. Indeed, an $i_0 - j_0$-path $\bar{P}$ visiting players $i_0, i_1^{(1)}, ..., i_{k-1}^{(k-1)}, j_0$ corresponds to a simple $i_0 - j_0$ path $P \subseteq E$ in G plus possibly a number of cycles $C_1, ..., C_s \subseteq E$. Furthermore, by definition of $\bar{w}$, we have $\bar{w}(\bar{P}) = p(P) - w(P) + \sum_i p(C_i) - w(C_i) \geq p(P) - w(P)$, as we assume that $p(C) \geq w(C)$ holds for all cycles. Hence, indeed, $\bar{w}(\bar{P}) \geq 0$, as claimed. $\qquad\square$

4 Future Work

We finish our paper with some directions for future research. Our reduction from SFP to SRP might be used to generalize more known results from SRP to SFP. For instance, can we generalize the path to stability result of Biró et al. [3] for SRP to be valid for SFP as well?

We do not know the answers for questions 1–3 on the core of multiple partners matching games (G, b, w) when $b(i) \geq 3$ for some player i. In particular, we tend to believe that testing core membership is NP-hard (for example, the case $b \equiv 3$ looks close to the maximum 3-regular subgraph problem, which is NP-complete, see e.g. [12]). However, we did not succeed in finding a proof. We recall that testing core membership is also open for multiple partners assignment games (except when $b \leq 2$).

Chalkiadakis et al. [6] defined cooperative games with overlapping coalitions, where players can be involved in coalitions with different intensities, leading to three alternative core definitions. It would be interesting to study the problem of finding a stable solutions and questions 1–3 in these settings. To illustrate this, suppose that the set of soccer teams from the example in Section 1 consists of international teams. Then it seems realistic that the home team needs to spend fewer days for playing the game than the visiting team, which must travel from another country. Hence, every team now has a number of days for playing friendly games instead of an upper limit (target) on the number of such games.

References

1. M. Baïou and M. Balinski, Many-to-many matching: stable polyandrous polygamy (or polygamous polyandry, Discrete Applied Mathematics 101 (2000) 1–12.
2. R. Bellman, On a routing problem, Quarterly of Applied Mathematics 16 (1958) 87–90.
3. P. Biró, M. Bomhoff, P.A. Golovach, W. Kern and D. Paulusma, Solutions for the Stable Roommates Problem with Payments, Theoretical Computer Science 540–541 (2014) 53–61.
4. P. Biró, W. Kern and D. Paulusma, Computing solutions for matching games, International Journal of Game Theory 41 (2012) 75–90.
5. K. Cechlárová and T. Fleiner, On a generalization of the stable roommates problem, ACM Transaction on Algorithms 1 (2005) 143–156.
6. G. Chalkiadakis, E. Elkind, E. Markakis, M. Polurkov, and N.R. Jennings, Cooperative Games with Overlapping Coalitions, Journal of Artificial Intelligence Research 39 (2010) 179–216.
7. E. Diamantoudi, E. Miyagawa and L. Xue, Random paths to stability in the roommates problem, Games and Economic Behavior 48 (2004) 18–28.
8. X. Deng, T. Ibaraki, and H. Nagamochi, Algorithmic aspects of the core of combinatorial optimization games. Mathematics of Operations Research 24 (1999) 751–766.
9. J. Egerváry, Matrixok kombinatorius tulajdonságairól, Matematikai és Fizikai Lapok 38 (1931) 16–28.
10. K. Eriksson and J. Karlander, Stable outcomes of the roommate game with transferable utility, International Journal of Game Theory 29 (2001) 555–569.
11. D. Gale and L.S. Shapley, College Admissions and the Stability of Marriage, American Mathematical Monthly 69 (1962) 9–15.
12. M. Garey and D. Johnson, Computers and Intractability, A Guide to the Theory of NP-Completeness, Freeman, San Francisco, 1979.
13. M. Grötschel, L. Lovász and A. Schrijver, The ellipsoid method and its consequences in combinatorial optimization, Combinatorica 1 (1981) 169–197 [corrigendum: Combinatorica 4 (1984), 291–295].
14. M. Grötschel, L. Lovász and A. Schrijver, Geometric Algorithms and Combinatorial Optimization (Second ed.), Springer-Verlag, Berlin, 1993.
15. R.W. Irving, An efficient algorithm for the stable roommates problem, Journal of Algorithms 6 (1985) 577–595.
16. R.W. Irving and S. Scott, The stable fixtures problem - a many-to-many extension of stable roommates, Discrete Applied Mathematics 155 (2007) 2118–2129.
17. W. Kern and D. Paulusma, The new FIFA rules are hard: complexity aspects of sport competitions, Discrete Applied Mathematics 108 (2001) 317–323.
18. L.G. Khachiyan, A polynomial algorithm in linear programming, Soviet Mathematics Doklady 20 (1979) 191–194.
19. E. Lawler, *Combinatorial Optimization* Courier Dover Publ., 1976.
20. B. Klaus and F. Payot, Paths to Stability in the Assignment Problem, Cahier de recherches conomiques du DEEP No. 13.14, 2013.
21. T.C. Koopmans and M. Beckmann, Assignment problems and the location of economic activities, Econometrica 25 (1957) 53–76.
22. A.N. Letchford, G. Reinelt and D.O. Theis, Odd minimum cut sets and b-matchings revisited, SIAM Journal of Discrete Mathematics 22 (2008) 1480–1487.
23. D. Manlove, Algorithmics of Matching Under Preferences Series on Theoretical Computer Science vol. 2, World Scientific, 2013.

24. N. Megiddo, Combinatorial Optimization with rational objective functions, Math. of OR 4 (1979) 414–424.
25. A.E. Roth and M. Sotomayor, Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis, Cambridge University Press, 1990.
26. A.E. Roth and J.H. Vande Vate, Random paths to stability in two-sided matching, Econometrica 58 (1990) 1475–1480.
27. L.S. Shapley and M. Shubik, The assignment game I: the core, International Journal of Game Theory 1 (1972) 111–130.
28. M. Sotomayor, The multiple partners game, In Majumdar M. (ed) Equilibrium and dynamics: essays in honor of David Gale. Macmillan Press Ltd, New York, 1992.
29. M. Sotomayor, The lattice structure of the set of outcomes of the multiple partners assignment game, International Journal of Game Theory 28 (1999) 567–583.
30. M. Sotomayor, Connecting the cooperative and competitive structures of the multiple-partners assignment game, Journal of Economic Theory 134 (2007) 155–174.
31. W.T. Tutte, A short proof of the factor factor theorem for finite graphs, Canadian Journal of Mathematics 6 (1954) 347–352.