

# Cooperation in multiorganization matching

Laurent Gourvès<sup>1,2</sup>, Jérôme Monnot<sup>1,2</sup>, and Fanny Pascual<sup>3</sup>

1. CNRS, UMR 7024, F-75775 Paris, France

2. Université de Paris-Dauphine, LAMSADE, F-75775 Paris, France

3. LIP6, Université Pierre et Marie Curie - Paris 6

104 avenue du Président Kennedy, 75016 Paris, France

{laurent.gourves, monnot}@lamsade.dauphine.fr, fanny.pascual@lip6.fr

**Abstract.** We study a problem involving a set of organizations. Each organization has its own pool of clients who either supply or demand one unit of an indivisible product. Knowing the profit induced by each buyer-seller pair, an organization's task is to conduct such transactions within its database of clients in order to maximize the amount of the transactions. Inter-organizations transactions are allowed: in this situation, two clients from distinct organizations can trade and their organizations share the induced profit. Since maximizing the overall profit leads to unacceptable situations where an organization can be penalized, we study the problem of maximizing the overall profit such that no organization gets less than it can obtain on its own. Complexity results, an approximation algorithm and a matching inapproximation bound are given.

## 1 Introduction

We are given a two-sided assignment market  $(B, S, A)$  defined by a set of buyers  $B$ , a disjoint set of sellers  $S$ , and a nonnegative matrix  $A = (a_{ij})_{(i,j) \in B \times S}$  where  $a_{ij}$  represents a profit if the pair  $(i, j) \in B \times S$  trade. In this market products come in indivisible units, and each participant either supplies or demands exactly one unit. The units need not be alike and the same unit may have different values for different participants.

We study a problem involving a set of organizations  $\{O_1, \dots, O_q\}$  which forms a partition of the market. A buyer (resp. seller) is a client of exactly one organization. It is assumed that for every transaction  $(i, j)$ , organizations of  $i$  and  $j$  make an overall profit  $a_{ij}$  which is divided between the seller's organization and the buyer's organization as follows. The seller's organization receives  $p_s a_{ij}$  while the buyer's organization gets  $p_b a_{ij}$ , where  $p_s$  and  $p_b$  are fixed numbers between 0 and 1 and such that  $p_b + p_s = 1$ . Thus  $a_{ij}$  is a sort of commission that these two organizations divide according to  $p_b$  and  $p_s$ . We assume without loss of generality that  $0 \leq p_b \leq p_s \leq 1$ .

In this model, buyers and sellers do not make pairs by themselves, but these pairs are formed by their organizations. Each organization acts as a selfish agent who only knows its list of clients and only cares about its profit. Thus, each organization  $O_i$  shall maximize the weight of a matching on its own list of clients

(this task can be done in polynomial time). However the global profit can be better if transactions between clients of distinct organizations are allowed. This leads to a situation of cooperation where the agents accept to disclose their lists of clients by reporting them to a *trusted entity*. This trusted entity can conduct transactions between a buyer and a seller from distinct organizations, and of course, it can also do it for two clients of the same organization. The trusted entity shall maximize the collective profits. However, maximizing the collective profits by returning a maximum weight matching may lead to unacceptable situations: each organization is selfish so it does not want to cooperate if its profit is worse than it could obtain on its own. The optimization problem faced by the trusted entity is then to maximize the collective profit so that no organization is penalized.

### 1.1 The multiorganization assignment problem

The market is modelled with a weighted bipartite graph  $G = (B, S; E; w)$  and  $q$  sets (or organizations)  $O_1, \dots, O_q$  forming a partition of  $B \cup S$ . Every buyer (resp. seller) is represented by a vertex in  $B$  (resp.  $S$ ),  $E \subseteq B \times S$  is the edge set representing pairs and  $w : E \rightarrow \mathbb{R}_+$  is a nonnegative weight function. The subgraph of  $G$  induced by  $O_i$  is denoted by  $G_i$ . We have  $G_i = (B_i, S_i; E_i, w)$  where  $B_i = B \cap O_i$  and  $S_i = S \cap O_i$ . A set  $M \subseteq E$  is an *assignment* (or a *matching*) iff each vertex in  $(B, S; M, w)$  has degree at most one. The weight of an assignment  $M$  (i.e. the sum of the weights of its edges) is denoted by  $w(M)$ , and the profit of organization  $O_i$  in  $M$  is denoted by  $w_i(M)$  and defined as

$$w_i(M) = \sum_{\{[x,y] \in M: (x,y) \in B_i \times S\}} p_b w([x,y]) + \sum_{\{[x,y] \in M: (x,y) \in B \times S_i\}} p_s w([x,y])$$

where  $p_s$  and  $p_b$  are two nonnegative rational numbers such that  $p_s + p_b = 1$  and  $0 \leq p_b \leq p_s \leq 1$ .

We say that an edge whose endpoints are in the same organization (resp. in distinct organizations) is *internal* (resp. *shared*). The maximum weight matching of  $G$  reduced to its internal edges is denoted by  $\tilde{M}$ . Let  $\tilde{M}_i$  be the restriction of  $\tilde{M}$  to  $G_i$ . The *multiorganization assignment problem* (MOA for short) is to find a maximum weight matching  $M$  of  $G$  such that  $w_i(M) \geq w_i(\tilde{M})$  for all  $i \in \{1, \dots, q\}$ . Here  $w_i(\tilde{M})$  is what organization  $O_i$  can get on its own. Then  $\tilde{M}$  is a feasible solution to the MOA problem. As a notation,  $M^*$  denotes a maximum weight matching of  $G$  whereas  $M_{cont}^*$  is an optimum for MOA.

### 1.2 Applications

We give here two applications where MOA arises.

**The “agencies problem”** Each organization has its own pool of sellers ( $S$ ) and buyers ( $B$ ) who either supply or demand one unit of an indivisible product. Consider for example that organizations are real estate agencies. Each organization receives a commission on each transaction it deals, and its goal is to maximize its profit. Therefore each organization accepts the assignment given by a trusted entity if and only if its profit is at least equal to the profit it would have had without sharing its file with the other organizations. The overall aim is then to find an assignment which maximizes the total amount of transactions done, while guaranting that no organization decreases its profit by sharing its file.

**A scheduling example** Each organization (which can be a university, laboratory, etc.) owns unit tasks (given by its users), and several (possibly different) machines. During some given time slots, the machines are available to schedule the tasks of the users. Each user gives her preferences for a given machine and a given time slot. These preferences are represented by integers ( $a_{ij}$ ) between 0 (a task cannot be scheduled on this machine at this time), and a given upper bound. The goal of each organization is to maximize the average satisfaction of its users, represented by the sum of the satisfactions of its users divided by the number of users, in the returned assignment. Therefore an organization will accept a multiorganization assignment if and only if the average satisfaction of its users is at least as high as when the organization accepts only the tasks from its users. Here, an unmatched user’s satisfaction is 0. This corresponds to MOA when  $S$  is the set of users,  $B$  the set of couples (time slot, machine),  $p_s = 1$  and  $p_b = 0$ .

### 1.3 Related work

The multi-organization assignment problem is a variant of the old assignment problem (see [11] for a recent survey). Besides its combinatorial structure, MOA involves self-interested agents whose cooperation can lead to significant improvements but a solution is feasible only if it does not harm any local utility.

Non cooperative game theory studies situations involving several players whose selfish actions affect each other [10]. In Tucker’s prisoner’s dilemma, two players can either cooperate (C), i.e. stay loyal to the other prisoner, or defect (D), i.e. agree to testify against the other.

	C	D
C	3, 3	0, 4
D	4, 0	1, 1

the prisoner’s dilemma

A social optimum is reached if both play C but the situation where both prisoners defect is the only stable situation (a Nash equilibrium). In fact, the game designer of the prisoner’s dilemma filled the payoff matrix in way that any prisoner has incentive to defect. MOA models the opposite situation where the game designer tries to fill the payoff matrix such that each organization’s (weakly) dominant strategy is to cooperate, i.e. disclose its list of clients and follow the trusted entity. The game designer has to compute a Nash equilibrium (a stable matching) that optimizes the social welfare (total profit).

The maximum weight matching  $M^*$  is sometimes unstable because the organizations are selfish. Then, one has to consider a different optimum  $M_{cont}^*$  which is the maximum weight Nash equilibrium (no organization can increase its profit by using its own maximum weight matching instead of the solution returned by the trusted entity). Interestingly, a theoretical measure of this loss of profit due to the selfishness of the organizations exists. Known as the *price of stability* (PoS) [12, 1], it is defined as the (worst case) ratio between the most socially valuable state and the worth of the best Nash equilibrium. For MOA,  $PoS = w(M_{cont}^*)/w(M^*)$ .

MOA is also related to cooperative game theory [10]. A central issue in this field is to allocate the worth of a coalition to its members. Shapley and Shubik associate to any two-sided assignment market  $(B, S, A)$  a cooperative game with transferable utility (the assignment game) and show that its core is nonempty, and has a lattice structure [13].

MOA is close in spirit to other works which study, at an algorithmic level, how to make organizations cooperate. In [9], the authors study a scheduling problem involving several organizations. Each of them has a set of jobs to be completed as early as possible and its own set of processors. A selfish schedule is such that the processors only execute jobs of their owner. The authors propose an algorithm which returns schedules with good makespans and in which the organizations cooperate without being penalized. In [6, 5], the authors study the selfish distributed replication problem. This problem involves several nodes of a network whose task is to fetch electronic contents (objects) located at distant servers. Instead of taking an object from its server at each request, the nodes can save time by making a local copy. An intermediate strategy is to get an object from another node which is closer than the server. The optimization problem is to fill the (bounded) memory of each node in order to minimize the overall expected response time. Since an optimum solution can be unacceptable to selfish nodes (e.g. a node's memory is filled with objects that it rarely requests), the authors of [5] propose equilibrium placement strategies where no one is penalized.

#### 1.4 Contribution

We investigate the computational complexity of MOA in Section 2. In particular, we show that the problem is strongly **NP**-hard if the number of organizations is not fixed. It is weakly **NP**-hard for two organizations. A possible proof of strong **NP**-hardness for a fixed number of organizations is discussed and some pseudo-polynomial and polynomial cases are given as well. We provide an approximation algorithm with performance guarantee  $p_b$  and a matching proof of inapproximation in Section 3. We also show in this section that the price of stability of MOA is  $p_b$ . Section 4 is devoted to generalizations of MOA and also generalizations of the results of this article. We conclude in Section 5.

Our results apply for *any* values of  $p_s$  and  $p_b$  such that  $0 \leq p_b \leq p_s \leq 1$  and  $p_b + p_s = 1$ . Some proofs are omitted due to space limitations.

## 2 Complexity results

We prove that MOA is strongly **NP**-hard in the general case, even if the weights are polynomially bounded. We also show that the restriction of MOA to 2 organizations is weakly **NP**-hard. Next we show pseudopolynomial and polynomial cases.

### 2.1 Computationally hard cases

Given a positive profit  $P$  and an instance of MOA, the decision version asks whether the instance admits a matching  $M$  such that  $\forall_{i \in \{1, \dots, q\}} w_i(M) \geq w(\tilde{M}_i)$  and  $w(M) \geq P$ .

**Theorem 1.** *The decision version of MOA is strongly **NP**-complete.*

We make a reduction from 3-PARTITION which is strongly **NP**-complete (problem [SP15] in [3]).

**Theorem 2.** *The decision version of MOA is **NP**-complete, even if there are 2 organizations and the underlying graph is of maximum degree 2.*

*Proof.* Let  $p_s$  and  $p_b$  be two reals such that  $1 \geq p_s \geq p_b \geq 0$  and  $p_s + p_b = 1$ . The reduction is done from PARTITION: given a set  $\{a_1, \dots, a_n\}$  of  $n$  integers such that  $\sum_{i=1}^n a_i = 2W$ , decide whether  $J \subset \{1, \dots, n\}$  such that  $\sum_{j \in J} a_j = W$  exists. PARTITION is known to be **NP**-complete (problem [SP12] in [3]).

From  $I$ , instance of PARTITION, we build  $I'$ , instance of MOA by the following way:

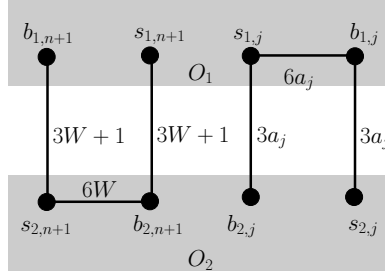
- we are given 2 organizations  $O_1$  and  $O_2$
- $O_1$  has  $n + 1$  sellers and  $n + 1$  buyers respectively denoted by  $s_{1,i}$  and  $b_{1,i}$  for  $i = 1, \dots, n + 1$
- $O_2$  has also  $n + 1$  buyers and  $n + 1$  sellers respectively denoted by  $b_{2,i}$  and  $s_{2,i}$  for  $i = 1, \dots, n + 1$
- The edge set of the underlying graph is given by  $\{[s_{1,n+1}, b_{2,n+1}]\} \cup \{[b_{2,n+1}, s_{2,n+1}]\} \cup \{[s_{2,n+1}, b_{1,n+1}]\} \cup \{[b_{1,i}, s_{1,i}], [s_{1,i}, b_{2,i}], [b_{1,i}, s_{2,i}] : i = 1, \dots, n\}$

The weights are defined by:

- $w([b_{1,i}, s_{1,i}]) = 6a_i$  and  $w([b_{2,i}, s_{1,i}]) = w([s_{2,i}, b_{1,i}]) = 3a_i$  for  $i = 1, \dots, n$
- $w([b_{2,n+1}, s_{2,n+1}]) = 6W$  and  $w([s_{1,n+1}, b_{2,n+1}]) = w([b_{1,n+1}, s_{2,n+1}]) = 3W + 1$

The underlying graph is made of a collection of  $n + 1$  disjoint paths of length 3. Figure 1 gives an illustration of this construction.

Organization  $O_1$  can make a profit  $w_1(\tilde{M}) = (p_s + p_b) \sum_{i=1}^n 6a_i = 12W$  if it works alone. The local profit of organization  $O_2$  is  $w_2(\tilde{M}) = (p_s + p_b)6W = 6W$ . Thus, globally, the weight of this matching is  $18W$ .



**Fig. 1.** The construction of  $I'$ .

We claim that  $I'$  admits a feasible assignment  $\hat{M}$  such that  $w(\hat{M}) \geq 18W + 2$  if and only if  $I$  admits a set  $J \subseteq \{1, \dots, n\}$  with  $\sum_{j \in J} a_j = W$ .

Let  $J$  be a subset of  $\{1, \dots, n\}$  such that  $\sum_{j \in J} a_j = W$  (and then,  $\sum_{j \notin J} a_j = W$ ). We build the assignment  $\hat{M}$  as follows:  $\hat{M} = \{[b_{2,j}, s_{1,j}], [s_{2,j}, b_{1,j}] : j \in J\} \cup \{[b_{1,j}, s_{1,j}] : j \notin J\} \cup \{[s_{1,n+1}, b_{2,n+1}], [b_{1,n+1}, s_{2,n+1}]\}$

Clearly, the cost of  $\hat{M}$  is given by  $w(\hat{M}) = 18W + 2$ . Now, let us verify that  $\hat{M}$  is a feasible solution. The local profit of organization  $O_1$  is  $(p_s + p_b) \sum_{j \notin J} 6a_j + (p_s + p_b) \sum_{j \in J} 3a_j + (p_s + p_b)(3W + 1) = 12W + 1 \geq w_1(\tilde{M})$  whereas the profit of organization  $O_2$  becomes  $(p_s + p_b) \sum_{j \in J} 3a_j + (p_s + p_b)(3W + 1) = 6W + 1 \geq w_2(\tilde{M})$ .

Conversely, let  $\hat{M}$  be a feasible assignment such that  $w(\hat{M}) \geq 18W + 2$ . The following property can be easily proved.

*Property 1.* Any feasible solution of MOA can be supposed maximal with respect to inclusion.

Now, remark that  $\hat{M}$  necessarily contains the edges  $[s_{1,n+1}, b_{2,n+1}]$  and  $[b_{1,n+1}, s_{2,n+1}]$  since on the one hand, the weight of any maximal matching on the graph induced by all vertices except  $\{s_{1,n+1}, s_{2,n+1}, b_{1,n+1}, b_{2,n+1}\}$  is  $12W$ , and on the other hand  $w([b_{2,n+1}, s_{2,n+1}]) = 6W$ . Thus,  $\hat{M}$  must contain some edges  $[b_{2,j}, s_{1,j}]$  or  $[b_{1,j}, s_{2,j}]$  in order to compensate the loss of edge  $[b_{2,n+1}, s_{2,n+1}]$ . Let  $J = \{j \leq n : [b_{2,j}, s_{1,j}] \in \hat{M}\}$ . By property 1,  $\hat{M}$  is completely described by  $\hat{M} = \{[b_{2,j}, s_{1,j}], [b_{1,j}, s_{2,j}] : j \in J\} \cup \{[b_{1,j}, s_{1,j}] : j \notin J\} \cup \{[s_{1,n+1}, b_{2,n+1}], [b_{1,n+1}, s_{2,n+1}]\}$ .

The profit of organization  $O_2$  is  $(p_s + p_b) \sum_{j \in J} 3a_j + (p_s + p_b)(3W + 1) = 3 \sum_{j \in J} a_j + 3W + 1$ . Since that profit is at least  $w_2(\tilde{M}) = 6W$ , we deduce that  $\sum_{j \in J} a_j \geq W - \frac{1}{3}$ . Finally,  $\sum_{j \in J} a_j$  must be an integer, so  $\sum_{j \in J} a_j \geq W$ . On the other hand, the profit of organization  $O_1$  is given by  $(p_s + p_b) \sum_{j \notin J} 6a_j + (p_s + p_b) \sum_{j \in J} 3a_j + (p_s + p_b)(3W + 1) = 6 \sum_{j=1}^n a_j - 3 \sum_{j \in J} a_j + 3W + 1$ . This quantity must be at least  $w_1(\tilde{M}) = 6 \sum_{j=1}^n a_j$ . Since  $\sum_{j \in J} a_j$  is an integer, we obtain  $\sum_{j \in J} a_j \leq W$ . In conclusion,  $\sum_{j \in J} a_j = W$  which means that  $\{a_1, \dots, a_n\}$  can be partitioned into two sets of weight  $W$ .  $\square$

Is MOA strongly **NP**-complete for two organizations? We were not able to answer but we can relate the question to another one stated more than 25 years ago and still open: Is the *exact weighted perfect matching* problem in bipartite graphs strongly **NP**-complete?

Given a graph whose edges have an integer weight and given a bound  $W$ , EXACTPM is to decide whether the graph contains a perfect matching  $M$  of total weight exactly  $W$  [2, 4, 7, 8]. Papadimitriou and Yannakakis [8] prove that EXACTPM is (weakly) **NP**-complete in bipartite graphs. Barahona and Pulleyblank [2] propose a pseudopolynomial algorithm in the case of planar graphs and Karzanov [4] gives a polynomial algorithm when the graph is either complete or complete bipartite and the weights are restricted to 0 or 1. Mulmuley, Vazirani and Vazirani [7] show that EXACTPM has a randomized pseudo-polynomial-time algorithm. However, the deterministic complexity of this problem remains unsettled, even for bipartite graphs. (Papadimitriou and Yannakakis conjectured that it is strongly **NP**-complete [8]).

EXACTPM is an auto-reducible problem, that is, finding a perfect matching of weight  $W$  is polynomially equivalent to decide whether such a matching exists.

Here, we prove that there is a Turing reduction from MOA when there are 2 organizations to EXACTPM. Thus, we conclude that if MOA with 2 organizations is strongly **NP**-complete then EXACTPM is also strongly **NP**-complete in bipartite graphs. Notice that this result also holds when there is a constant number of organizations.

**Proposition 1.** *If EXACTPM is polynomial in bipartite graphs when weights are polynomially bounded, then MOA with two organizations and weights polynomially bounded is polynomial for every values of  $p_s, p_b$  such that  $1 \geq p_s \geq p_b \geq 0$  and  $p_s + p_b = 1$ .*

*Proof.* Let  $p_b, p_s$  be two rational numbers such that  $1 \geq p_s \geq p_b \geq 0$  and  $p_s + p_b = 1$ , and let  $I = (G, w)$  be an instance of MOA with two organizations where  $G = (V, E)$ . W.l.o.g.  $w(e), p_s w(e)$  and  $p_b w(e)$  are integers for every edge  $e \in E$ . Moreover  $\forall e \in E, w(e) \leq P(|V|)$  for some polynomial  $P$ . Let  $R$  be the weight of a maximum weight matching of  $G$ . Consider the bipartite graph  $G' = (V', E')$  built from  $G$  by adding dummy vertices and edges with weight 0 such that any matching of  $G$  can be completed into a perfect matching of  $G'$  with the same value. Formally, we add a copy of  $K_{|S|, |B|}$  and each new  $B$ -vertex (resp.,  $S$ -vertex) is completely linked to the  $S$ -vertices (resp.,  $B$ -vertices) of  $G$ . Then, each shared edge  $e = [u, v] \in E$  is replaced by a path of length 3  $[u, u_e], [u_e, v_e], [v_e, v]$  where  $u_e, v_e$  are new vertices. Remark that either  $\{[u, u_e], [v_e, v]\}$  or  $\{[u_e, v_e]\}$  is included in a perfect matching of  $G'$ . Consider the weight function  $w'$  defined as  $w'(e) = (R+1)^3 w(e)$  if  $e$  is internal to organization  $O_1$  and  $w'(e) = (R+1)^2 w(e)$  if  $e$  is internal to organization  $O_2$ . Moreover, if  $e = [u, v] \in E$  is a shared edge then  $w'([u, u_e]) = (R+1)p_s w([u, v])$  if  $u \in S \cap O_1$  and  $w'([u, u_e]) = (R+1)p_b w([u, v])$  otherwise (i.e.  $u \in B \cap O_1$ ). We also set  $w'([v, v_e]) = p_s w([u, v])$  if  $u \in S \cap O_2$  and  $w'([v, v_e]) = p_b w([u, v])$  otherwise. The weight of each remaining edge of  $G'$  is 0. It is clear that  $G'$  is built within polynomial time and  $w'$  remains polynomially bounded. Let  $I' = (G', w')$ .

For any matching  $M$ , we denote by  $M_1$  (resp.,  $M_2$ ) the restriction of  $M$  to organization  $O_1$  (resp.,  $O_2$ ) and by  $M_{shared}$  the set of shared edges of  $M$ . Denote by  $W_1$  (resp.,  $W_2$ ) the contribution of the shared edges of  $M$  to the profit of organization  $O_1$  (resp.,  $O_2$ ). We have  $w(M_{shared}) = W_1 + W_2$  since  $p_s + p_b = 1$ .

We claim that  $w(M) = w(M_1) + w(M_{shared}) + w(M_2)$  if and only if there exists a matching of  $I'$  with weight  $W = (R+1)^3w(M_1) + (R+1)^2w(M_2) + (R+1)W_1 + W_2$ . Moreover,  $M$  is a feasible solution to MOA iff  $w(M_i) + W_i \geq w_i(\tilde{M})$  for  $i = 1, 2$ .

One direction is trivial. So, let  $M'$  be a matching of  $I'$  with value  $w'(M') = W = (R+1)^3A + (R+1)^2B + (R+1)C + D$ . By the choice of  $R$ , we must get  $w(M'_1) = A$ ,  $w(M'_2) = B$  and  $w(M'_{shared}) = C + D$ , where  $C$  (resp.,  $D$ ) is the contribution of  $M'_{shared}$  to the profit of organization  $O_1$  (resp.,  $O_2$ ). The profit of organization  $O_1$  (resp.  $O_2$ ) according to  $M'$  is  $A + C$  (resp.  $B + D$ ).

In conclusion by applying at most  $R^4$  times the polynomial algorithm for EXACTPM, we find an optimal solution of MOA. By an exhaustive search, we try all values of  $A, B, C, D$  at most equal to  $R$  such that  $A + C \geq w_1(\tilde{M})$  and  $B + D \geq w_2(\tilde{M})$ .  $\square$

**Proposition 2.** *MOA with a constant number of organizations can be solved in pseudopolynomial time when the underlying graph has a maximum degree 2.*

Actually, dynamic programming can improve the time complexity given in Proposition 2.

## 2.2 Polynomial cases

MOA is trivially polynomial when there is a unique organization or when the underlying graph is of maximum degree 1. Furthermore an exhaustive search can efficiently solve the problem if the underlying graph  $G = (V, E)$  contains  $\mathcal{O}(\log |E|)$  shared edges. Let  $\text{MOA}_{0,1}$  be the subcase where  $w([i, j]) \in \{0, 1\}$  for all  $(i, j) \in B \times S$ . We prove that an optimum to  $\text{MOA}_{0,1}$  is a maximum cardinality assignment of the underlying graph though a maximum cardinality assignment is not necessarily a solution of  $\text{MOA}_{0,1}$ .

**Theorem 3.**  *$\text{MOA}_{0,1}$  is polynomial.*

*Proof.* Let  $M$  be an assignment on an unweighted bipartite graph  $G = (B, S; E)$ . Recall that a path in  $G$  is *alternating* with respect to  $M$  if it alternates edges of  $M$  and edges of  $E \setminus M$ . Furthermore, an alternating path  $\pi$  is *augmenting* if no edge of  $M$  is incident to its extremal nodes. The word ‘‘augmenting’’ means that  $(M \setminus \pi) \cup (\pi \setminus M)$  is a matching of size  $|M| + 1$ . It is well known that  $M$  is of maximum size on  $G$  if  $G$  does not admit any augmenting alternating path with respect to  $M$ .

Let  $I$  be an instance of  $\text{MOA}_{0,1}$  defined upon  $G$ . Let  $\hat{M}$  be an optimal matching built as follows. Start with the feasible matching  $\tilde{M}$  and increase its size with augmenting alternating paths while it is possible.

Let  $\hat{M}^j$  be the matching produced at step  $j$ . We suppose that  $t$  steps are needed to reach  $\hat{M}$ . Hence,  $\hat{M}^0 = \tilde{M}$  and  $\hat{M}^t = \hat{M}$ . We mainly prove

$$w_i(\hat{M}^{j+1}) \geq w_i(\hat{M}^j), \forall i \in \{1, \dots, q\} \quad (1)$$

for all  $j \in \{0, \dots, t-1\}$ . This inequality states that the use of an augmenting alternating path cannot deteriorate the profit of any organization.

Given  $v \in V$  and a matching  $M$ , let  $c(v, M)$  be the *contribution* of  $v$  to the profit of its organization in  $M$ :

$$c(v, M) = \begin{cases} p_s & \text{if } v \in S \text{ and an edge of } M \text{ is incident to } v \\ p_b & \text{if } v \in B \text{ and an edge of } M \text{ is incident to } v \\ 0 & \text{otherwise} \end{cases}$$

Let  $V'$  be the vertices of  $\pi'$ , the augmenting alternating path such that  $\hat{M}^{j+1} = (\hat{M}^j \setminus \pi') \cup (\pi' \setminus \hat{M}^j)$ . We deduce that

$$w_i(\hat{M}^{j+1}) - w_i(\hat{M}^j) = \sum_{v \in V'} c(v, \hat{M}^{j+1}) - c(v, \hat{M}^j) \quad (2)$$

for all  $i \in \{1, \dots, q\}$ . One can observe that  $c(v, \hat{M}^j) = c(v, \hat{M}^{j+1})$  if  $v \in V'$  and  $v$  is not an extremal node of  $\pi'$ . Indeed, a buyer  $b \in V'$  matched with a seller  $s \in V'$  in  $\hat{M}^j$  is still matched in  $\hat{M}^{j+1}$  but with another seller. Similarly, a seller  $s \in V'$  matched with a buyer  $b \in V'$  in  $\hat{M}^j$  is still matched in  $\hat{M}^{j+1}$  but with another buyer. If  $v \in S \cap V'$  (resp.  $v \in B \cap V'$ ) and  $v$  is an extremal node of  $\pi'$  then  $c(v, \hat{M}^j) = 0$  and  $c(v, \hat{M}^{j+1}) = p_s$  (resp.  $c(v, \hat{M}^j) = 0$  and  $c(v, \hat{M}^{j+1}) = p_b$ ). Hence,

$$c(v, \hat{M}^{j+1}) - c(v, \hat{M}^j) \geq 0 \quad (3)$$

for all  $v \in V$  because  $p_s \geq p_b \geq 0$ . Using (2) and (3) we obtain  $w_i(\hat{M}^{j+1}) - w_i(\hat{M}^j) \geq 0$  for all  $i \in \{1, \dots, q\}$ .  $\hat{M}$  is a feasible assignment because  $w_i(\hat{M}^t) \geq w_i(\hat{M}^{t-1}) \geq \dots \geq w_i(\hat{M}^0) = w(\tilde{M}_i)$  for all  $i \in \{1, \dots, q\}$ . In addition,  $w(\hat{M}) = w(M^*)$  because the algorithm stops when no augmenting alternating path exists. In conclusion,  $\hat{M}$  is optimal because  $w(M^*) \geq w(M_{cont}^*)$ .  $\square$

### 3 Approximation

Recall that  $p_s$  and  $p_b$  are any values such that  $0 \leq p_b \leq p_s \leq 1$  and  $p_s + p_b = 1$ . We start by the following property.

*Property 2.*  $w_i(M^*) \geq p_b w(\tilde{M}_i)$ , and this bound is asymptotically tight.

Let us consider algorithm APPROX given below.

**Theorem 4.** APPROX is a  $p_b$ -approximate algorithm for MOA, and this bound is asymptotically tight.

**Algorithm APPROX**

- 
- Construct the graph  $G' = (V', E')$  from  $G = (V, E)$  as follows:  $V' = V$ , and  $E' = E$ , except that the weights of the edges are modified: for each edge  $[u, v]$  such that  $u$  belongs to organization  $O_i$  and  $v$  belongs to organization  $O_j$ ,  $w'([u, v]) = w([u, v])$  if  $u$  and  $v$  belong to the same organization ( $i = j$ ), and otherwise  $w'([u, v]) = p_b w([u, v])$ .
  - Return a maximum weight matching of  $G'$ .
- 

*Proof.* Let  $p_s, p_b$  be two numbers such that  $1 \geq p_s \geq p_b \geq 0$  and  $p_s + p_b = 1$ . Let  $M$  be a matching returned by algorithm APPROX on graph  $G$ . We first show that the profit of each organization  $O_i$  in  $M$  is at least  $w(\tilde{M}_i)$ . Thus  $M$  is a solution of MOA.

Let  $M^{int(i)}$  be the set of edges of  $M$  such that both endpoints belong to  $O_i$ , and let  $M^{ext(i)}$  be the set of edges of  $M$  such that exactly one endpoint belongs to  $O_i$ . Since  $M$  is a maximum weight matching of  $G'$ ,  $w'(M^{int(i)}) + w'(M^{ext(i)}) \geq w(\tilde{M}_i)$ , otherwise we could have a matching with a larger weight by replacing the edges of  $(M^{int(i)} \cup M^{ext(i)})$  in  $M$  by the edges of  $\tilde{M}_i$ . Thus the profit of  $O_i$  is at least  $w(M^{int(i)}) + p_b w(M^{ext(i)}) = w'(M^{int(i)}) + w'(M^{ext(i)}) \geq w(\tilde{M}_i) = w_i(\tilde{M})$ .

Let us now show that APPROX is  $p_b$ -approximate. The edges of  $G'$  are the same as the ones of  $G$ , except that the weight of some of them has been multiplied by  $p_b < 1$ . Thus  $M$ , which is a maximum weight matching of  $G'$ , has a weight  $w(M) \geq p_b w(M^*) \geq p_b w(M_{cont}^*)$ .

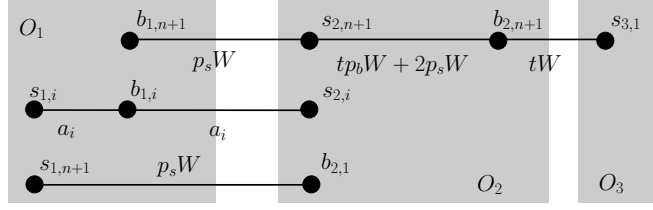
Let us show that this bound is asymptotically tight, by considering the following instance. Here, we assumed  $p_b > 0$ . Recall that  $p_b \leq 1/2$  since  $1 \geq p_s \geq p_b \geq 0$ . Let  $\varepsilon > 0$  such that  $\varepsilon < 1/p_b - 1$ . There are two organizations, organization  $O_1$ , which owns two vertices  $b_1$  and  $s_1$ , linked by an edge of weight 1, and organization  $O_2$ , which owns two vertices  $b_2$  and  $s_2$ , linked by an edge of weight 1. There are two shared edges, between  $b_1$  and  $s_2$ , and between  $b_2$  and  $s_1$ : both edges have weight  $\frac{1}{p_b} - \varepsilon$ . Algorithm APPROX returns the matching  $M = \{[b_1, s_1], [b_2, s_2]\}$  with weight 2 in  $G'$  because the weight of  $\{[b_1, s_2], [b_2, s_1]\}$  in  $G'$  is  $2(1 - p_b \varepsilon) < 2$ . The optimal solution would have been  $M_{cont}^* = \{[b_1, s_2], [b_2, s_1]\}$ . The ratio between the weights of these two solutions is  $\frac{w(M)}{w(M_{cont}^*)} = \frac{2}{2/p_b - 2\varepsilon}$ , which tends towards  $p_b$  when  $\varepsilon$  tends towards 0.  $\square$

Theorem 4 implies that the price of stability of MOA defined as  $w(M_{cont}^*)/w(M^*)$  is at least  $p_b$ . In fact, we are able to prove that PoS =  $p_b$ .

**Proposition 3.** *The price of stability is  $p_b$ .*

*Proof.* It follows from Theorem 4 that  $w(M_{cont}^*)/w(M^*) \geq p_b$  since APPROX returns a matching  $M$  such that  $w(M_{cont}^*) \geq w(M) \geq p_b w(M^*)$ .

Let us now show that this bound is tight. There are two organizations: organization  $O_1$ , which owns two vertices  $b_1$  and  $s_1$ , linked by an edge of weight  $W_1$ , and organization  $O_2$ , which owns one vertex  $s_2$ , linked to  $b_1$  by a link of weight



**Fig. 2.** The instance  $I_t$  resulting from the above reduction

$W_2$ . Suppose that  $W_1 = \varepsilon$  such that  $0 < \varepsilon < 1$  and  $W_2 = 1$  when  $p_b = 0$ . The ratio  $\frac{w(M_{cont}^*)}{w(M^*)} = \varepsilon$ , tends towards  $0 = p_b$  when  $\varepsilon$  tends towards 0. Suppose that  $W_1 = 1$  and  $W_2 = 1/p_b - \varepsilon$  such that  $0 < \varepsilon < 1/p_b - 1$  when  $p_b > 0$ . The ratio  $\frac{w(M_{cont}^*)}{w(M^*)} = \frac{p_b}{1 - \varepsilon p_b}$ , tends towards  $p_b$  when  $\varepsilon$  tends towards 0.  $\square$

We can prove that Theorem 4 is best possible if  $\mathbf{P} \neq \mathbf{NP}$ , i.e. we cannot obtain a  $(p_b + \varepsilon)$ -approximation for all  $\varepsilon > 0$ . Actually, we prove a slightly stronger result where  $n$  denotes the number of vertices.

**Theorem 5.** *For any polynomial  $P$ , it is  $\mathbf{NP}$ -hard to obtain a  $(p_b + \frac{1}{\Theta(2^{P(n)})})$ -approximation for MOA where at least three organizations are involved.*

*Proof.* We describe a gap reduction. We start with an instance of PARTITION given by a set of  $n$  integers  $\{a_1, \dots, a_n\}$  such that  $\sum_{i=1}^n a_i = 2W$ . For any real  $t > 1$ , we construct an instance  $I_t$  of MOA as follows:

- we are given 3 organizations  $O_1$ ,  $O_2$  and  $O_3$ .
- $O_1$  has  $n + 1$  buyers and  $n + 1$  sellers respectively denoted by  $b_{1,i}$  and  $s_{1,i}$  for  $i = 1, \dots, n + 1$ .
- $O_2$  has 2 buyers denoted by  $b_{2,1}, b_{2,n+1}$  and  $n + 1$  sellers denoted by  $s_{2,i}$  for  $i = 1, \dots, n + 1$ .
- $O_3$  has one seller  $s_{3,1}$ .
- The edge set of the underlying graph is  $\{[s_{1,i}, b_{1,i}], [b_{1,i}, s_{2,i}] : i = 1, \dots, n\} \cup \{[s_{1,n+1}, b_{2,1}]\} \cup \{[b_{1,n+1}, s_{2,n+1}], [s_{2,n+1}, b_{2,n+1}], [b_{2,n+1}, s_{3,1}]\}$

The weights are given by:

- $w([s_{1,i}, b_{1,i}]) = w([b_{1,i}, s_{2,i}]) = a_i$  for  $i = 1, \dots, n$ .
- $w([s_{1,n+1}, b_{2,1}]) = p_s W$ ,  $w([b_{1,n+1}, s_{2,n+1}]) = p_s W$ ,  $w([s_{2,n+1}, b_{2,n+1}]) = t p_b W + 2 p_s W$ , and  $w([b_{2,n+1}, s_{3,1}]) = t W$ .

An illustration of this construction is given in Figure 2.

If  $t = \mathcal{O}(2^{P(|V|)})$  where  $|V| = 3n + 6$  is the order of the underlying graph, then it is not difficult to see that the above construction is given within polynomial time.

The profits the organizations can make on their own are respectively  $w_1(\tilde{M}) = (p_s + p_b) \sum_{i=1}^n a_i = 2W$ ,  $w_2(\tilde{M}) = (p_s + p_b)(tp_bW + 2p_sW) = tp_bW + 2p_sW$  and  $w_3(\tilde{M}) = 0$ .

We prove that there are only two distinct values for the optimal value of MOA, that are  $OPT(I_t) = tp_bW + 3p_sW + 2W$  or  $OPT(I_t) = tW + 2p_sW + 2W$ , and  $OPT(I_t) = tW + 2p_sW + 2W$  if and only if  $\{a_1, \dots, a_n\}$  admits a partition.

Observe that  $tW + 2p_sW + 2W > tp_bW + 3p_sW + 2W$  if and only if  $t > 1$  since  $p_b = 1 - p_s$  and  $p_s > 0$ . Let  $M_{cont}^*$  be an optimal solution of MOA (with value  $OPT(I_t)$ ). Let us consider two cases:

**Case**  $[s_{2,n+1}, b_{2,n+1}] \in M_{cont}^*$ . An optimal solution can be described by

$$\{[s_{1,i}, b_{1,i}] : i = 1, \dots, n\} \cup \{[s_{1,n+1}, b_{2,1}], [s_{2,n+1}, b_{2,n+1}]\}.$$

Actually,  $[s_{1,n+1}, b_{2,1}] \in M_{cont}^*$  because  $M_{cont}^*$  is maximal by Property 1 (cf page 6). Moreover, the weight of any maximal matching on the graph induced by  $\{s_{1,i}, b_{1,i}, s_{2,i} : i = 1, \dots, n\}$  has the same value  $2W$ . In this case, we get  $OPT(I_t) = tp_bW + 3p_sW + 2W$ .

**Case**  $[s_{2,n+1}, b_{2,n+1}] \notin M_{cont}^*$ . Edges  $\{[b_{1,n+1}, s_{2,n+1}], [b_{2,n+1}, s_{3,1}], [s_{1,n+1}, b_{2,1}]\}$  belong to  $M_{cont}^*$  by Property 1. The contribution of these 3 edges to the profit of  $O_2$  is  $p_s w([b_{1,n+1}, s_{2,n+1}]) + p_b w([b_{2,n+1}, s_{3,1}]) + p_b w([s_{1,n+1}, b_{2,1}]) = tp_bW + p_sW < tp_bW + 2p_sW = w([s_{2,n+1}, b_{2,n+1}])$  since  $p_s > 0$ . Hence, a subset of shared edges between  $O_1$  and  $O_2$  must belong to  $M_{cont}^*$ . Let  $J^* = \{j \leq n : [b_{1,j}, s_{2,j}] \in M_{cont}^*\}$  be this subset. Then,  $M_{cont}^*$  is entirely described by  $\{[b_{1,n+1}, s_{2,n+1}], [b_{2,n+1}, s_{3,1}], [s_{1,n+1}, b_{2,1}]\} \cup \{[b_{1,j}, s_{2,j}] : j \in J^*\} \cup \{[s_{1,j}, b_{1,j}] : j \notin J^*\}$ .

To be feasible,  $M_{cont}^*$  must satisfy  $w_1(M_{cont}^*) \geq w(\tilde{M}_1)$ , i.e.  $\sum_{j \notin J^*} a_j + p_b \sum_{j \in J^*} a_j + (p_s + p_b)p_sW \geq \sum_{j=1}^n a_j$  from which we deduce  $W \geq \sum_{j \in J^*} a_j$  because  $p_b = 1 - p_s$  and  $p_s > 0$ .  $M^*$  must also satisfy  $w_2(M_{cont}^*) \geq w(\tilde{M}_2)$ , i.e.  $p_s \sum_{j \in J^*} a_j + (p_s + p_b)p_sW + tp_bW \geq tp_bW + 2p_sW$ , which is equivalent to  $\sum_{j \in J^*} a_j \geq W$ . Then, we obtain  $\sum_{j \in J^*} a_j = \sum_{j \notin J^*} a_j = W$ . On the one hand  $OPT(I_t) = tW + 2p_sW + 2W$  and on the other hand  $\{a_1, \dots, a_n\}$  has a partition given by  $J^*$ .

Conversely, if  $\{a_1, \dots, a_n\}$  admits a partition then it is not difficult to prove that  $OPT(I_t) = tW + 2p_sW + 2W$ .

Now, assume that there is a  $(p_b + \frac{1}{c2^{P(|V|)}})$ -approximation of MOA given within polynomial time for some  $c > 0$ . Consider  $t_0 = 5c2^{P(|V|)}$  and let  $apx(I_{t_0})$  denote the value of the approximate solution on instance  $I_{t_0}$ .

- $\{a_1, \dots, a_n\}$  does not admit a partition. One has  $OPT(I_{t_0}) = 5c2^{P(|V|)}p_bW + 3p_sW + 2W$  and then  $apx(I_{t_0}) \leq 5c2^{P(|V|)}p_bW + 3p_sW + 2W$ .
- $\{a_1, \dots, a_n\}$  admits a partition. We have  $OPT(I_{t_0}) = 5c2^{P(|V|)}W + 2p_sW + 2W$ . Since  $apx(I_{t_0}) \geq (p_b + \frac{1}{c2^{P(|V|)}})OPT(I_{t_0})$  by hypothesis and  $p_s \leq 1$ , we deduce  $apx(I_{t_0}) > 5W + 5c2^{P(|V|)}p_bW \geq 5c2^{P(|V|)}p_bW + 3p_sW + 2W$ .

In conclusion,  $apx$  allows us to distinguish within polynomial time whether  $\{a_1, \dots, a_n\}$  has a partition or not, which is impossible if  $\mathbf{P} \neq \mathbf{NP}$ .  $\square$

## 4 Generalizations

### 4.1 Relaxation of the selfishness of the organizations

Suppose that each organization  $O_i$  accepts a proposed global matching if its own profit is at least  $w(\tilde{M}_i)/x$  where  $x \geq 1$  is fixed. This means that each organization accepts to divide by  $x$  the profit it would have without sharing its file with the other organizations. The problem, denoted by  $\text{MOA}(x)$  is then to find a maximum weight matching  $M$  such that  $w_i(M) \geq w(\tilde{M}_i)/x$  for all  $i \in \{1, \dots, q\}$ . Let  $M_{cont(x)}^*$  denote such a maximum weight matching.

If  $x = 1$ , an organization does not accept to reduce its profit, and this problem is the one stated in the Introduction. If  $x \geq 1/p_b$ , the organizations accept to divide their profits by  $1/p_b$ . Property 2 page 9 shows that in a maximum weight matching  $M^*$ , the profit of organization  $O_i$  is at least  $p_b w(\tilde{M}_i)$ . Thus  $M_{cont(x)}^* = M^*$ . Our aim is now to solve  $\text{MOA}(x)$  for  $1 \leq x < 1/p_b$ . With a slight modification of the proof of Theorem 1, we can show that this problem is strongly **NP**-hard for each value  $x$  smaller than  $1/p_b$ . One can also extend **APPROX** to a slightly modified algorithm<sup>1</sup>  $\text{APPROX}(x)$  and prove that it is  $(x p_b)$ -approximate algorithm for  $\text{MOA}(x)$  and this bound is tight. In addition, the price of stability is  $x p_b$  for this generalization.

### 4.2 General graphs

One can extend **MOA** to general graphs when  $p_s = p_b = 1/2$ . In this case, the distinction between buyers and sellers is lost. For example, the problem has the following application: Numerous web sites offer to conduct home exchanges during holidays. The concept is simple, instead of booking expensive hotel rooms, pairs of families agree to swap their houses for a vacation. We model the situation with a graph  $G = (V, E)$  whose vertices are candidates for house exchange. The vertex set is partitioned into  $q$  sets/organizations  $O_1 \dots O_q$ . Vertices within an organization are its clients. Every edge  $[a, b] \in E$  has a weight  $w([a, b])$  representing the satisfaction of candidates  $a$  and  $b$  if they swap. Pairs are formed by the organizations which only care about the satisfaction of their clients. In case of a mixed-organizations exchange  $[a, b]$ , it is assumed that the satisfaction of both participants is  $w([a, b])/2$ . The problem is to maximize the collective satisfaction while no organization is penalized.

Theorems 3 to 5 and Proposition 3 (where  $p_b$  is replaced by  $1/2$ ) hold for general graphs since the proofs do not use the fact that  $G$  is bipartite.

## 5 Conclusion

We studied cooperation, at an algorithmic level, between organizations. We showed that the price of stability is  $p_b$ , and we studied the complexity of **MOA**. We presented polynomial cases, and showed that the problem is **NP**-hard in the

<sup>1</sup> The weight of shared edges is multiplied by  $x p_b$  instead of  $p_b$ .

general case. We also gave an approximation algorithm, matching the inapproximation bound when there are at least 3 organizations. There remains some open problems: is it possible to have an algorithm with a better approximation ratio when there are two organizations? Is this problem strongly NP-hard in this case (we notice that this problem is related to the open Exact Perfect Matching problem)? When we consider that each organization accepts a solution if it does not reduce its profit by a factor larger than  $x$ , is it possible to get an algorithm with an approximation ratio better than  $x p_b$ ? An interesting direction would also be to study fairness issues in this problem. For example, among all the solutions of the same quality, return the one which maximizes the minimum  $w_i(M_{cont}) - \tilde{M}_i$ , that is the minimum increase of profit of the organizations.

## References

1. E. Anshelevich, A. Dasgupta, J. M. Kleinberg, É. Tardos, T. Wexler and T. Roughgarden. *The Price of Stability for Network Design with Fair Cost Allocation*. In Proc. of FOCS 2004, pp. 295-304, 2004.
2. F. Barahona and W.R. Pulleyblank. *Exact arborescences, matchings, and cycles*, in Discrete Appl. Math. vol. 16, pp. 91-99, 1987.
3. M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, New York, 1979.
4. A.V. Karzanov. *Maximum matching of given weight in complete and complete bipartite graphs*, In Cybernetics, vol. 23, pp. 8-13, 1987 (translation from Kibernetika, vol. 1, pp. 711, 1987).
5. N. Laoutaris, O. Telelis, V. Zissimopoulos and I. Stavrakakis. *Distributed Selfish Replication*, IEEE Trans. Parallel Distrib. Syst., vol 17(12), pp. 1401-1413, 2006.
6. A. Leff, L. Wolf, and P. S. Yu. *Efficient LRU-based buffering in a LAN remote caching architecture*, IEEE Trans. Parallel Distrib. Syst., vol. 7(2), pp. 191-206, 1996.
7. K. Mulmuley, U. Vazirani and V.V. Vazirani. *Matching is as easy as matrix inversion*, In Combinatorica, vol. 7, pp. 105-113, 1987.
8. C.H. Papadimitriou and M. Yannakakis. *The complexity of restricted spanning tree problems*, In J. ACM, vol. 29, pp. 285-309, 1982.
9. F. Pascual, K. Rzdca and D. Trystram. *Cooperation in Multi-organization Scheduling*, In Proc. of Euro-Par 07, LNCS 4641, pp. 224-233, 2007.
10. M. Osborne and A. Rubinstein. *A Course in Game Theory*, MIT Press, 1994.
11. D. W. Pentico. *Assignment problems: A golden anniversary survey*. EJOR, vol. 176, pp. 774-793, 2007.
12. A.S. Schultz and N. Stier Moses. *On the performance of user equilibria in traffic networks*, In Proc. of SODA 03, pp. 86-87, 2003.
13. L.S. Shapley and M. Shubik. *The assignment game I: The core*, International Journal of Game Theory, vol. 1, pp. 111-130, 1972.