Note on an Auction Procedure for a Matching Game in Polynomial Time

Winfried Hochstättler[1], Hui Jin[2], and Robert Nickel[1]

[1] Department of Mathematics FernUniversität in Hagen, D-58084 Hagen
[2] Department of Mathematics Brandenburg Technical University Cottbus, D-03044 Cottbus

Abstract. We derive a polynomial time algorithm to compute a stable solution in a mixed matching market from an auction procedure as presented by Eriksson and Karlander [5]. As a special case we derive an $\mathcal{O}(nm)$ algorithm for bipartite matching that does not seem to have appeared in the literature yet.

1 Introduction

In past years scientists from different fields such as game theory, economics, computer science, and combinatorial optimization have focused on the problem of two-sided matching markets where there are two finite and disjoint sets of agents, P and Q, that are to be matched in pairs consisting of one P-agent and one Q-agent. Two famous models of two-sided matching markets are the marriage model of Gale and Shapley [9] and the assignment game of Shapley and Shubik [17].

In the marriage model (see e. g. [13, 11]) the two sets of agents are usually referred to as the eligible marriage candidates in some small village. Each agent has preferences over the agents of the opposite set. A marriage is called *stable* when there is no pair which is not matched but prefers each other over their partners. Using the algorithm named "men propose – women dispose" Gale and Shapley [9] proved the existence of such a stable marriage when the preference lists are strict.

In the assignment game money plays a prominent role. It is modeled as a continuous variable. A matching and an allocation of its weight to the players compose a solution of the assignment game which is called *outcome*. By a stable outcome we mean a solution where no pair gets allocated less than the weight of its connecting edge. Shapley and Shubik [17] observed that stable outcomes coincide with the primal-dual pairs of solutions of the maximum weighted matching linear program, thereby showing the existence of a stable outcome. However, algorithms and complexity issues of game theoretical solution concepts have raised attention only recently (see e. g. Deng and Papadimitriou [3], Faigle et al. [6], Deng et al. [4]) and the classical algorithm for weighted bipartite matching, namely the Hungarian Method of Kuhn [14], is not as prominent in game theory as it is in combinatorial optimization.

Quite similar results such as the non-emptiness of the set of stable matchings and the lattice structure of the core have been established for these two models. To find a satisfactory explanation for the similarities in behavior between the two models Roth and Sotomayor [16] themselves offered a first model containing both the two old models as special cases and showed that its set of stable solutions, if it is non-empty, also has the lattice property under certain conditions. Eriksson and Karlander [5] modified this model to the *RiFle assignment game*, another common generalization of the two old models, and gave an algorithmic proof of the non-emptiness of its set of stable solutions. This algorithm computes a stable solution not in polynomial time but in pseudopolynomial time. For the classical special cases, it coincides with "men propose – women dispose", respectively with the "exact" auction procedure of [2]. The existence of stable solutions in presence of irrational data is proved by Eriksson and Karlander only via arguments from non-standard analysis.

We consider the model of Eriksson and Karlander [5]. A careful analysis of their algorithm reveals that a proper implementation solves the problem in $\mathcal{O}(n^4)$. This implementation was developed in parallel with [12] where we derive another polynomial time algorithm, to compute a stable solution for the same model, from the key lemma in Sotomayor [18]. Both algorithms run in $\mathcal{O}(n^4)$, where $2n$ is the number of players and n^2 is the size of a problem instance.

In the next section we briefly introduce the model and its notion of stability. Then we design a polynomial time algorithm to compute a stable solution in Section 3. Finally, we discuss the behavior of the algorithm in the special cases of Stable Matching, Assignment Game and cardinality matching and summarize differences from and similarities to the algorithm from [12].

2 Notation

We have two sets of players P (firms indexed by i) and Q (workers indexed by j) w.l.o.g. satisfying $|P| = |Q| =: n$. Let furthermore $P \cup Q$ be partitioned into *flexible* players (F) and *rigid* players (R). Consider the complete bipartite graph on $P \dot{\cup} Q$. An edge (i, j) is called *rigid* if one of i or j is in R and *flexible*, otherwise. For each edge (i, j) there are nonnegative numbers a_{ij} and b_{ij}. The sum $a_{ij} + b_{ij}$ is the *productivity* of a cooperation between i and j. A pair of functions $u : P \to \mathbb{R}$ and $v : Q \to \mathbb{R}$ is called a *payoff*. If i cooperates with j and (i, j) is a free edge the productivity can be freely divided into payoffs u_i and v_j while $u_i = a_{ij}$ and $v_j = b_{ij}$ must hold if (i, j) is a rigid edge.

Definition 1. *A payoff* (u, v) *is called* stable *if for any edge* $(i, j) \in P \times Q$ *we have*

(i) $u_i + v_j \geq a_{ij} + b_{ij}$ *if* (i, j) *is a free edge and*
(ii) $u_i \geq a_{ij}$ *or* $v_j \geq b_{ij}$ *if* (i, j) *is a rigid edge.*

A stable outcome *is a stable payoff* (u, v) *together with a bijective map* $\mu : P \to Q$ *(denoted by* $(u, v; \mu)$*) so that*

(iii) $u_i \geq 0$ *and* $v_j \geq 0$ *for all* $(i,j) \in P \times Q$.
(iv) $u_i + v_j = a_{ij} + b_{ij}$ *for* $\mu(i) = j$ *and* $\{i,j\} \subseteq F$.
(v) $u_i = a_{ij}$ *and* $v_j = b_{ij}$ *for* $\mu(i) = j$ *and* $\{i,j\} \cap R \neq \emptyset$.

Let $\mu : P \to Q$ be a map. If $\mu(i) = j$ then we say i *proposes to* j. A proposal is called *free* or *rigid* if the corresponding edge is free resp. rigid. A firm i (a worker j) is called *mapped* if $i \in \mu^{-1}(Q)$ (resp. $j \in \mu(P)$) and *unmapped*, otherwise. If there are firms i_1, i_2 so that $\mu(i_1) = \mu(i_2) = j$ then j is called *doubly mapped*. We denote by

Q_U the set of unmapped workers,
$Q_{2\mu}$ the set of doubly mapped workers,
Q_R the set of workers that have a rigid proposal, and by
Q_{2R} the set of workers with at least 2 rigid proposals.

Let furthermore

$$f_{ij}^{(v,\mu)} := \begin{cases} a_{ij} + b_{ij} - v_j & \text{if } (i,j) \text{ is a free edge} \\ a_{ij} & \text{if } (i,j) \text{ is rigid and } v_j < b_{ij} \\ a_{ij} & \text{if } (i,j) \text{ is rigid and } v_j = b_{ij} \text{ and } \mu(i) = j \\ 0 & \text{otherwise} \end{cases}$$

define the possible profit of i from j if j receives v_j.

The strategy of the algorithm is the following: The map μ always defines stable relations but is not necessarily injective. In the course of the algorithm we will try and increase $|\mu(P)|$, keeping stability of the relations, until the map is injective. The procedure to increase $|\mu(P)|$ acts on the *augmentation digraph* $G^{(v,\mu)} = (P \cup Q, A)$ with backward arcs (j,i) for $\mu(i) = j$ and forward arcs (i,j) for $j \in D_i^{(v,\mu)}$ where

$$D_i^{(v,\mu)} = \{j \in Q \mid f_{ij}^{(v,\mu)} = \max_k f_{ik}^{(v,\mu)}\}$$

is the set of workers that maximize the potential benefit of firm i. A directed path $\mathcal{P}$ in $G^{(v,\mu)}$ that connects a doubly mapped worker $j_1 \in Q_{2\mu}$ with another worker j_s is called *(μ-)alternating* resp. *(μ-)augmenting* if j_s is not mapped.

3 An Algorithm to Find a Stable Outcome

Eriksson and Karlander [5] assume integer data and in one step increase a free payoff by at most one. We modify this approach in such a way that we increase the payoff by the smallest possible amount that changes the augmentation digraph. Our strategy to make the map $\mu : P \to Q$ bijective is as follows: As in the classical "men propose – women dispose" algorithm from Gale and Shapley [9] workers with more than one rigid proposal choose the best one and dispose the rest. This way some firms become temporarily unmapped. Each of these unmapped firms has to place another proposal until every worker has at most one rigid proposal. Next, we search the graph $G^{(v,\mu)}$ for alternating paths that reach a worker in $Q_U \cup Q_R$ and alternate the map μ along the path. If none of the above is possible, we increase the payoffs v of workers which are reachable by an alternating path until $G^{(v,\mu)}$ receives a new edge and the process is repeated until the map becomes injective.

The algorithm uses several sub-procedures:

Propose(i): Places a proposal from i to a worker in $D_i^{(v,\mu)}$, i. e. chooses $\mu(i) \in D_i^{(v,\mu)}$.

Dispose(j, i^*): Disposes all firms $i \neq i^*$ that made a rigid proposal to j, i. e. sets $\mu(i)$ to be undefined for all $i \in \mu^{-1}(j) \setminus \{i^*\}$.

Alternate($\mathcal{P}$): μ is alternated along the alternating path $\mathcal{P}$, i. e. all arcs are reoriented and μ is modified such that it uses the new backward arcs. If $\mathcal{P}$ is augmenting then the size of the image of μ increases by 1.

BFS($G, Q_{2\mu}$): Returns all vertices reachable from $Q_{2\mu}$ in G.

PlaceRigidProposals: This procedure is the "men propose – women dispose" algorithm of Gale and Shapley [9]. Here, we denote by P_U the set of temporarily unmapped firms. See Algorithm 2.

HungarianUpdate: Increases the payoffs of all workers reachable from a doubly mapped worker. See Algorithm 3 for details.

Algorithm 1 An Algorithm to Find a Stable Outcome

```
v ← 0
PlaceRigidProposals
while Q_2μ ≠ ∅
  while ∃ μ-alternating path to j ∈ (Q \ μ(P)) ∪ Q_R do
    Alternate(𝒫)
    PlaceRigidProposals
  end while
  HungarianUpdate
end while
```

Theorem 1. *Algorithm 1 eventually finishes with a stable outcome and can be implemented to run in $\mathcal{O}(n^4)$ time.*

Proof. In any iteration of the inner loop of line 4 in Algorithm 1 $|\mu(P)|$ is increased or a rigid proposal is disposed. If there is a path to $Q \setminus \mu(P)$ then $|\mu(P)|$ increases. If the path ends in $j \in Q_R$ then PlaceRigidProposals is called and disposes at least one rigid edge. Note, that a rigid edge once disposed will never be proposed again. If no path exists at all then v is increased by HungarianUpdate until this is the case and in each call of HungarianUpdate at least one new arc shows up in $G^{(v,\mu)}$. Thus, the procedure is finite.

Algorithm 2 PLACERIGIDPROPOSALS

while $P_U \neq \emptyset$ **do**
  for all $i \in P_U$ **do**
    PROPOSE(i)
  end for
  for all $j \in Q_{2R}$ **do**
    Let i^* be the favorite proposal in $\mu^{-1}(j)$
    DISPOSE(j, i^*)
    $v_j \leftarrow b_{i^*j}$
  end for
end while

Algorithm 3 HUNGARIANUPDATE

$\bar{P}\dot{\cup}\bar{Q} \leftarrow$BFS$(G^{(v,\mu)}, Q_{2\mu})$
$u_i \leftarrow \max_j f_{ij}^{(v,\mu)}$
$\Delta \leftarrow \min\{u_i - f_{ik}^{(v,\mu)} \mid i \in \bar{P},\, k \in Q \setminus \bar{Q}\}$
for all $j \in \bar{Q}$ **do**
  $v_j \leftarrow v_j + \Delta$
end for

The while-loop in line 4 of Algorithm 1 might be iterated more than once without finding a path as desired. Anyway, HUNGARIANUPDATE can be implemented so that its consecutive calls until a path is found need $\mathcal{O}(n^2)$ time in sum including an update of the augmentation graph by reusing the BFS-structure from the previous call and storing a minimum distance Δ_j from unmapped vertices and vertices in Q_R to the current BFS forest (see e. g. Galil [10] or Hochstättler et al. [12] for details). Hence, after $\mathcal{O}(n^2)$ time steps we can augment μ or dispose a rigid edge which can happen at most $\mathcal{O}(n^2)$ times. Hence, without considering the complexity of PLACERIGIDPROPOSALS the algorithm runs in $\mathcal{O}(n^4)$.

We also can implement PLACERIGIDPROPOSALS at a total cost of $\mathcal{O}(n^4)$ without any effort. For the first call we have to place n proposals taking $\mathcal{O}(n^2)$ time including the time to find a favorite partner for any $i \in P$. Note that the preference lists may change during the course of the algorithm thus, sorting the lists in a preprocessing does not suffice to speed up the procedure. For each discarded rigid edge we have to find a new favorite partner and after each call of PLACERIGIDPROPOSALS in the inner while-loop we may freely use $\mathcal{O}(n^2)$ to update the augmentation graph which happens at most $\mathcal{O}(n^2)$ times taking the number of rigid edges into account. Thus, the overall complexity of PLACERIGIDPROPOSALS is $\mathcal{O}(n^4)$. As the total cost of ALTERNATE is bounded by $\mathcal{O}(n^3)$ we get a total complexity of $\mathcal{O}(n^4)$.

Next we will show that the algorithm produces a stable outcome. In any stage of the algorithm let $\bar{u}_i := \max_j f_{ij}^{(v,\mu)}$. Then $(\bar{u}, v)$ is stable and $(\bar{u}, v; \mu)$ satisfies (iv) and (v) of Definition 1 since $\mu(i) = j$ implies $j \in D_i^{(v,\mu)}$. As v monotonically increases we also have $v \geq 0$. A worker with no proposer always has payoff zero and is therefore of non-negative value to all firms. Hence together with (iv) and (v) this implies $u \geq 0$. When the algorithm terminates μ is bijective and thus, $(\bar{u}, v; \mu)$ is a stable outcome. $\square$

4 Special Cases and Remarks

Cardinality Matching If $R = \emptyset$ and $a_{ij} + b_{ij} \in \{0, 1\}$ for any edge (i, j) the problem reduces to finding a matching of maximum cardinality among edges with productivity 1 (referred to as *1-edges*). The presented algorithm (see Algorithm 4 for the reduced version) does not seem to have appeared in the literature yet and differs from the standard approach which starts with an empty matching M and searches the graph of 1-edges G^1_M for an M-augmenting path. The algorithm presented here starts with a total but not surjective (and therefore not injective) map μ on the set of nodes in P with at least one 1-edge. A μ-alternating path in the graph of 1-edges G^1_μ is a path from a doubly mapped worker to an unmapped worker using only 1-edges (forward) and μ-edges (backward) and is used to modify the map in a similar fashion as the augmentation of matchings is done in more classical algorithms. Here, the size of the image of μ increases. If no such μ-augmenting path exists, then the set of doubly mapped workers $Q_{2\mu}$ together with the set of firms which are mapped to a worker not in $Q_{2\mu}$ form a vertex cover of G^1_μ with the same cardinality as the image of μ resulting in a maximum matching constructed from μ as in Algorithm 4 (e. g. [8]). If a perfect matching exists, we turn a total (not necessarily injective) map into an injective map instead of making a partial injective map (i. e. a matching) total.

Algorithm 4 Cardinality Matching by Increasing the Image of a Map

```
for all i ∈ P do
    μ(i) ← j    ((i, j) is a 1-edge)
end for
while ∃ μ-augmenting path in G^1_μ to j ∈ (Q \ μ(P)) do
    ALTERNATE(𝒫)
end while
for all j ∈ P, μ^{-1}(j) ≠ ∅ do
    M ← M ∪ {(i, j)}    (i ∈ μ^{-1}(j))
end for
```

While the standard approach is essentially due to Ford and Fulkerson [7] the approach presented here reminds of the preflow-push algorithm (see e. g. [1]), as in the first step we send as much flow as possible from nodes in P to nodes in Q. Then, nodes in $Q_{2\mu}$ correspond to *excess nodes*, i. e. nodes that violate Kirchhoff's law. However, the strategy of lifting node potentials in preflow-push in successive steps does not seem to have anything in common with the augmenting path procedure used here.

A naive implementation of Algorithm 4 leads to an $\mathcal{O}(nm)$ algorithm. Note, that the main difference to the classical approach is in the orientation of the arcs in the search graph. While in the standard approach backward arcs are matchings, here we have exactly one backward arc ending in each non-isolated vertex of P. Thus, the ratio of forward to backward arcs decreases and the search tree in average should be shorter. We wonder if this approach might lead to more efficient implementations for cardinality matching.

Weighted Bipartite Matching If $R = \emptyset$ the algorithm reminds of the Hungarian Method. Like the latter our method is a primal-dual algorithm and can be viewed to start with a weighted vertex cover (u, v) if we set $u_i \leftarrow \max_j f_{ij}^{(v,\mu)}$. We then search for alternating paths or update the payoffs if no such path can be found. Up to a different notion of an augmenting path (i. e. a different algorithm for cardinality matching) and a different orientation of the search graph this strategy is identical with that of the Hungarian Method (see e. g. Frank [8] for a transparent presentation).

Stable Marriage When $F = \emptyset$ our model coincides with the Stable Marriage Model, since the a_{ij} at firm i resp. b_{ij} at worker j may as well be replaced by preference lists. The algorithm is identical to the classical "men propose – women dispose" algorithm of Gale and Shapley [9], that proceeds in rounds.

Comparison with the Algorithm in [12] The algorithm in [12] to find a stable outcome differs from the algorithm presented here in various ways. In [12] (especially rigid) proposals are made asynchronously and not in rounds as in the present implementation. Furthermore, this algorithm is a direct extension of the Hungarian Method as introduced in Kuhn [15, Variant 2], while the algorithm presented here is a direct extension of the original "men propose – women dispose" algorithm of Gale and Shapley [9]. Also the concepts of augmenting paths differ as described above.

Bibliography

[1] Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network Flows*. Prentice Hall, 1993.

[2] Gabrielle Demange, David Gale, and Marilda Sotomayor. Multi-item auctions. *Journal of Political Economy*, 94(4):863–872, 1986.

[3] Xiaotie Deng and Christos H. Papadimitriou. On the complexity of cooparative game solution concepts. *Mathematics of Operations Research*, 19: 257–266, 1994.

[4] Xiaotie Deng, Toshihide Ibaraki, and Hiroshi Nagamochi. Combinatorial optimization games. In *Proceedings of the 8th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 720–729, New Orleans, LA, 1997.

[5] Kimmo Eriksson and Johan Karlander. Stable matching in a common generalization of the marriage and assignment models. *Discrete Mathematics*, 217(1-3):135–156, 2000.

[6] Ulrich Faigle, Sandor P. Fekete, Winfried Hochstättler, and Walter Kern. On the complexity of testing membership in the core of min cost spanning tree games. *International Journal of Game Theory*, 26:361–366, 1997.

[7] Lester R. Ford and Delbert R. Fulkerson. A simple algorithm for finding maximal network flows and an application to the hitchcock problem. *Canadian Journal of Mathematics*, 9:210–218, 1957.

[8] András Frank. On Kuhn's Hungarian method – A tribute from Hungary. Technical report, Egerváry Research Group on Combinatorial Optimization, October 2004.

[9] David Gale and Lloyd S. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69:9–15, 1962.

[10] Zvi Galil. Efficient algorithms for finding maximum matchings in graphs. *ACM Computing Surveys*, 18(1):23–38, 1986.

[11] Dan Gusfield and Robert W. Irving. *The stable marriage problem: Structure and algorithms*. MIT Press, Cambridge, MA, USA, 1989.

[12] Winfried Hochstättler, Hui Jin, and Robert Nickel. The hungarian method in a mixed matching market. Technical report, FernUniversität in Hagen, Germany, October 2005.

[13] Donald E. Knuth. Stable marriage and its relation to other combinatorial problems. In *CRM Proceedings and Lecture Notes*, volume 10. American Mathematical Society, 1997.

[14] Harold W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quaterly*, 2:83–97, 1955.

[15] Harold W. Kuhn. Variants of the Hungarian method for the assignment problem. *Naval Research Logistics Quaterly*, 3:253–258, 1956.

[16] Alvin E. Roth and Marilda Sotomayor. Stable outcomes in discrete and continuous models of two-sided matching: A unified treatment. *Revista de Econometria, The Brazilian Review of Econometrics*, 16(2), November 1996.

[17] Lloyd S. Shapley and Martin Shubik. The assignment game I: The core. *International Journal of Game Theory*, 1:111–130, 1972.

[18] Marilda Sotomayor. Existence of stable outcomes and the lattice property for a unified matching market. *Mathematical Social Sciences*, 39:119–132, 2000.