Dynamic Pareto Optimal Matching

Rudolf Fleischer Yihui Wang

Department of Computer Science
Shanghai Key Laboratory of Intelligent Information Processing
Fudan University, Shanghai, China

E-mail: {rudolf,yihuiwang}@fudan.edu.cn

Abstract

We consider the problem of assigning houses to agents, where each agent has his own partial ranking of the houses (i.e., a preference list of a subset of the houses). A matching M is Pareto optimal if there exists no other matching M' such that all agents have a house in M' not worse than in M and at least one agent has a better house in M'. In a dynamic market where agents and houses can be added or removed at any time, a Pareto optimal matching may lose its optimality. In this paper, we give an $O(m)$ time algorithm to maintain a maximum cardinality and Pareto optimal matching, where m is the total size of all preference lists. Furthermore, we show that a Pareto optimal matching can be transformed to any other Pareto optimal matching through a certain constructed graph in linear time.

1 Introduction

In the *house allocating problem*, we have a set of *agents* and a set of *houses*. Each agent has a preference list of a subset of the houses he considers suitable. The objective is to assign the houses to the agents and maintain Pareto optimality in a dynamic market where agents and houses may be added or removed at any time. Recall that a matching M is *Pareto optimal* if there exists no other matching M' such that all agents have a house in M' not worse than in M and at least one agent has a better house in M'. We note that Pareto optimal matchings are not only of interest for the house allocating problem [1, 13], but also when we assign primary students to high schools, graduate students to training positions, etc. [10].

Abraham et al. [3] gave an $O(\sqrt{n}m)$ time algorithm to find a Pareto optimal matching of maximum cardinality, where n is the number of agents and m is the total length of all preference lists. In a real housing market, after we found a Pareto optimal matching, new agents and new houses may appear, or some agents may stop doing business and some houses may be withdrawn from the market. In this case, the original matching may not be Pareto optimal anymore. We could use the algorithm in [3] to find a new Pareto optimal matching for the new market situation, but it would take a long time and involve a big change of the matching. Instead, we propose an $O(m)$ time algorithm to update the old matching in case of changes.

A related one-sided matching problem is the problem of finding a matching that most people prefer. Such a matching is called a *popular matching* [4, 12]. In a dynamic market Abraham and Kavitha [5] presented an algorithm to convert any given matching to some popular matching via a 2-step voting path. We present a similar result for Pareto optimal matchings. Any two Pareto optimal matchings can be transformed into each other by switching edges along some disjoint cycles in a specially constructed graph.

Another related criterion is exchange-stability, first introduced by Alcalde [6] (who called it ξ-

stability). Given a matching, agents can exchange their houses to improve their utility. A matching is called ξ-stable or *coalition-exchange-stable* if no such exchanges are possible by any set of agents. In the special case when there exists no two agents who could improve their situation by exchanging their houses, the resulting matching is called *exchange-stable* [7]. Cechlarova and Manlove [8] showed that the exchange-stable matching problem where both sides must maintain stability is NP-hard. Irving [11] showed that it is NP-complete to decide whether a problem instance admits a stable matching with the additional exchange stability property. This is quite different from the classical stable matching and one-sided Pareto optimal (exchange-stable) matching problem which are all linear time solvable (a Pareto optimal matching that is not necessarily of maximum cardinality can be found using a greedy algorithm in $O(m)$ time [1]).

In Section 2, we will give some formal definitions of the Pareto optimal matching problem. In Section 3 we will propose our dynamic update algorithm. In section 4, we will show how switch between any two Pareto optimal matchings in linear time. We conclude in Section 5 with some remarks and an open problem how to generate efficiently all Pareto optimal matchings.

2 Preliminaries

An instance of the Pareto optimal matching problem consists of a bipartite graph $G = (A \cup H, E)$, where $A = a_1, a_2, \ldots, a_r$ is the set of agents and $H = h_1, h_2, \ldots, h_s$ the set of houses. An edge $(a, h) \in E$ denotes that the house h is acceptable to agent a. Let $n = r+s$ and $m = |E|$. A matching M of G is a subset of edges such that no two edges share a common vertex. We denote the cardinality of a matching by $|M|$. For each agent a, there is a preference list ranking all the houses acceptable to a. The rank of an edge (a, h) is the rank (or priority) that agent a has assigned to house h.

We say an agent a *prefers* matching M_2 to matching M_1 if

(a) agent a is matched to some house in matching M_2, but not in M_1, or

(b) agent a is matched in both matchings but likes the house in M_2 better than the house in M_1.

If an agent is unmatched in both matchings, he is *indifferent* between the two matchings.

Definition 1. *A matching M is* Pareto optimal *if there does not exist a matching M' such that*

(a) there is at least one agent who prefers the house he got in matching M' to the house he got in M, and

(b) no agent prefers the house he got in M to the house he got in M'.

□

3 Dynamic Pareto Optimal Matching

In this section, we first review the $O(\sqrt{n}m)$ algorithm [3] to compute a maximum cardinality Pareto optimal matching in the static case. Then we give our linear algorithm to update the maximum cardinality Pareto optimal matching after some changes in the market.

3.1 Maximum Cardinality Pareto Optimal Matching

If we just want to find some Pareto optimal matching, we can use the $O(m)$ time greedy algorithm that assigns each agent one of the remaining houses of highest rank. However, there will usually be many Pareto optimal matchings of various sizes for a given instance. Consider for example the following three preference lists for three agents: $a_1 : h_1, h_2, h_3$ (h_1 is the most favorite and h_3 is the least favorite house); $a_2 : h_1, h_2$; and $a_3 : h_1, h_2$. Fig. 1 shows two Pareto optimal matchings of different size.

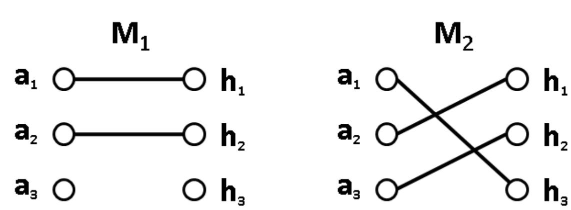

Figure 1. Pareto optimal matchings of different sizes.

The greedy algorithm cannot guarantee to find a maximum cardinality Pareto optimal matching. Abraham et al. [3] gave an $O(\sqrt{n}m)$ algorithm to find a maximum cardinality Pareto optimal matching, based on Gales's Top Trading Cycles Method, which basically executes the following three steps:

1. Use the Hopcroft-Karp algorithm [9] to find a maximum cardinality matching M in G.

2. *Trade-in-free test:* test whether there exists an agent that prefers an unmatched house to the one he got in M. If it exists, we change the matching.

3. *Coalition-free test:* test whether there exists some sequence of matched agents $(a_0, a_1, \ldots, a_{k-1})$ such that each a_i prefers the house matched by a_{i+1} to their own house (the subscripts here are all modulo k). If there is such a sequence, match each agent a_i to the house that had been matched to a_{i+1}.

The Hopcroft-Karp algorithm takes $O(\sqrt{n}m)$ time to find the maximum matching in a bipartite graph. Obviously the trade-in-free and coalition-free tests only take $O(m)$ time. Thus the total time for this algorithm is $O(\sqrt{n}m)$. The correctness is based on the following observation by Abraham et al. [3].

Theorem 2. *A matching M is Pareto optimal if and only if M is maximal, trade-in-free, and coalition-free.* □

3.2 Dynamic Pareto Optimal Matching

Consider a market where agents and houses can leave or enter the market at any time. In this section, we give an $O(m)$ time algorithm to update a maximum cardinality Pareto optimal matching.

Theorem 3. *We can maintain a maximum cardinality Pareto optimal matching in $O(m)$ time per update.*

Proof. We must consider four cases: an agent or house may enter the market, and an agent or house may leave the market. The two cases where an agent or house is removed can easily be transformed into a case of an agent or house being added to the market.

Case 1. *A new agent enters the market.*

If a new agent a enters the market, we first check whether there exists a free house acceptable to a. If yes, we add the edge (a, h) to M, where h is the highest ranked free house acceptable to a. Obviously, the resulting matching still satisfies the three conditions of Theorem 2 and is thus Pareto optimal. It is also a maximum matching because the new agent got matched.

If there is no free house acceptable to a, we try to find an augmenting path from a. An *augmenting path* is a path $P = v_1, v_2, ..., v_k$ with two unmatched end vertices v_1 and v_k such that all the edges on the path satisfy $(v_i, v_{i+1}) \in M$ if and only if $(v_{i+1}, v_{i+2}) \notin M$ for $1 \leq i \leq k-2$. Starting at a, we use depth first search to find a path whose every second edge edge appears in M until we reach an unmatched house h'. Then we add to M all the edges on the path that originally did not belong to M, and we delete from M the edges that originally belonged to M. See Fig. 2 for an example.

Now we only need to test the coalition-free condition for the resulting new matching M' (and maybe update the matching if a coalition was found) to make it Pareto optimal according to Theorem 2. Since agent a and house h' had been unmatched in M and are matched in M', we have $|M'| = |M| + 1$, i.e., the new matching is still a maximum cardinality matching.

If no augmenting path starting at a exists, the original matching M is still a Pareto optimal matching of maximum cardinality. The time complexity for computing an augmenting path and testing the coalition-free condition is $O(m)$, thus the total time is $O(m)$.

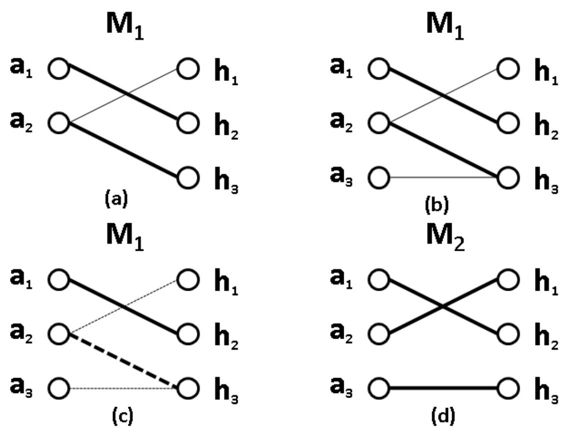

Figure 2. Augmenting path for the new agent.

Case 2. *A new house enters the market.*

We assume that in this case all agents who would like to accept the new house will add it to their preference list.

If a new house h enters the market, we first check whether there is some unmatched agent who would like to accept h. If there are several such agents, we choose an agent which gives the house the highest rank. Now we need to test the coalition-free condition (and update the matching if a coalition has been found).

If no unmatched agent would like to accept h, we try to find an augmenting path starting from h and ending in an unmatched agent vertex. Changing the edges on the path as above gives a new matching with one more edge (i.e., it is a maximum cardinality matching), but again we must test the coalition-free condition. The time complexity is in this case again $O(m)$.

Case 3. *An agent leaves the market.*

If the leaving agent is unmatched, the original matching M is still an optimal one. If a matched agent a leaves the market, his house $M(a)$ becomes unmatched. Since the matching M without a is still Pareto optimal and of maximum cardinality, we may consider this a case of a new house $M(a)$ entering the market.

Case 4. *A house leaves the market.*

If the house is unmatched, the original matching M is still an optimal one. If the house is matched, we may consider it as a case of an agent entering the market.

□

4 Switching Between Different Pareto Optimal Matchings

We have seen earlier that there may be many Pareto optimal matchings for a given problem instance. In this section we will use the method in [2] to construct a directed graph to convert one Pareto optimal matching to another in linear time. Furthermore, we will show that any two Pareto optimal matchings can be transformed into each other using only one step in linear time through certain directed graph.

Note that for two arbitrary matchings M_1 and M_2 there exists at least one agent prefering M_1 to M_2, and at least one agent prefering M_2 to M_1. Otherwise the matching would violate the coalition-free condition.

Given a Pareto optimal matching M, we construct a complete directed graph $G' = (A, E')$, where A is the set of all agents. If agent a_i is matched and prefers his matched house $M(a_i)$ to the house $M(a_j)$, the directed edge (a_i, a_j) has cost 0. If agent a_i prefers $M(a_j)$ to his own house $M(a_i)$, or if a_i is unmatched but can accept the house $M(a_j)$, the directed edge (a_i, a_j) has cost -1.

Now we try to find a negative cycle $C = (a_1, a_2, \ldots, a_l)$ in G', i.e., a cycle of total length smaller than 0. We construct a new matching $M' = M - \{(a_1, M(a_1)), (a_2, M(a_2)), \ldots, (a_l, M(a_l))\} + \{(a_1, M(a_2)), (a_2, M(a_3)), \ldots, (a_l, M(a_1))\}$. Since C is a negative cycle, at least one agent prefers the new matching M' to M. Also, there is at least one edge in the cycle of cost 0, otherwise M is not coalition-free and thus not Pareto optimal. Then at least one agent prefers the original matching M to M'. However, the new matching M' may not satisfy the trade-in-free and coalition-free condition anymore, so we still need to do some trade-in-free and coalition-free tests.

Now that we know how to use the directed graph G' to switch between Pareto optimal matchings, we will show that any two matchings can be converted with a few disjoint negative cycles and without the need of any other tests.

Theorem 4. *Any Pareto optimal matching can be converted to any other Pareto optimal matching by augmenting a disjoint set of negative cycles in G'.* □

Proof. Consider two Pareto optimal matchings M_1 and M_2, $M_1 \neq M_2$. Then there must exist an agent subset $A_1 = a_{11}, a_{12}, \ldots, a_{1p}$ such that all the agents in this subset prefer matching M_1 to M_2, and there must exist an agent subset $A_2 = a_{21}, a_{22}, \ldots, a_{2q}$ such that all the agents in this subset prefer matching M_2 to M_1.

We start from agent a_{11}. We assume $M_2(a_{11})$ has been matched to a_i in M_1. Then edge (a_{11}, a_i) has cost 0. Suppose a_i has been matched to house h in M_2, then there are two cases.

1. a_i prefers M_1 to M_2, which means $a_i \in A_1$;

2. a_i prefers M_2 to M_1, which means $a_i \in A_2$.

We continue to find the agent matched to $M_2(a_i)$ in M_2, let us assume it was a_j. Then the edge (a_i, a_j) has either cost 0 or -1, depending on the two cases above. We continue to construct a sequence of edges. Each time we reach a new agent vertex it must belong to A_1 or A_2. Since A_1 and A_2 are finite sets, the sequence of edges must close a cycle. If there are some agents in A_1 or A_2 left that do not appear in the cycle, we arbitrarily choose one agent to start another cycle. In the end, all the agents in A_1 and A_2 will form several disjoint cycles. Each of these cycles must be negative cycles. If there is some cycle with cost 0, the matching converted from it will have a cycle with all edges of cost -1 violating the coalition-free condition. Thus all these negative cycles belong to the graph G', which means we can use graph G' to convert matching M_1 to M_2 or conversely.

The time complexity to adjust the negative cycle is $O(n)$. The trade-in-free and coalition-free conditions can be tested in $O(m)$ time. Therefore, the total time needed is $O(m)$. □

Now we know that any two Pareto optimal matchings can be converted into each other using one or several negative cycles in G'. If we arbitrarily choose some negative cycles in G', it may result in the new graph violating the trade-in-free and coalition-free condition. The proof above shows that any two Pareto optimal matchings can be converted into each other without testinbg the trade-in-free or coalition-free condition. This might be used to greedily generate all the Pareto optimal matchings.

5 Conclusion and Future Work

We gave an $O(m)$ time algorithm to maintain the maximum cardinality Pareto optimality in the dynamic situation where some vertex of the bipartite matching graph may enters or leave arbitrarily. We also showed that any Pareto optimal matching can be converted to another one by ffinding a certain set of disjoint negative cycles in some directed graph G'.

In Section 1 we mentioned that many problems about Pareto optimality (or exchange-stability) in the context of the stable marriage problem are NP-hard. Thus, to generate all Pareto optimal matchings it might be helpful to study both the Pareto optimal problems and related stable marriage problems. In Section 4 we have seen that some negative cycles in G' will result in a violation of the trade-in-free or coalition-free conditions which would increase the time needed for a conversion. The time for the exchange of houses of the vertices on the negative cycles is just $O(n)$, but trade-in-free and coalition-free tests need $O(m)$ time. When we try to generate all the Pareto optimal matchings, these non-trade-in-free or non-coalition-free negative cycles may also result in repetitions in the matching enumeration. We leave it as an open problem to efficiently generate all Pareto optimal matchings.

Acknowledgements

This work was supported by a grant from the National Natural Science Foundation of China (no. 60573025), and the Shanghai Leading Academic Discipline Project (project number B114).

We would like Prof Zhu Hong for his help.

According to international convention, the authors of this paper are ordered alphabetically by their family name. The 'first author' order would be Y. Wang and R. Fleischer.

A preliminary version of these results was presented in Chinese at the 14th General Meeting of the Special Interest Group on Fuzzy Systems and Mathematics of the Systems Engineering Society of China, 2008.

References

[1] A. Abdulkadiroglu and T. Sonmex. Random serial dictatorship and the core from random endowments in house allocation problems. *Econometrica*, 66(3):689–701, 1998.

[2] D.J. Abraham. Algorithmics of Two-sided Matching Problems, Master Thesis, University of Glasgow, 2003.

[3] D.J. Abraham, K. Cechlarova, D.F. Manlove, and K. Mehlhorn. Pareto optimality in house allocation problems. *In Proc. of 15th ISAAC*, 2004, pages 3–15.

[4] D.J. Abraham, R.W. Irving, T. Kavitha, and K. Mehlhorn. Popular matchings. *In Proc. of 16th SODA*, 2005, pages 424–432.

[5] D.J. Abraham and T. Kavitha. Dynamic matching markets and voting paths. *SWAT 2006*, LNCS 4095: 65–76.

[6] Alcalde. Exchange-proofness or divorce-proofness? Stability in one-sided matching markets. *Econom. Design*, 1:275–287, 1995.

[7] K. Cechlarova. On the complexity of exchange-stable roommates. *Discrete Appl. Math.*, 116(3):279–287, 2002.

[8] K. Cechlarova and D.F. Manlove. The exchange-stable marriage problem. *Discrete Applied Mathematics*, 152(1-3):109–122, 2005.

[9] J.E. Hopcroft and R.M. Karp. A $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing*, 2:225–231, 1973.

[10] A. Hylland and R. Zeckhauser. The efficient allocation of individuals to positions. *Journal of Political Economy*, 87(2):293–314, 1979.

[11] R.W. Irving. The man-exchange stable marriage problem. *Technical Report*, University of Glasgow, Department of Computing Science, TR-2004–177.

[12] J. Mestre. Weighted popular matchings. *In Proc. of 33rd ICALP*, 2006, LNCS 4051: 715–726.

[13] A.E. Roth and A. Postlewaite. Weak versus strong domination in a market with indivisible goods. *Journal of Mathematical Economics*, 4:131–137, 1997.