Operations Research Letters 47 (2019) 344–347

Contents lists available at ScienceDirect

Operations Research Letters

journal homepage: www.elsevier.com/locate/orl

Algorithms for Pareto optimal exchange with bounded exchange cycles

Haris Aziz

UNSW Sydney and Data61 CSIRO, Australia

ARTICLE INFO

Article history:
Received 21 February 2019
Received in revised form 10 June 2019
Accepted 10 June 2019
Available online 18 June 2019

Keywords:
Kidney exchange
Housing markets
Strategyproofness
Pareto optimality
Individual rationality

ABSTRACT

We consider exchange markets with single-unit endowments and demands where there is a bound on the size of the exchange cycles. The computational problem we study is that of computing a Pareto optimal and individually rational allocation. We present polynomial-time algorithms to compute a Pareto optimal and individually rational allocation when preferences are strict, the exchange bound is two, or when Pareto optimality is replaced with weak Pareto optimality.

1. Introduction

We consider the basic Shapley–Scarf housing market in which each agent owns one item and the goal is to exchange the items in a mutually beneficial manner. Typical properties that are desirable in this context are Pareto optimality and individual rationality [2,4,9,11,14].

An allocation as a result of exchanges of items can alternatively be seen as a set of disjoint cycles consisting of agents and items. Each agent is seen as giving her item to the next agent in the 'exchange' cycle. We consider allocations that are a result of disjoint exchange cycles such that there are at most L agents in each cycle. We will call such allocations L-allocations. They are referred to as allocations that are a result of most L-way exchanges. Considering short exchange cycles has several motivating factors. The most prominent motivation of short cycles is in the context of kidney exchange in which exchange cycles have a natural size limit due to operational and logistical factors [1]. For example, there is a bound on how many operations can be conducted simultaneously. Even for other barter markets, we may not want to depend on too many people coming together for an exchange hence we may want to resort to short exchange cycles.

We especially focus on a concept called L-efficiency (see e.g., [10]). An allocation is L-efficient if it is an L-allocation and it is not Pareto dominated by another L-allocation. An allocation is weakly L-efficient if it is an L-allocation and it is not strongly Pareto dominated by another L-allocation. In this note, we focus on achieving individual rationality and L-efficiency. The existence of allocations satisfying both properties is easy to see: with the set of L-allocations, we start from the endowment allocation and make a finite number of Pareto improvements until no further improvement can be made. We focus on *computing* such an allocation.

We present a polynomial-time algorithm to find an individually rational and L-efficient allocation for strict preferences. If we allow for indifferences, we show that the problem of finding an L-efficient allocation is NP-hard. However, our algorithm can be used to find an individually rational and weak L-efficient allocation for weak preferences. We also show that for $L = 2$, the problem of computing an individually rational and L-efficient allocation is polynomial-time solvable. The complexity classification is summarized in Table 1.

2. Related work

The focus on L-allocations is well-established in the literature (see e.g., [1,5,10]). In contrast to requiring Pareto optimality, the main focus in the algorithmic literature is (1) on either maximizing the number of satisfied agents assuming that preference are dichotomous [1] or (2) examining the complexity of checking whether a stable allocation exists or not (see e.g., [5]). Both problems are NP-hard. In recent years, integer programming techniques have been used for these problems (see e.g., [1,7])

When there is no restriction on the size of the exchange cycles, then variants of the well-known Gale's Top Trading Cycles algorithm satisfy polynomial-time computability, strategyproofness, individual rationality, and Pareto optimality (see e.g., [8,11,13]).

The problem of finding a Pareto optimal allocation is NP-hard when there are multiple items to be allocated to each agent (see e.g., [3]). It becomes easier when we consider Pareto optimality with respect to the responsive set extension (see e.g., [6]).

E-mail address: haris.aziz@unsw.edu.au.

https://doi.org/10.1016/j.orl.2019.06.002
0167-6377/

Table 1
Complexity of finding an IR and L-efficient allocation.

Restrictions	Cycle bound			
	2	3	n	General L
Strict	In P (Proposition 2)	In P (Proposition 2)	In P (TTC)	In P (Proposition 2)
Dichotomous	In P [12]	NP-c (Corollary 1)	In P (GATTC)	NP-c (Corollary 1)
Weak	In P (Proposition 1)	NP-c (Corollary 2)	In P (GATTC)	NP-c (Corollary 2)

3. Model and concepts

Let $N = \{1, \ldots, n\}$ be a set of n agents and $H = \{h_1, \ldots, h_n\}$ a set of n items. The endowment function $e : N \to H$ assigns to each agent the item he originally owns.

Each agent has complete and transitive preferences $\succsim_i$ over the items and $\succsim = (\succsim_1, \ldots \succsim_n)$ is the preference profile of the agents. The *housing market* is a quadruple $M = (N, H, e, \succsim)$. For $S \subseteq N$, we denote $\bigcup_{i\in S}\{e(i)\}$ by $e(S)$. A function $x : S \to H$ is an *allocation* on $S \subseteq N$ that allocates item $x(i)$ to agent $i \in N$. We will generally consider weak orders. Two restrictions of weak orders are strict preferences and dichotomous preferences. Under strict preferences, there are no indifferences: if $h \succsim_i h'$ then $h' \not\succsim_i h$. Under dichotomous preferences, an agent partitions the items into at most two indifference classes.

An allocation x is *individually rational (IR)* if $x(i) \succsim_i e(i)$ for all $i \in N$. An allocation x is *Pareto optimal* if there exists no other allocation y such that $y(i) \succsim_i x(i)$ for all $i \in N$ and $y(i) \succ_i x(i)$ for some $i \in N$. An L-allocation x is *L-efficient* if there exists no other L-allocation y such that $y(i) \succsim_i x(i)$ for all $i \in N$ and $y(i) \succ_i x(i)$ for some $i \in N$. An L-allocation x is *weakly L-efficient* if there exists no other L-allocation y such that $y(i) \succ_i x(i)$ for all $i \in N$.

For any housing market, $M = (N, H, \succsim, e)$, there exists a *corresponding exchange graph* $G_M = (V, E)$ where $G_M = (V, E)$ where $V = N \cup H$ and $E = \{(h, i) : h = e(i)\} \cup \{(i, h) : h \succsim_i e(i)\}$. We note that any individually rational allocation for $M = (N, H, \succsim, e)$ has a one-to-one correspondence with a set of vertex-disjoint cycles in which each agent gets the item she points to in the cycle. We note that any individually rational L-allocation for $M = (N, H, \succsim, e)$ has a one-to-one correspondence with a set of vertex-disjoint cycles of size at most 2L in which each agent gets the item she points to in the cycle. We will denote $|E|$ by m.

4. Computing an individually rational and L-efficient allocation

We first note that if preferences are dichotomous, finding an L-efficient allocation is NP-hard even if we do not require individual rationality. The argument heavily relies on the fact that checking whether there exists an L-allocation in which each agent gets an acceptable item is NP-complete [1]. Dichotomous preferences are standard in the framework of kidney exchange. Recently there has been work on exact algorithms to save the maximum number of lives [15]. We give the argument for the sake of completeness.

Corollary 1. *Under dichotomous preferences, computing an L-efficient allocation is NP-hard for $L = 3$.*

Proof. It is known that if $L = 3$, agents have dichotomous preferences, checking whether there exists an L-allocation in which each agent gets maximally preferred item is NP-complete [1]. If there exists an allocation in which each agent gets maximally preferred allocation, then any Pareto optimal allocation will be such an allocation. If there exists a polynomial-time algorithm to find a Pareto optimal allocation, it will find such an allocation if it exists. Therefore, computing an L-efficient allocation is NP-hard for $L = 3$. □

Corollary 2. *Under weak preferences, computing a Pareto optimal allocation is NP-hard.*

If $L = n$, then L-efficiency coincides with Pareto optimality. Under strict preferences, the famous Top Trading Cycles (TTC) algorithm returns an individually rational and Pareto optimal allocation. Even when ties are allowed, it is well-known that extensions of the TTC algorithm find an IR and PO allocation (see e.g., [8,11,13]). All of the algorithms in the cited papers fall in the class of algorithms called GATTC [4].

Next, we point out that even for $L = 2$, there exists a polynomial-time algorithm that returns an individually rational and L-efficient allocation. Previously, for $L = 2$ and dichotomous preferences, a polynomial-time was presented that returns an L-efficient and individually rational allocation [12]. The following algorithm for pairwise exchange works for any number of items agents may have as long as they are only interested in successful pairwise exchanges.

Proposition 1. *For weak preferences and $L = 2$, there exists a polynomial-time algorithm that returns an individually rational and L-efficient allocation in time $O(n^3)$.*

Proof. Construct a weighted graph $G = (V, E, w)$ where $\{i, j\} \in E$ if $e(i) \succsim_j e(j)$ and $e(j) \succsim_i e(i)$. We set the weights of edges as follows $w(e) = r_i^j + r_j^i$ where r_i^j is the number of rank improvements i gets when she $e(j)$ in place of $e(i)$. We then compute a maximum weight matching M. The matching corresponds to the pairwise exchanges between the agents and hence to an allocation of items to the agents. If some $i \in N$ is unmatched, she is not part of any pairwise exchange. If $\{i, j\} \in M$ then, i and j exchange their items. The allocation is IR because an agent only exchanges with an agent whose item she finds individually rational.

We prove that the allocation x_M corresponding to matching M is L-efficient. Note that the weight of M is equal to the total number of rank improvements over all agents. Suppose for contradiction that x_M is not L-efficient. Then there exists another matching M' in which each agent gets at least the same rank item and one agents gets a more preferred item. This implies that if i is unmatched in M', then it is also unmatched in M. If i is matched in $e \in M$, she contributes as much to some edge in M as she does to $e \in M$. Since $x_{M'}$ Pareto dominates x_M, at least one agent contributes more weight to M' than she does to M. This implies that M' has higher total weight than M. But this is a contradiction. □

Next, we present a polynomial-time algorithm that returns an individually rational and L-efficient allocation. The result contrasts sharply with the fact that finding L-allocations that are stable is NP-hard even under strict preferences [5]. The high-level idea of the algorithm is as follows. We take an agent i_1 owning g_1 and try to refine the set of L-allocations in which the agents get a most preferred item g_2. We consider the agent i_2 who owns g_2. We then refine the set of allocations further by considering only those L-allocations in which i_2 gets her most preferred item. Instead of enumerating exponentially many allocations under consideration, we implicitly track them by only focusing on those edges of the exchange cycles that we have fixed. We continue until we reach some i_j who must get g_1. We remove this cycle and continue until all the items have been allocated. The algorithm is formally specified as Algorithm 1.

Algorithm 1 Priority Cycles Algorithm (PCA) to find an L-efficient and individually rational allocation

Input: $(N, H, \succsim, e)$ where $\succsim$ is a strict preference profile, and permutation π on N.

$N' \longleftarrow N$; $H' \longleftarrow H$.
Construct a graph $G = (V, E)$ where $V = N \cup H$ and $E = \{(h, i) : h = e(i)\} \cup \{(i, h) : h \succsim_i e(i)\}$. % Takes time $O(n + m)$
while G is not empty **do**
  Consider any agent i_1 with the smallest index in G.
  $C \longleftarrow \{i_1\}$.
  while $i_{|C|}$ has a path to i_1 of length at most $2L - 2|C|+2$ in G % Takes time $O(n + m)$ **do**
    Find the most preferred item h that $i_{|C|}$ points to such that there exists a path of length at most $2L - 2|C|+1$ from h to i_1.
    Set G to one in which $i_{|C|}$ points only to h.
    Consider the owner of h. Call it $i_{|C|+1}$. $C \longleftarrow C \cup \{i_{|C|+1}\}$.
  end while
  Give each agent i the item she points to in C. Remove the agents in C and the items they point to. Remove from G any edges involving the removed the vertices. % Takes time $O(n + m)$
end while

Proposition 2. *For strict preferences and any given integer L, there exists a polynomial-time $O(n(n + m))$ algorithm that returns an individually rational and L-efficient allocation.*

Proof. Consider Algorithm 1. We first claim that the resultant allocation is an L-allocation. We consider the agent i_1 with the lowest index and include her in the exchange cycle. We know that the agent has at least one feasible L exchange cycle: the one in which she gets her own item. Therefore, by choosing i_1, we can guarantee a feasible L-exchange in which i is involved. Consider any exchange cycle C' involving an alternating sequence of agents and items that includes i_1. We claim that the length of the cycle is at most $2L$ which would imply that at most L agents exchange among themselves. When we build the cycle, we also build a corresponding set C of agents in the cycle. Suppose we have the set C currently consists of agents $i_1, \ldots, i_{|C|}$. Then by construction, we know that there exists a path of length exactly $2|C| - 2$ from i_1 to $i_{|C|}$. When we choose the edge for $i_{|C|}$ to an item, we check in the condition of the while loop that the edge allows for a path of length at most $2L - 2|C| + 2$ from $i_{|C|}$ to i_1. This can be checked in time $O(m)$ by using breadth first search on the constructed graph G. Now consider the length of the cycle from i_1 to i_1. It is the sum of the length of the path from i_1 to $i_{|C|}$ (which is exactly $2|C| - 2$) and the length of the path from $i_{|C|}$ to i_1 (which is at most $2L - 2|C| + 2$). Therefore the length of the cycle from i_1 to i_1 is at most $(2|C| - 2) + 2L - 2|C| + 2 = 2L$.

In each iteration of the outer while loop, one exchange cycle is identified and the vertices are removed from the graph. In each iteration of the inner while loop, either another vertex is added to the exchange cycle or one vertex is removed from consideration so it takes time at most $O(n+m)$. Therefore the overall algorithm takes time $O(n(n + m))$.

When we consider exchange cycles, note that an agent i only points to an item o if $o \succsim_i e(i)$. Therefore the allocation is individually rational.

Next, we argue that the allocation is L-efficient. Consider $C_1, \ldots, C_\ell$ the 'C' sets discovered during the running of the PCA. We prove by induction on order of the C sets that no agent is involved in a Pareto improvement. Consider C_1 the first C set. Consider the first agent i_1 added to C_1. Note that i_1 gets the most preferred item she can get in an exchange of size at most L. Therefore i_1 is not part of any Pareto improvement. Given the condition i_1 gets a most preferred item, we give the most preferred item to i_2 and so on. If some agent i_j were to get a more preferred item than her allocated item, it implies that at least some agent among $\{i_1, i_1, \ldots, i_{j-1}\}$ gets a less preferred item. Therefore we have established that no agent in C_1 is part of a Pareto improvement. Therefore we can remove these agents and their allocated items from consideration from any Pareto improvement that results in an L-allocation. We can repeat the same argument for the next C set until all the agents are accounted for. Thus the allocation is L-efficient. □

Proposition 3. *For weak preferences and any given integer L, there exists a polynomial-time algorithm that returns an individually rational and weakly L-efficient allocation.*

Proof. The algorithm works as follows. For input preference profile $\succsim$ we break ties arbitrarily to get an artificial preference profile $\succsim'$ and then run PCA on profile $\succsim'$ to get outcome y. We argue that y is weakly L-efficient under $\succsim'$. Note that for any two allocations z and y, $z(i) \succ_i y(i)$ implies $z(i) \succ_i' y(i)$. Next we argue that y is weakly L-efficient under $\succsim$. Suppose y is not weakly L-efficient. Then there exists another allocation z such that $z(i) \succ_i y(i)$ for all $i \in N$. But this means that $z(i) \succ_i' y(i)$ for all $i \in N$. Therefore y is *not* L-efficient under strict preference profile $\succsim'$ which is a contradiction of the fact that PCA returns an L-efficient allocation if the input preferences are strict. Thus y is weakly L-efficient under $\succsim$. □

We call an algorithm *non-bossy* if no agent can change her preferences and get the same allocation but some other agent gets a different allocation. We observe that PCA is non-bossy.

Proposition 4. *PCA is non-bossy.*

Proof. Compare $x = PCA(\succsim)$ with $x' = PCA(\succsim)$ in which i changes her preferences and gets the same item. Let the cycles discovered while returning x be $C_1, C_2, \ldots, C_k$ where $i \in C_j$. Since the only stage where a different cycle can be involved is when i is involved, we note that for $\ell \in \{1, \ldots, j - 1\}$, the cycles discovered while finding x' are $C_1, C_2, \ldots, C_{j-1}, \ldots$. At this point, the graphs for both x and x' are the same. Note that since i points to x and all agents who find $e(i)$ individually rational point to $e(i)$. Therefore C_j is executed both under $PCA(\succsim)$ and $PCA(\succsim')$. At this point i is no more part of the graph and subsequently $PCA(\succsim)$ and $PCA(\succsim')$ take exactly the same steps and find the same cycles. Hence no agent gets a different item. □

One may wonder whether it is possible to additionally satisfy strategyproofness. Nicoló and Rodríguez-Álvarez [10] proved that for strict preferences, there exists no IR, L-efficient, and strategyproof algorithm even for single-unit allocations and for $L = 2$. As a corollary, PCA is not strategyproof.

Remark 1. A possible approach towards finding an efficient matching is to run a priority algorithm based on a pre-fixed ordering of agents in which we start from the set of all individually rational L-efficient allocations each agent in the ordering refines the set of allocation to her most preferred ones. However, PCA is not a priority algorithm because the ordering of the agents depends on the preferences of the agents. Since the priority algorithm returns an L-efficient allocation even under weak preferences, it cannot run in polynomial time unless P=NP. Under strict preferences and $2 < L < n$, it is not clear whether a priority rule has some algorithmic specification that is polynomial time.

Remark 2. Even for strict preferences and $L = n$, PCA does not coincide with the TTC algorithm. The reason is that in TTC, we do not necessarily find an exchange of the agent with the lowest index.

Acknowledgments

Haris Aziz is supported by a Scientia Fellowship, Australia. He thanks Péter Biró for useful pointers and Zhaohong Sun for helpful discussion.

References

[1] D. Abraham, A. Blum, T. Sandholm, Clearing algorithms for barter exchange markets: Enabling nationwide kidney exchanges, in: Proceedings of the 8th ACM Conference on Electronic Commerce (ACM-EC), ACM Press, 2007, pp. 295–304.

[2] D.J. Abraham, K. Cechlárová, D.F. Manlove, K. Mehlhorn, Pareto optimality in house allocation problems, in: Proceedings of the 16th International Symposium on Algorithms and Computation (ISAAC), in: 3341 of Lecture Notes in Computer Science (LNCS), 2005, pp. 1163–1175.

[3] H. Aziz, P. Biro, J. Lang, J. Lesca, Monnot, J, Optimal reallocation under additive and ordinal preferences, in: Proceedings of the 15th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS), 2016, pp. 402–410.

[4] H. Aziz, B. de Keijzer, Housing markets with indifferences: a tale of two mechanisms, in: Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI), 2012, pp. 1249–1255.

[5] P. Biró, Stable exchange of indivisible goods with restrictions, in: Proceedings of the 5th Japanese-Hungarian Symposium, 2007, pp. 97—105.

[6] P. Biró, F. Klijn, S. Papai, Serial dictatorships in a multi-object shapley-scarf market, 2018.

[7] V. Costa, X. Klimentova, P. Biro, A. Viana, J.P. Pedroso, New integer programming formulations for the stable exchange problem, 2018.

[8] P. Jaramillo, V. Manjunath, The difference indifference makes in strategy-proof allocation of objects, J. Econom. Theory 147 (5) (2012) 1913–1946.

[9] J. Ma, Strategy-proofness and the strict core in a market with indivisibilities, Internat. J. Game Theory 23 (1) (1994) 75–83.

[10] A. Nicoló, C. Rodríguez-Álvarez, Transplant quality and patients' preferences in paired kidney exchange, Games Econom. Behav. 74 (1) (2012) 299–310.

[11] C.G. Plaxton, A simple family of top trading cycles mechanisms for housing markets with indifferences, in: Proceedings of the 24th International Conference on Game Theory, 2013.

[12] A.E. Roth, T. Sönmez, M.U. Ünver, Pairwise kidney exchange, J. Econom. Theory 125 (2005) 151–188.

[13] D. Saban, J. Sethuraman, House allocation with indifferences: a generalization and a unified view, in: Proceedings of the 14th ACM Conference on Electronic Commerce (ACM-EC), ACM Press, 2013, pp. 803–820.

[14] L.S. Shapley, H. Scarf, On cores and indivisibility, J. Math. Econom. 1 (1) (1974) 23–37.

[15] M. Xiao, X. Wang, Exact algorithms and complexity of kidney exchange, in: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18, International Joint Conferences on Artificial Intelligence Organization, 2018, pp. 555–561.