Exchange Market Mechanisms without Money

Zeinab Abbassi[1], Nima Haghpanah[2], Vahab Mirrokni[3]

[1] Columbia University, zeinab@cs.columbia.edu
[2] MIT, nima@csail.mit.edu
[3] Google Research, New York, mirrokni@google.com

Abstract

We introduce and study the following exchange market mechanism problem without money: consider a set of agents who wish to obtain a set of items, and have a set of items to offer to others. An exchange market mechanism specifies for each agent a subset of items to give away, and another subset of items that he would receive in exchange. Each agent would like to maximize the number of items he receives from his wish list, but will have a large dis-utility if he gives away more items than what he receives. Our goal is to design *truthful* mechanisms that maximize the total number of items being exchanged. This problem is a generalization of the kidney exchange problem, and is motivated by several barter exchange websites on the Internet. We show that an exchange can be viewed as a collection of simple cycles in a directed bipartite graph with agents and items as the two sides. Any cycle represents a trade in which each participating agent gives and receives exactly one item. We study two variants of the exchange market problem: the length-constrained variant where the number of items exchanged in each cycle should be at most a given constant $k \geq 2$, and the unconstrained variant where cycles of any length are allowed. For the length-constrained variant, we show that no truthful deterministic or randomized mechanism can achieve an approximation factor better than $\frac{3k+1}{3k+2}$ and $\frac{3k+1.89}{3k+2}$ respectively. We present a $\frac{1}{8}$-approximate truthful mechanism for the problem with $k = 2$. For the unconstrained version, we present a polynomial-time algorithm solving the optimal exchange market.

1 Introduction

Mechanism design without money has been a major subject of study in economics and mechanism design [18, 9, 19]. This line of research has been studied in the economics literature in the context of two-sided matching markets [9, 18], markets where monetary transactions are repugnant [20], and house allocation problems [24]. Recently this field has gained more attention in the computer science literature due to the fact that monetary compensations are not always easily applicable [17, 8]. In some cases, payments are hard to implement or to collect, e.g., implementing secure money transaction systems is costly in general and some people do not feel safe enough sharing sensitive information online fearing internet fraud [7, 23]. Moreover, in some repugnant markets, there may be legal or ethical issues with monetary transactions, e.g., in the case of kidney donation [21, 3]. In this paper, we initiate the study of a fundamental exchange market problem without money that is a natural generalization of the well-studied kidney exchange problem. From the practical point of view, the problem is motivated by barter websites on the Internet, e.g., swap.com, and u-exchange.com.[4] We will elaborate on these applications after the problem description.

Consider a set of agents where each agent has some items to offer, and wishes to receive some items from other agents. A mechanism specifies for each agent a set of items that he gives away, and a set of items that he receives. Each agent would like to receive as many items as possible from the items that he wishes, that is, his utility is equal to the number of items that he receives and wishes. However, he will have a large dis-utility if he gives away more items than what he receives, because he considers such a trade to be unfair. To ensure voluntary participation (also known as individual rationality), we require the mechanism to avoid this. We show that any individually rational exchange can be viewed as a collection of directed cycles, in which each agent receives an item from the agent before him, and gives an item to the agent after him. In addition to simplifying the statement of the problem, this suggests that we can implement an exchange by separately carrying out one-to-one trades among subsets of agents. In some settings, carrying out cycle-exchanges of large size is undesirable or infeasible. If there is a chance that each trade in a cycle fails, the chance that the whole cycle of exchanges is realized will exponentially decrease as the length of the cycle increases. Because of this and other problems with implementation, for example, most of the previous work on the exchange of kidneys focuses on short exchanges [21, 22]. Therefore, we distinguish the restricted problem in which the number of agents in each cycle is bounded above by some given constant $k \geq 2$. The most natural and commonly practiced cycles are of length 2, (i.e., swaps). Most of the results of this paper are for the two extremes in which there is no limit on the length of the cycles, and $k = 2$.

As an example, consider a simple instance with 3 agents and 4 items, where agent a owns item 1, agent b owns item 2, and agent c owns items 3 and 4. Assume that agents a and c both wish to receive item 2, and agent b wishes to receive items $1, 3$, or 4. Therefore, agents a and c each would like to be the one who gets the chance to trade his item(s) for item 2. Now consider another instance in which agent c does not own item 4. Consider a mechanism that, given the first instance, picks agent a to trade with b, but given the second instance, picks agent c to trade with b. Then, if c truly owns both 3 and 4, he would prefer to claim that he only owns 3, and be the person who trades with b. The problem can be easily fixed here by making consistent decisions. The question is, then, can we design a mechanism that always finds the exchange that maximizes social welfare, and yet incentivizes truthfulness? Interestingly, we show that the answer differs in unconstrained and constrained problems. We elaborate upon this further after reviewing some real-world applications of this model.

Applications. Motivated by concerns about money transaction on the Internet, and simplicity and convenience of swapping items in local economies, barter websites (also referred

[4] See http://abcnews.go.com/International/buying-barter-economy-matures-niche-trend/story?id=18193023 for a recent news coverage.

to as barter economy sites) have become more popular in the recent years.[5] Such barter websites help users exchange items with each other. Various types of items may be exchanged in these websites: from smaller used items like books, DVDs, cellphones, or children's clothing, to bigger items like boats, vehicles and vacation rentals. Some of these sites also support exchanging services like dental work and installing hardwood flooring. In most cases, users swap items with one another, i.e., only exchanges of size 2 are allowed. One can extend their setting to multiple exchanges over a cycle at the same time. We model such barter websites as networks amongst users where each user has two associated lists: an item list which consists of items the user is willing to give away to other users, and a wish list which consists of items the user is interested in receiving. A transaction involves a user giving an item to another user. Users are motivated to transact in expectation of realizing their wishes. Some examples of such marketplace applications are as follows:

- swap.com focuses on media like books and CDs. It claims about 1.2 million members and focuses on more local trades as they want to avoid expensive shipping fees.
- readitswapit.co.uk allows book lovers to exchange their already read books and receive new books in return. Almost all of the matching is done manually by the user herself, meaning that she has to go and find her desired book in a library and then mark it. The owner of the desired book will be informed by an email and will check the seeker's list of books and if willing to do the exchange, they will post the books for each other.

Other than these applications, the aforementioned exchange market problem without money is a natural generalization of the well-studied kidney exchange problem [21, 2, 3] where each agent wishes one item (a healthy kidney) and has only one item to offer.

Our Contributions. For the length-constrained variant of the problem, we rule out the existence of a $1-o(1)$-approximate truthful mechanism for $k \geq 2$. We show that no truthful deterministic or randomized mechanism can achieve an approximation factor better than $\frac{3k+1}{3k+2}$ or $\frac{3k+1.89}{3k+2}$, respectively.

The above impossibility results are caused by incentive issues, and are not based on the computational complexity of the problem. We strengthen the hardness of the problem by proving that, even without the truthfulness requirement, the problem is APX-hard for any k. Finally, we present a $\frac{1}{8}$-approximately optimal truthful mechanism for the problem with $k = 2$. The mechanism visits pairs of agents in some fixed order, and considers adding a subset of exchanges when visiting a pair. The ordering of pairs is done such that, at any stage during the process, an agent can not affect the relevant future cycles by changing his strategy. We formalize this by defining an interaction set for each agent, which denotes the

[5] See examples at http://mashable.com/2011/08/19/barter-sites/ and http://gigaom.com/2012/07/07/summer-is-for-swapping-startups-boost-the-barter-economy/. For a recent news coverage, see http://abcnews.go.com/International/buying-barter-economy-matures-niche-trend/story?id=18193023.

set of agents that are (possibly indirectly) affected by him at any stage, and making sure that an agent does not trade with any other agent who is currently in his interaction set.

For the unconstrained version, we present a class of polynomial-time algorithms solving the optimal exchange market problem, closely following algorithms for maximum flow and circulation problems. An algorithm maintains a set of feasible exchanges, and iteratively augments the current solution until the residual graph does not contain any more cycles.

1.1 Related Work

The Kidney Exchange Problem A related problem in exchange markets is the "national kidney exchange" problem. For many patients with kidney disease, the best option is to find a living donor – a healthy person willing to donate one of her two kidneys. The problem is that frequently, a potential donor and her intended recipient are blood or tissue-type incompatible. In the past, the incompatible donor's kidney was not used, and the patient had to wait for a deceased-donor kidney. However, now through regional kidney exchange programs in the United States, patients can swap their incompatible donors with each other, in order to each obtain a compatible donor [21, 22, 2, 3]. The kidney exchange problem is a special case of our problem where each user has only one item to offer and wishes one item. As a result, the kidney exchange problem is fundamentally simpler, and can be solved in polynomial time in the case of one-to-one exchanges (i.e., for $k = 2$), however, for the case of length-constrained exchanges even with $k = 2$, our problem is NP-hard (and also APX-hard as observed in this paper). From the mechanism design point of view, [3, 5] study the kidney exchange problem in the presence of strategic hospitals that may have an incentive not to list all their current available organ donors. By not listing donors, hospitals will still have the option of matching pairs internally. In our model, however, agents can have positive utility only by exchanging items.

Mechanism Design without Money Our work fits in a line of research that seeks to design strategy-proof mechanisms without monetary transfers. This line of research has been studied in the economics literature in the context of two-sided matching markets, repugnant markets, and house allocation problems:

- Two-sided matching markets have applications in college admissions and allocating interns to hospitals [19]. Incentive issues in such markets have been studied in several papers [13, 4]. A special class of the stable matching problem with dichotomous preferences, studied by [6], is remotely related to our problem as it employs the theory of bipartite matchings.
- Repugnant markets are markets that are considered by society to be outside of the range of market monetary transactions, due to moral issues. It applies to organ donation (like exchange of kidneys), and reproduction (e.g., child adoption and surrogate mothers). As discussed earlier, our problem can be thought of as a generalization of the kidney exchange problem.

– House allocation problems [24] are resource allocation problems where a set of items (houses) are to be allocated to a set of people each with a preference list over the items. These problems have applications in organ allocation (e.g., deceased donor waiting list), university dormitory room, parking space, and office space allocation.

Although about mechanism design without money, none of the above papers discuss approximately optimal truthful mechanisms. Recently, approximate mechanism design without money has become more popular in the computer science community. Initiated by [17], various assignment problems [8] as well as network design problems [15, 16] have been studied in this context. Our results fit in a similar framework, but our exchange market problem and the techniques we employ are different from all the above problems.

Algorithmic Results The length-constrained variant of the problem from an algorithmic perspective was studied in [1], where it is shown that the problem for length constraint of $k = 2$ is NP-hard, and a 5/3-approximation is deriven using a reduction to the k-set packing problem. The bounded cycle cover problem, which constrains cycles to be of bounded size as well as simple and node-disjoint, was introduced by [11]. They present a heuristic for the problem along with empirical analysis. For the unconstrained exchange market problem, we design a polynomial-time algorithm which is a variant of the well-studied minimum cost circulation problem [10].

2 Preliminaries

Consider a set of n agents A and a set of m items I. In an instance $(A, I, \{(I_a, W_a)|a \in A\})$ of the *exchange market* problem, each agent a has an *item list* $I_a \subseteq I$ (items that he owns) and a *wish list* $W_a \subseteq I$ (items that he needs) such that $I_a \cap W_a = \emptyset$. An *exchange* $C : A \to I^2$ assigns to each agent a a set $C_1(a)$ of items that he receives in exchange for a set of items $C_2(a)$ that he gives away. An exchange is feasible if for each item i, $|\{a|i \in C_2(a)\}| \geq |\{a|i \in C_1(a)\}|$. The utility of agent a for exchange C is specified by a function u as follows:

$$u(a, C) = \begin{cases} |C_1(a) \cap W_a| & \text{if } C_2(a) \subseteq I_a \text{ and } |C_1(a) \cap W_a| \leq |C_2(a)|, \\ -\infty & \text{otherwise.} \end{cases}$$

In other words, in a feasible exchange, the utility of an agent is $-\infty$ if he must provide an item he does not own, or has to give away more items than he receives, and otherwise, his utility is the number of items that he receives from his wish list.

Our goal is to find a feasible exchange maximizing the *social welfare*, i.e., sum of utilities of agents. This goal corresponds to finding a feasible exchange that maximizes the number of items exchanged, and no agent has $-\infty$ utility. Notice that feasibility implies that the total number of items collected from agents must be at least the total number of items received by agents. Therefore, in order to find an exchange with non-negative total social welfare, we should make sure that the number of items each agent receives *and* is in his wish list is not more than the number of items he gives away, i.e., we must have $C_1(a) \subseteq W_a$, and also $|C_1(a)| = |C_2(a)|$.

An *exchange mechanism* extracts the private information, i.e, I_a and W_a of each agent a, and maps it to an exchange. We are interested in designing *truthful* (or strategyproof) mechanisms in which it is a dominant strategy for each agent a to report his true private information (I_a, W_a). Our goal is to design truthful exchange mechanisms maximizing the social welfare. We will give a more formal definition of the problem after defining a bipartite graph representation of the problem.

2.1 Bipartite Graph Modeling

The above formulation of the problem allows for a clean representation using a directed bipartite graph. This representation will help in deriving a polynomial-time algorithm for the unconstrained problem and also argue about the constrained-length problem. This representation will be used throughout the rest of the paper. Here, we first define this representation, and then formally state our problem in terms of this graph representation.

Given an instance $(A, I, \{(I_a, W_a)|a \in A\})$ of the exchange mechanism problem, define a bipartite directed graph $G = (A \cup I, E)$, where $E = \{(a, i)|i \in I_a\} \cup \{(i, a)|i \in W_a\}$. That is, there is an edge from an agent to an item if the agent owns the item, and from an item to an agent if the agent needs the item. A (directed) cycle in graph G is a sequence of directed edges in G where each edge appears at most once. This graph representation helps in arguing about the exchange market problem, since *any feasible exchange in the exchange market problem corresponds to a set of edge-disjoint directed cycles in G and vice versa.*

Proposition 1. *In an exchange market problem $(A, I, \{(I_a, W_a)|a \in A\})$, any feasible exchange corresponds to a set of* edge-disjoint directed cycles *in its graph representation $G(A \cup I, E)$, and vice versa.*

Proof. First of all, we can interpret a simple directed cycle in this graph as a feasible exchange as follows: any agent in this cycle gives away the item immediately after him in the cycle, and receives the item immediately before him. More generally, any set of edge-disjoint cycles can be interpreted as a feasible exchange. Conversely, any feasible exchange corresponds to a subgraph in which the in-degree of each vertex is equal to the out-degree $(|C_1(a)| = |C_2(a)|)$, and therefore it can be decomposed to a set of edge-disjoint cycles. The utility of an agent in each cycle is equal to the number of cycles to which his corresponding vertex belongs and in which he receives an item in his wish list. $\square$

We can now formally state the definition of *truthful exchange mechanisms* using our graph representation of the problem. Truthfulness states that by misreporting, an agent will either be asked to provide an item he does not own or receive fewer items that he wants.

Definition 1. *Consider a bipartite graph $G(A \cup I, E)$ and let $a \in A$ be a vertex where I_a is the outgoing neighbors of a and W_a is incoming neighbors for a. Consider any other subsets $I'_a \subseteq I$ and $W'_a \subseteq I$, $I'_a \cap W'_a = \emptyset$, and let G' be the graph representation of the problem where we replace (I_a, W_a) with (I'_a, W'_a) in G. An exchange mechanism is* truthful *if for any such G and G', in the set of cycles produced by the mechanism on G' there is either a cycle with an edge from a to $I \backslash I_a$, or the number of cycles with edges from W_a to a is no more than the number of cycles including a in G.*

We can now define the *unconstrained exchange market* problem as follows:

Definition 2. *Given a graph representation $G(A \cup I, E)$ of an instance $(A, I, \{(I_a, W_a)|a \in A\})$ of the exchange market problem, the goal of the* unconstrained exchange maximization *problem is to find a set E of edge-disjoint (directed) cycles with the maximum number of edges.*

We also define the *length-constrained* exchange market problem, or equivalently, *the k-exchange market* problem as follows:

Definition 3. *Given a graph representation $G(A \cup I, E)$ of an instance $(A, I, \{(I_a, W_a)|a \in A\})$ of the exchange market problem and a constant k, the goal of the* length-constrained exchange maximization *problem is to find a set E of edge-disjoint cycles, each of size at most $2k$, with the maximum number of edges.*

Note that by restricting the size of the cycles by $2k$, rather than k, we are limiting the number of agents (equivalently, items) in each cycle by k. For example, the case of $k = 2$ corresponds to swapping two items among two agents, and thus a cycle of size 4 in the bipartite graph.

Let $OPT(G)$ be the maximum social welfare of a set of feasible exchanges given a graph G. In this paper, we are interested in designing truthful mechanisms to approximate $OPT(.)$ on every instance. We say that an algorithm f is α-approximation if for all G, $S(f(G), G) \geq \alpha OPT(G)$. Notice that any approximate mechanism avoids selecting an infeasible exchange, since the optimal social welfare is always at least 0, achieved by not picking any exchanges.

3 Length-Constrained Exchange Markets

In this section, we study the length-constrained exchange market problem. We show several impossibility results from truthful mechanism design and computational complexity point of views for any $k \geq 2$, and one approximately optimal mechanism for the length-constrained problem with $k = 2$.

3.1 Inapproximability of truthful mechanisms

In this section, we show the inapproximability of truthful mechanisms for length-constrained market exchange problem for $k \geq 2$. First we show a result for deterministic mechanisms and then extend it to randomized mechanisms.

Theorem 1. *No deterministic truthful mechanism for the k-constrained problem can have an approximation ratio better than $\frac{3k+1}{3k+2}$.*

Proof. Consider an instance of the k-exchange market problem with $k+1$ agents $a, b, c_1, \ldots, c_{k-1}$ and $3k+3$ items. Each agent owns 3 items (exclusively), and each item is in the wish list of one or two other agents. Each item is coded by a pair, where the first element is the agent owning it, and the second element is the agent(s) wishing for it. The agents and items are,

- Items owned by a: $(a, b), (a, c_1), (a, bc_1)$.
- Items owned by b: $(b, a), (b, c_1), (b, ac_1)$.
- Items owned by c_i for $1 \leq i \leq k-2$: $(c_i, c_{i+1}), (c_i, c_{i+1})', (c_i, c_{i+1})''$.
- Items owned by c_{k-1}: $(c_{k-1}, ab), (c_{k-1}, ab)', (c_{k-1}, ab)''$.

For example, (b, ac_1) is an item that is owned by agent b, and is wished for by agents a and c_1. Agents a and b are symmetric in this instance. Notice that any cycle involving any agent c_i for $1 \leq i \leq k-1$, also involves all such agents. In particular, any cycle involving c_1 involves all c_i, $1 \leq i \leq k-1$, and therefore has size at least $k-1$. We conclude that no feasible cycle can involve a, b, and c_1 all together (otherwise it will have size $k+1$). Also notice that no feasible cycle can involve exactly one of a, b, or c_1. As a result, the sum of the utilities of a, b, and c_1 for any feasible cycle is an even number. Since there are 9 items that a, b, and c_1 want, the sum of the utilities of a, b, and c_1 in any feasible exchange is at most 9. But since the sum of their utilities is an even number, it can be at most 8. We therefore conclude that at least one of these three agents will have utility at most 2 in any feasible solution to this instance, regardless of the approximation factor. We next show by considering cases that that agent will benefit from misreporting. Since a and b are symmetric, there are two cases:

1. Agent a's utility is at most 2 (the case for agent b is similar). Assume that agent a removes item (b, a) from his wish list. The following exchange is still feasible, and has social welfare $3k+2$:
   - $(a, b), (b, ac_1)$.
   - $(a, c_1), (c_i, c_{i+1})$ for $1 \leq i \leq k-2$, and (c_{k-1}, ab).
   - $(a, bc_1), (c_i, c_{i+1})'$ for $1 \leq i \leq k-2$, and $(c_{k-1}, ab)'$.
   - $(b, c), (c_i, c_{i+1})'$ for $1 \leq i \leq k-2$, and $(c_{k-1}, ab)''$.
2. Agent c_1's utility is at most 2. Assume that agent c_1 removes item (a, c_1) from his wish list. The following exchange is still feasible, and has social welfare $3k+2$:
   - $(a, b), (b, a)$.

– $(a, bc_1), (c_i, c_{i+1})$ for $1 \leq i \leq k-2$, and (c_{k-1}, ab).
– $(b, c_1), (c_i, c_{i+1})$ for $1 \leq i \leq k-2$, and $(c_{k-1}, ab)'$.
– $(b, ac_1), (c_i, c_{i+1})$ for $1 \leq i \leq k-2$, and $(c_{k-1}, ab)''$.

Notice that in each case an agent removed an item from his wish list that was exclusively wanted by him. Therefore, in each instance the social welfare can be at most $3k+2$, and the specified exchange is optimal. In the specified exchange the agent removing an item has utility 3. In fact, we show that in each the agent removing his item will have utility 3 in any exchange with social welfare $3k+2$.

1. Assume for contradiction that a has utility at most 2 in an exchange with social welfare $3k+2$. Since there are k other agents and the utility of each agent is at most 3, all other agents must have utility 3. But since a participates in at most 2 exchanges, the number of items that either b and c_1 wants that are offered is at most 5. Therefore b and c_1 can not both have utility 3.
2. This case is similar. Assume that c_1 has utility at most 2 in an exchange with social welfare $3k+2$. Since c_1 participates in at most 2 exchanges, the number of items that either a and c_1 wants that are offered is at most 5. Therefore a and c_1 can not both have utility 3.

We conclude that the agent removing his item will have utility 3 in any exchange with social welfare $3k+2$. Since any algorithm with approximation factor $\frac{3k+1}{3k+2}$ must choose such an exchange, it can not be truthful. □

A randomized mechanism may choose exchanges at random. In this case, we assume that agents are risk-neutral and try to maximize their *expected utility*. The question is if it is possible to design a truthful mechanism that does not give any incentive to agents to misreport their private information in order to increase their expected utility. The proof of the following theorem is based on the instance of Theorem 1, and is deferred to the appendix.

Theorem 2. *No randomized truthful mechanism for the k-exchange problem can have an approximation factor better than $\frac{3k+\frac{17}{9}}{3k+2}$.*

3.2 Truthful $\frac{1}{8}$-approximation for the 2-exchange problem

In this section, we present a $\frac{1}{8}$-approximation truthful mechanism for the length-constrained exchange problem with $k=2$. The algorithm is as follows:

1. Partition agents into sets A and B by placing each agent independently at random with probability $1/2$ into set A (and otherwise in B).
2. Let $a_1, \dots, a_k$ be the agents in A, and $b_1, \dots, b_{n-k}$ the agents in B.
3. Visit every pair of agents in $A \times B$ in order $(a_1, b_1), (a_1, b_2), \dots, (a_1, b_{n-k}), (a_2, b_1), \dots, (a_k, b_{n-k})$.

4. When visiting a pair of agents, consider exchanging all pairs of items in an arbitrary order, and add that exchange if feasible.

First, we show the above algorithm is a $\frac{1}{8}$-approximation algorithm, and then we show it corresponds to a truthful implementation.

Lemma 1. *The above algorithm is a* $1/8$ *approximation algorithm for the* 2*-exchange market problem.*

Proof. Consider an optimum set of exchanges OPT. Let $OPT(A)$ be the subset of OPT consisting only of exchanges between an agent in A and an agent in B. Since every element of OPT will be in $OPT(A)$ with probability $1/2$, we must have $E_A[|OPT(A)|] \geq |OPT|/2$. Fixing A, the algorithm heuristically considers adding exchanges in $A \times B$. Since each possible exchange intersects with at most 4 exchanges in $OPT(A)$, and also every element of $OPT(A)$ has intersection with at least an exchange picked by the algorithm (otherwise it would have been picked), this algorithm picks at least $OPT(A)/4$ exchanges. This implies that the algorithm is a $1/8$-approximation. $\square$

We next prove the truthfulness of the algorithm by showing that it satisfies a property, which we call *interaction-freeness*, that is a sufficient condition for truthfulness of greedy algorithms. Equivalently, we say that the algorithm is interaction-free. A greedy algorithm fixes an ordering over a subset of all exchanges, visiting them one by one, adding a cycle whenever it is feasible (according to the item lists and wish lists of the two agents), and it does not intersect with a cycle that is already picked. At any time during the process of the algorithm, and for any agent a, define the *interaction set* $S(a) \subseteq A$ to be the set of agents that are already affected (possibly indirectly) by a as follows: At the start of the process, let $S(a) = \{a\}$ for all a. Assume that (b, c) are the agents currently considered for an exchange. Then for any agent a such that $b \in S(a)$ (respectively for c), update $S(a)$ by adding c (respectively b). Intuitively, a greedy algorithm is interaction-free if for any two agents that are being considered for the first time in the algorithm, they have not previously interacted, i.e., they are not in each other's interaction set. More formally, a greedy algorithm satisfies the interaction-free property, if for any two agents a and b, whenever an exchange involving agents a and b is considered in the process of the greedy algorithm, one of the following is true:

- $a \notin S(b)$ and $b \notin S(a)$, or
- the only exchanges that are already considered for agents a and b involve both a and b.

Lemma 2. *Any greedy algorithm that satisfies the interaction-free property is truthful.*

Proof. Fix an agent a. For any $b \neq a$ such that an exchange between b and a is ever considered, let $\hat{I}_b \subseteq I_b$ and $\hat{W}_b \subseteq W_b$ be the sets of items that are not used in any exchange before the first time an exchange between a and b is considered. Directly from the definition of interaction-freeness, it follows that $\hat{I}_b$ and $\hat{W}_b$ are independent of the strategy of a. That is, from the perspective of agent a, the algorithm ranks a subset of other agents with $\hat{I}_b$ and $\hat{W}_b$, and then considers greedily adding a subset of all possible exchanges according to the ranking. To prove truthfulness, we only need to show a does not benefit by misreporting in this simple procedure.

Consider an order σ on a subset of exchanges each involving a. Assume for contradiction that a benefits from reporting I'_a and W'_a. That is, a's utility when reporting truthfully is k, and his utility when reporting I'_a and W'_a is $k' > k$. Since all the exchanges involve a, his utility when reporting truthfully is equal to the number of cycles picked by the algorithm. Let $C = \{c_1, \ldots, c_k\}$ be such a set. Let $C' = \hat{C} \cup \bar{C}$ be the set of cycles picked when a misreports, where $\hat{C}$ is the set of exchanges in which a does not receive an item that he wants, and $\bar{C}$ is the set of exchanges in which a receives and item he wants (and therefore $|\bar{C}| = k'$). Assume that the set of items a gives away in C' is a subset of I_a, since otherwise he will have a large dis-utility. Let σ' be the projection of σ on $C \cup C'$. The outcome of the greedy algorithm on σ and σ' is the same, and therefore i will still benefit by misreporting in σ'. We can further assume that $C \cap C' = \emptyset$, by removing any element of $C \cap C'$ from σ'. Removing such an element will decrease i's utility of being truthful and his utility by misreporting by the same amount. Notice that for any element $\bar{c}$ of $\bar{C}$ there must be an element of C that has non-empty intersection with $\bar{c}$ (otherwise $barc$ would have been picked by the algorithm when a reports truthfully). We say that c covers $\bar{c}$ in this case. Let $C_1 \subseteq C$ be all the elements that cover at most one element of $\bar{C}$, and let $\bar{C}_1 \subseteq \bar{C}$ be all the elements that are covered by C_1. Remove C_1 and $\bar{C}_1$ from σ'. Notice that i would still benefit from misreporting since the number of elements removed from C is at least that of $\bar{C}$. So we can assume that any element of C now covers at least two elements from $\bar{C}$. Consider the first element of $C \cup \bar{C}$ according to σ'. Such a cycle must be in C, since otherwise it would have been picked when a is truthful. Let c be this element. Since c is not picked when a misreports, there must be an element $\hat{c}$ of $\hat{C}$ that appears before c in σ', and has intersection with it. That is, there are 3 elements of C' that have intersection with c. Recall that all these cycles involve a. Let items i and j be the items a gives and receives in c, respectively. For a cycle to have intersection with c, either a must give i or receive j. This implies that there are at most two non-intersecting cycles that intersect with c. This is a contradiction to the feasibility of C'. □

Lemma 3. *The above algorithm is a truthful algorithm.*

Proof. Consider the time when the algorithm first visits pair (a_i, b_j). It can be shown inductively that $S(a_i) = \{b_1, \ldots, b_{j-1}\}$, and $S(b_j) = \{a_1, \ldots, a_{i-1}, b_{j+1}, \ldots, b_n\}$. This shows that $a_i \notin S(b_j)$ and $b_j \notin S(a_i)$. □

We next show by an example that a very simple violation of interaction-free-ness property by a greedy algorithm can make it not truthful.

Example 1. Assume that we have 3 agents and 5 items, and consider the instance shown in Figure 1. Consider an order σ such that $\{(a,2),(b,1)\} \succ_\sigma \{(a,3),(b,1)\} \succ_\sigma \{(a,3),(c,5)\} \succ_\sigma$

$\{(b,4),(c,5)\}$ (what σ does on the rest of the exchanges is irrelevant and thus not represented here for simplicity). This means that the algorithm first considers an exchange in which agent b gives item 1 to agent a and receives 2, and so on. The greedy algorithm parameterized by σ chooses $\{(a,2),(b,1)\}$ and $\{(a,3),(c,5)\}$. The valuation of agent b for this set of exchanges is 1. Now assume that b misreports his wish list as $W'_b = \{3,5\}$, instead of the true set which is $W_b = \{2,3,5\}$. The outcome of the algorithm on this instance is going to be $\{(a,3),(b,1)\}$ and $\{(b,4),(c,5)\}$. The valuation of b for this set is 2. Therefore, the static algorithm parameterized by σ is not truthful.

Notice that this algorithm violates the interaction-free-ness property because at the time when the last exchange is considered between b and c, we have $S(b) = \{a,b,c\}$ and $S(c) = \{a,c\}$. Since $c \in S(b)$, thus, the algorithm is not interaction-free.

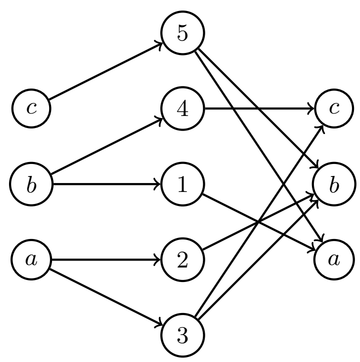

Fig. 1. The exchange market of Example 1. For the ease of visualization, each vertex representing an agent is split into two vertices.

3.3 Computational Complexity

In this section, we show that the length-constrained market exchange problem is APX-hard for any $k \geq 2$. First, we discuss the case of $k = 2$. To prove APX-hardness for $k = 2$, we use the fact that there exists a factor-preserving reduction from the edge-disjoint 3-cycle partitioning of 3-partite graphs to the market exchange problem with $k = 2$ [1]. Here, we show that edge-disjoint 3-cycle partitioning of 3-partite graphs is APX-Hard, which in turn, implies our desired result. Formally, the problem edge-disjoint 3-cycle partitioning of 3-partite graphs problem, given a tripartite graph $G(V_1, V_2, V_3; E)$ where V_1, V_2 and V_3 are disjoint sets of vertices, and $E \subseteq \{V_1 \times V_2\} \cup \{V_1 \times V_3\} \cup \{V_2 \times V_3\}$, the goal is to find the maximum number of edge-disjoint triangles in G. A factor-preserving reduction from EdgeDisjTrianglePar to the 2-exchange market problem is given in [1]. To show the

APX-hardness of 2-exchange market problem, it remains to prove that EdgeDisjTrianglePar is APX-hard.

Holyer [12] proved that edge-partitioning of general graphs into edge-disjoint triangles is NP-complete. A more careful analysis of this proof shows that the edge-partitioning of general graphs is APX-hard [14]. We first note that the set of graphs that Holyer used in his NP-hardness proof for edge-partitioning triangles is in fact tripartite. To show this, we define some notations from [12]. Let graph $H_{3,n}$ be a graph with n^3 vertices $V = \{(x_1, x_2, x_3) \in \{0, 1, 2\}^n \mid \sum_{i=1}^{3} x_i = 0(\text{mod } n)\}$. Let $((x_1, x_2, x_3), (y_1, y_2, y_3))$ be an edge in $H_{3,n}$ if there exists $i, j \in \{1, 2, 3\}$, $i \neq j$, such that $x_k = y_k(\text{mod } n)$ for $k \neq i, j$ and $y_i = (x_i + 1)(\text{mod } n)$ and $y_j = (x_j + 1)(\text{mod } n)$. The resulting graph reduced from any 3SAT instance in Holyer's proof is a result of combining and joining $H_{3,p}$'s. It is not hard to verify that $H_{3,n}$ is 3-vertex-colorable and any combination and joint of these graphs is also 3-vertex-colorable. As a result, Holyer's proof of NP-hardness [12] and its extension for APX-hardness [14] of edge-partitioning of general graphs implies the APX-hardness of EdgeDisjTrianglePar. This, in turn, implies that 2-exchange market problem is APX-hard.

The APX-hardness proof for the k-exchange market problem where $k > 2$ is very similar to that of the 2-exchange market problem. First, one can give a similar factor-preserving reduction from the problem of edge-partitioning of a k-partite graph to k-cycles to the k-exchange market problem. Now the APX-hardness of the k-exchange market problem boils down to the APX-hardness of edge-partitioning of k-partite graphs to k-cycles, which can be shown by giving a reduction from EdgeDisjTrianglePar to edge-partitioning of a k-partite graph to k-cycles. To see this, given an instance $G(V_1, V_2, V_3; E)$ of EdgeDisjTrianglePar, we construct a k-partite graph $G'(U_1, U_2, \ldots, U_k; E')$ where $U_1 = V_1, U_2 = V_2, U_3 = V_3$, and for $i \geq 4$, $U_i = E_3$ where $E_3 = E \cap \{V_2 \times V_3\}$, i.e., each node in U_i corresponds to an edge $e \in E_3$ from V_2 to V_3. Denote by $e_4, e_5, \ldots, e_k$ the $k-3$ nodes in G' corresponding to edge $e \in E_3$. We also form the edges of $E' = \cup\{(u, v)|u \in V_2, v \in V_3$, as follows:

- include all edges from nodes in U_1 to nodes in U_2 and U_3 for each pair of nodes whose corresponding nodes in G are connected, i.e., add $\{(u, v)|u \in V_1, v \in V_2 \cup V_3, (u, v) \in E(G)\}$.
- for each edge $e = (u, v) \in E_3$, add the following edges to E': $(u, e_4), (e_4, e_5), \ldots, (e_{k-1}, e_k), (e_k, v)$.

It is not hard to see that any triangle (w, u, v) where $(u, v) = e \in E_3$ in graph G corresponds to the k-cycle $(w, u, e_4, e_5, \ldots, e_k, v)$ in G' and vice versa. G is a tripartite graph and G' is a k-partite graph. As a result, the above is a factor-preserving reduction from EdgeDisjTrianglePar to the problem of edge-partitioning a k-partite graph to k-cycles. Therefore, APX-hardness of k-exchange market problem follows from APX-hardness of EdgeDisjTrianglePar.

4 Unconstrained Exchange Market Problem

In this section, we give a polynomial-time algorithm for the unconstrained exchange market problem. As we stated earlier, we would like to maximize the number of edges by picking a set of edge-disjoint cycles. One can write this problem as a maximum circulation problem (or minimum cost circulation problem with negative cost on the edges [6]), and solve it in polynomial time using an algorithm that interactively improves the current solution with a cycle in an augmenting graph [10]. We will present a variant of this algorithm that satisfies a desired set of properties (e.g, a specific monotonicity property). We start by a high-level description of the solution for the maximum circulation problem [10].

Given a directed unweighted anti-symmetric graph G, a flow is a function $f : G \to \mathbb{Z}$. Flow f is feasible in G if it satisfies:

- $\forall e \in G, f_e \leq 1$, and $\forall e \notin G, f_e \leq 0$,
- $f_{(u,v)} = -f_{(v,u)}$,
- For any vertex v, $\sum_{e \sim v} f_e = 0$, where $e \sim v$ if v is an endpoint of e.

The goal is to find a feasible flow that maximizes *weight* $w_G(f) = \sum_{e \in G} f_e$. Given a flow f, we now define the residual graph G_f corresponding to f:

Definition 4. *Given a graph G and a circulation f, we define the residual graph corresponding to f to be $G_f(V, E_f)$ with $E_f = \{(u,v) \in E(G) | f(u,v) = 0\} \cup \{(v,u) \in E(G) | f(u,v) = 1\}$.*

We say G_f admits a flow f' if f' is a feasible circulation flow in G_f. The following is a well-known result which connects optimality of the flows to absence of cycles with positive weight in the residual graph.

Lemma 4. *[10] A flow f is optimal if and only if its residual graph G_f does not admit any flow f' with $w_G(f') \geq 0$.*

The above lemma suggests the following algorithm for the exchange market problem:

1. Initialize flow $f := 0$, and maintain a feasible flow.
2. Construct G_f.
3. If G_f admits a flow f' with positive weight, augment f with f' by setting $f := f + f'$, and go back to Line 2. Otherwise, terminate.

Notice that since the weight of the integral flow increases by at least 1 (since optimal flows are integral) in each step, the pseudo-algorithm terminates after polynomial number of steps. Thus, if the selection of an augmenting flow is done in polynomial time in each step, the algorithm runs in polynomial time.

[6] The minimum cost circulation problem is as follows: Given a graph G and capacities and costs u_e and c_e for each edge $e \in E(G)$, find a circulation flow f with capacities $f(e) \leq u_e$, and the minimum total cost $\sum_{e \in E} c_e f(e)$. Our maximum circulation problem can be modeled by setting $c_e = -1$ for each edge in $E(G)$, and applying the algorithm for the minimum cost circulation problem [10].

Bibliography

[1] Zeinab Abbassi and Laks V. S. Lakshmanan. On efficient recommendations for online exchange markets. In *ICDE*, pages 712–723, 2009.

[2] David J. Abraham, Avrim Blum, and Tuomas Sandholm. Clearing algorithms for barter exchange markets: Enabling nationwide kidney exchanges. In *ACM Conference on Electronic Commerce*, pages 295–304, June 13-16 2007.

[3] I. Ashlagi and A. Roth. Individual rationality and participation in large scale, multi-hospital kidney exchange. In *ACM Conference on Electronic Commerce*, pages 321–322, 2011.

[4] Itai Ashlagi, Mark Braverman, and Avinatan Hassidim. Matching with couples revisited. In *ACM Conference on Electronic Commerce*, pages 335–336, 2011.

[5] Itai Ashlagi, Felix A. Fischer, Ian A. Kash, and Ariel D. Procaccia. Mix and match: A strategyproof mechanism for multi-hospital kidney exchange. In *ACM Conference on Electronic Commerce*, pages 305–314, 2010.

[6] Anna Bogomolnaia and Herve Moulin. Random matching under dichotomous preferences. *Econometrica,*, 72:257–279, 2004.

[7] M. Peralta D. L. Hoffman, Th. P. Novak. Building consumer trust online. *Communications of the ACM*, 42(4):80–85, 1999.

[8] Shaddin Dughmi and Arpita Ghosh. Truthful assignment without money. In *ACM Conference on Electronic Commerce*, pages 325–334, 2010.

[9] D. Gale and L. S. Shapley. College admissions and the stability of marriage. *Americal Mathematical Monthly*, 69:9–15, 1962.

[10] Andrew V. Goldberg and Robert Endre Tarjan. Finding minimum-cost circulations by canceling negative cycles. *J. ACM*, 36(4):873–886, 1989.

[11] D. Hochbaum and E. Olinick. The bounded cycle-cover problem. *INFORMS Journal on Computing*, 13(2):104–109, 2001.

[12] Holyer. The NP-completeness of some edge partitioning problems. *SIAM journal of Computing*, 10(3):713–717, 1981.

[13] N. Immorlica and M. Mahdian. Marriage, honesty, and stability. In *Symposium on Discrete Algorithms (SODA)*, 2005.

[14] N. Immorlica, V. S. Mirrokni, and M. Mahdian. Cycle cover with short cycles. In *STACS*, pages 641–653, 2005.

[15] P. Lu, X. Sun, Y. Wang, and Z.A. Zhu. Asymptotically optimal strategy-proof mechanisms for two-facility games. In *Proceedings of the 11th ACM conference on Electronic commerce*, pages 315–324. ACM, 2010.

[16] P. Lu, Y. Wang, and Y. Zhou. Tighter bounds for facility games. *Internet and Network Economics*, pages 137–148, 2009.

[17] Ariel D. Procaccia and Moshe Tennenholtz. Approximate mechanism design without money. In *ACM Conference on Electronic Commerce*, pages 177–186, 2009.

[18] A. E. Roth and M. Sotomayor. *Two-Sided Matching: A Study in Game-Theoretic Modelling and Analysis*. Cambridge University Press, 1991.

[19] A.E. Roth. The national resident matching program as a labor market. *JAMA. Journal of the American Medical Association,*, 275:1054–1056, 1996.

[20] Alvin E Roth. Repugnance as a constraint on markets. Technical report, National Bureau of Economic Research, 2006.

[21] Alvin E. Roth, Tayfun Snmez, and Utku Unver. Kidney exchange. *Quarterly Journal of Economics*, 19, 2004.

[22] Alvin E. Roth, Tayfun Snmez, and Utku Unver. Pairwise kidney exchange. *Journal of Economic Theory*, 125:151–188, 2005.

[23] H. Zhang S. Ba, A. B. Whinston. Building trust in online auction markets through an economic incentive mechanism. *Decision Support Systems*, 35(3):273–286, 2003.

[24] Tayfun Snmez and Utku Unver. House allocation with existing tenants: An equivalence. *Games and Economic Behavior*, 52:153–185, 2004.

APPENDIX

Proof. [of Theorem 2] The main idea of the proof is similar to that of Theorem 1, and it is based on the same instance. Recall that in any exchange of the original instance, one agent among a, b, and c_1 must have utility at most 2. This implies that one agent among a, b, and c_1 must have expected utility of at most 8/3 when outcomes are chosen at random.

Now consider the agent with utility at most 8/3, and assume that he removes the item specified in the above cases from his wish list. Let p be the probability that the algorithm picks an exchange with social welfare $3k + 2$. We argued before that the agent will have utility 3 in such an outcome. The expected utility of the agent is therefore at least $3p$. For the algorithm to be truthful, we must therefore have $p \leq 8/9$. This implies that the expected social welfare is at most $(3k + 2)8/9 + (3k + 1)/9$. The approximation ration of the algorithm is therefore at most $\frac{3k+\frac{17}{9}}{3k+2}$. □

Theorem 3. *For $k \geq 6$, no deterministic truthful algorithm for the k-exchange problem can have an approximation factor better than $\frac{k-1}{k}$.*

Proof. We show the claim by studying the instance shown in Figure 2.

In this instance, the set of agents is $\{a, b, c, d\} \cup \{a_1, \ldots, a_{k-2}\}$, and the set of items is $\{1, 2, 3, 4, 5\} \cup \{i_1, \ldots, i_{k-2}\}$.

First consider another instance in which i_{k-2} is not in the wish list of agent c. Notice that in this case, agents $a_1, \ldots, a_{k-2}$ can not be in any exchanges. Excluding those agents and their items, the set of items will be equal to $\{1, 2, 3, 4, 5\}$. Since each of these items is owned by one agent, the maximum utility in any exchange will be at most 5. Since c is the only agent owning 2 items, he is the only agent that has utility of 2 in an exchange with total utility 5. Such an exchange is possible. For example, $\{\{(c,4),(d,5)\},\{(c,3),(b,2),(a,1)\}$. This implies that the outcome of any algorithm with approximation factor better than $4/5$ has total utility exactly 5, and the utility of agent c is 2.

Now consider another instance in which i_{k-2} is in the wish list of c. This can add a new set of cycles involving $a_1,\ldots,a_{k-2}$. In order to have a cycle of size at most k, at most two other agents can be involved in any such cycle. Those two agents must be c and a. This means that the only new cycle of size at most k must be $C_1 = \{(c,4),(a,1),(a_1,i_1),\ldots,(a_{k-2},i_{k-2})\}$. This cycle produces total utility of $k > 5$. Notice that there is no other cycle that is edge-disjoint with C_1. This is because after removing C_1, the only remaining players are b, d, and c who now own only item 3. This implies that the outcome of any algorithm with approximation factor better than $\frac{k-1}{k}$ is C_1. Notice that c has utility 1 in this exchange.

Now notice that if i_{k-1} is in the wish list of c, he will prefer to not report it and get utility of 2 instead of 1. Therefore, c will have an incentive to misreport in any algorithm with approximation factor better than $\max(\frac{4}{5},\frac{k-1}{k}) = \frac{k-1}{k}$.

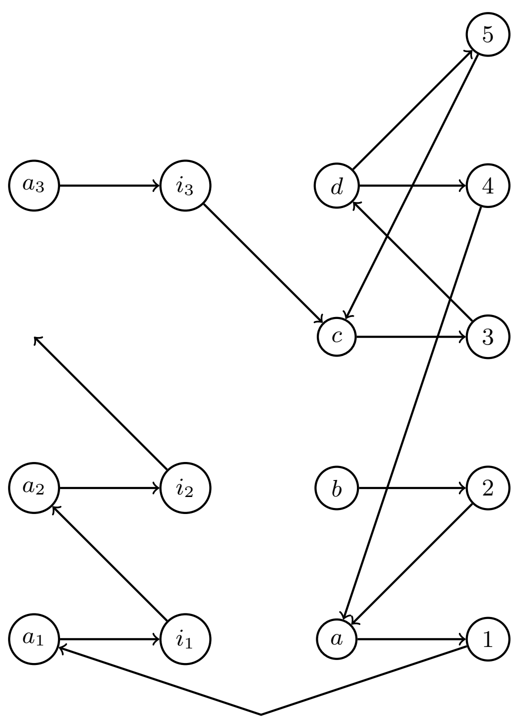

Fig. 2. The instance for which the optimum solution is not truthful. We have $I_a = \{1\}, I_b = \{2\}, I_c = \{3,4\}, I_d = \{5\}$.

□