Mix and Match: A Strategyproof Mechanism for Multi-Hospital Kidney Exchange*

Itai Ashlagi† Felix Fischer‡ Ian A. Kash§ Ariel D. Procaccia¶

October 15, 2011

Abstract

As kidney exchange programs are growing, manipulation by hospitals becomes more of an issue. Assuming that hospitals wish to maximize the number of their own patients who receive a kidney, they may have an incentive to withhold some of their incompatible donor-patient pairs and match them internally, thus harming social welfare. We study mechanisms for two-way exchanges that are strategyproof, i.e., make it a dominant strategy for hospitals to report all their incompatible pairs. We establish lower bounds on the welfare loss of strategyproof mechanisms, both deterministic and randomized, and propose a randomized mechanism that guarantees at least half of the maximum social welfare in the worst case. Simulations using realistic distributions for blood types and other parameters suggest that in practice our mechanism performs much closer to optimal.

1 Introduction

Transplantation of a healthy kidney is the best treatment today for severe kidney disease. Since humans normally have two kidneys and need only one to survive, many patients have a family member or friend willing to donate them a kidney. However, not all potential donors

*We have benefitted from valuable discussions with Moshe Tennenholtz and David Parkes. A shorter version of this paper appeared in the proceedings of the 11th ACM Conference on Electronic Commerce.

†Department of Operations Management, Sloan School of Management, Massachusetts Institute of Technology, iashlagi@mit.edu

‡School of Engineering and Applied Sciences, Harvard University, fischerf@seas.harvard.edu. Supported by the Deutsche Forschungsgemeinschaft under grant FI 1664/1-1.

§Microsoft Research Cambridge, iankash@microsoft.com

¶Computer Science Department, Carnegie Mellon University, arielpro@cs.cmu.edu

are compatible with their desired recipient. This raises the possibility of *kidney exchange*, in which two or more incompatible donor-patient pairs exchange kidneys such that each patient receives a compatible kidney from the donor of another patient.[1]

Incentives of donor-patient pairs and efficiency in kidney exchange programs have respectively been studied by Roth et al. (2005, 2004) and Roth et al. (2007). As kidney exchange programs are growing, however, manipulation by hospitals also becomes an issue: in particular, a hospital may choose to withhold some of its incompatible donor-patient pairs and match them internally, in order to maximize the number of its own patients who receive a kidney. This kind of strategic behavior has a negative effect on social welfare and runs counter to the whole idea of having a large exchange. It is therefore an interesting question how hospitals can be incentivized to fully participate in an exchange by submitting all of their incompatible donor-patient pairs.

This problem can be modeled formally as a matching problem on a graph in which each vertex corresponds to an incompatible donor-patient pair and an edge between two such pairs indices that the donor of each pair is compatible with the recipient of the respective other pair. Moreover disjoint sets of vertices are controlled by self-interested agents, in the sense that their existence is private information of the agent controlling them. Agents then reveal subsets of their vertices, and matches are determined based on the induced subgraph. An agent can seek to manipulate by hiding some of its vertices and then proceeding to benefit both from the inter-agent matches and matches on its hidden and unmatched vertices. We assume that each agent seeks to maximize the number of its own vertices that end up being matched.[2]

The above model was first used by Sönmez and Ünver and Ashlagi and Roth (2011) in order to study the incentive of the hospitals in an exchange. Sönmez and Ünver observed that no efficient and strategyproof mechanisms exist for this problem. Ashlagi and Roth showed that no deterministic strategyproof mechanism can guarantee more than half the size of an efficient matching, whereas nearly efficient incentive compatible mechanism exists in a Bayesian setting. A more detailed discussion of these results can be found in Section 2.

In this paper we take a prior-free approach to the nonexistence of efficient and strat-

[1]These cyclic exchanges can also be combined with chains, starting with a deceased donor or an "undirected" donor without a particular intended recipient and ending with a patient who has a high priority on the deceased-donor waiting list or with a donor who will donate at some point in the future.

[2]This model more generally applies to settings where information about clients and potential trades among clients is partitioned among a set of agents. What distinguishes kidney exchanges from other such settings is the absence of monetary transfers: in most countries, payments in return for organs are both illegal and considered immoral, so we are interested in mechanisms without payments.

egyproof mechanisms and relax efficiency rather than strategyproofness. We say that a mechanism is an α-*approximation* mechanism if the size of the maximum cardinality matching is always at most α times that of the matching returned by the mechanism.[3] Our goal is to design mechanisms that are strategyproof and at the same time provide a good approximation ratio. This approach interesting for at least two reasons. First, strategyproof mechanisms are more robust in the worst case against information hospitals might have about each others' patients. Interestingly, we will see that their efficiency loss *in practice* is still very small, much smaller than in the worst case. Second, together with the results of Ashlagi and Roth (2011), our results provide insights into the tradeoff between different degrees of incentive compatibility on the one hand and social welfare on the other.

We begin in Section 4 by establishing lower bounds on the approximation ratio of strategyproof mechanisms. To this end, we refine an example used by Sönmez and Ünver to illustrate that no efficient mechanism can strategyproof and observe that no deterministic strategyproof mechanism can provide an approximation ratio better than 2[4], and no randomized strategyproof mechanism can provide an approximation ratio better than 8/7.[5]

In Section 5 we then introduce a mechanism, termed $\textsc{Match}_\Pi$, that is parameterized by a bipartition $\Pi = (\Pi_1, \Pi_2)$ of the agents. Roughly speaking, for any given graph, the mechanism returns a matching that has maximum cardinality among all the matchings that (i) contain no edges between the vertex sets of two agents on the same side of the bipartition, and (ii) are a maximum cardinality matching when restricted to the vertex set of each individual agent. We show that $\textsc{Match}_\Pi$ is strategyproof for any set bipartition of the players and can be executed in polynomial time. Unfortunately, for any fixed bipartition Π, $\textsc{Match}_\Pi$ does not generally provide a bounded approximation ratio. We observe, however, that $\textsc{Match}_\Pi$ yields a 2-approximation in the two-agent case when used with the obvious bipartition that places the two agents on opposite sides. This mechanism is in fact the optimal deterministic strategyproof mechanism for two agents, since the deterministic lower bound of 2 holds even in this case.

In Section 6 we finally construct a randomized mechanism, termed Mix-and-Match, that first *mixes* the agents by choosing a random bipartition Π, then *matches* the vertices by applying $\textsc{Match}_\Pi$. We show that Mix-and-Match is strategyproof and provides a 2-approximation.

[3]Since the social welfare of a matching is exactly twice its cardinality, approximating the two is equivalent.

[4]Ashlagi and Roth (2011) show this result in a slightly different setting.

[5]The preliminary version of this paper incorrectly stated the bound as 4/3 (Ashlagi et al., 2010).

2 Related Work

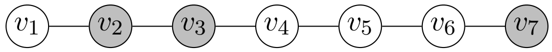
Most closely related to our work is that of Sönmez and Ünver and Ashlagi and Roth (2011), who consider mechanisms for multi-hospital kidney exchange that are individually rational. Individual rationality in our model requires that for each agent, the number of vertices matched by the mechanism is at least the number of vertices that the agent can match on its own. This is a weaker requirement than strategyproofness. Ashlagi and Roth (2011) show that for a prior distribution consistent with the real world, there exists an ϵ-Bayesian incentive compatible and individually rational mechanism that is almost efficient. Toulis and Parkes (2011) perform a similar kind of analysis of a different mechanism and under slightly different assumptions.[6]

Finding an efficient matching using exchanges involving more than two parties is computationally hard, and we restrict our attention to two-way exchanges in this paper. There are, however, algorithms that allow exchanges with multiple parties and have good performance in practice (Abraham et al., 2007; Biró et al., 2009). It is an interesting question for future work whether these algorithms can be made incentive compatible.

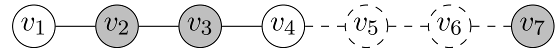
Our work is also closely related to a line of research that seeks to approximate "optimal" outcomes in mechanism design settings without monetary transfers. This line was initiated by Procaccia and Tennenholtz (2009), but dates back at least to work by Dekel et al. (2010) on truthful learning. The point of departure is the large body of work on algorithmic mechanism design, the study of truthful approximation mechanisms for game-theoretic versions of optimization problems (see, e.g., Nisan and Ronen, 2001; Lehmann et al., 2002; Lavi and Swami, 2005; Christodoulo et al., 2009). While mechanisms in this area typically use payments to align agents' incentives, Procaccia and Tennenholtz argue that monetary transfers are infeasible in many settings due to ethical, legal, or practical considerations. To achieve strategyproofness in such settings, it might be desirable to relax the requirement from exact to approximate efficiency. This approach is particularly intriguing in the context of problems that are computationally feasible: while there is no need to resort to approximate solutions for computational reasons, they might be used to achieve strategyproofness when the optimal solution is not strategyproof.

This approach was taken for example by Dughmi and Ghosh (2010), who studied approximate mechanisms without money in the context of the generalized assignment problem. An

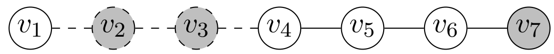
[6] Ashlagi and Roth and Toulis and Parkes use realistic values for parameters like the structure and frequency of blood types. While our theoretical results hold in the worst case and do not require any assumptions like this, we also study the performance of our mechanism on inputs drawn from the distribution used by Ashlagi and Roth.



(a) Graph G. Vertices of V_1 are shown in white, vertices of V_2 in gray.



(b) Graph G' obtained from G when agent 1 hides the vertices v_5 and v_6. The hidden vertices are not part of the graph, but are shown to give the complete picture.



(c) Graph G'' obtained from G when agent 2 hides vertices v_2 and v_3.

Figure 1: Construction used in the proof of Theorem 4.1

instance of this problem is given by a bipartite graph with jobs on one side and machines on the other, as well as a capacity for each machine and a value and a size for each edge. The agents in this setting are the jobs (corresponding to vertices rather than agents in our case), who hold their incident edges as private information. Dughmi and Ghosh in fact briefly look at maximum matchings as a special case of their model, but their motivation, setting, and results are all fundamentally different from ours. In particular, in the context of maximum unweighted matching, their model easily admits a strategyproof *optimal* mechanism, whereas this problem is quite intricate in our case and forms the core of this paper.

3 Preliminaries

Let $N = \{1, \ldots, n\}$ be a set of agents. For each $i \in N$, let V_i be a set of private vertices of agent i. Let $G = (V, E)$ with $V = \bigcup_{i \in N} V_i$ be an undirected labeled graph, that is, each vertex is labeled by its agent. We slightly abuse terminology by simply referring to such labeled graphs as "graphs."

A *matching* $M \subseteq E$ on G is a subset of edges such that each vertex is incident to at most one edge of M. For $i, j \in N$ we denote

$$M_{ij} = \{(u, v) \in M : \; u \in V_i \;\wedge\; v \in V_j\}.$$

Given $i \in N$, we refer to edges in M_{ii} as *internal edges* and to edges in M_{ij}, where $j \in N \setminus \{i\}$, as *external edges*.

Given a graph G and a matching M on G, the utility of agent i for this matching is

$$u_i(M) = |\{u \in V_i : \; \exists v \in V \text{ s.t. } (u, v) \in M\}|,$$

that is, it is equal to the number of vertices of V_i that are matched under M.

We now turn to the definition of a mechanism, without being too formal. For a fixed number n of agents, a *deterministic mechanism* is a function that maps any (labeled) graph for n agents to a matchings of this graph. A *randomized mechanism* maps any graph to a probability distribution over matchings, that is, it can select a matching randomly. For conciseness, we treat deterministic mechanisms as a special case of randomized mechanisms in the rest of this section.

For a randomized mechanism f and a (possibly random) graph G, define

$$u_i(f(G)) = \mathbb{E}_{M \sim f(G)}[u_i(M)],$$

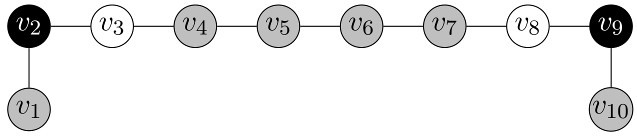
where the expectation is taken over the distribution on matchings returned by the mechanism. In other words, the utility of an agent simply equals the expected number of its vertices being matched.

We are concerned with situations where an agent "hides" a subset of its vertices and then internally matches them among themselves or with vertices not matched by the mechanism. To make this formal we need some notation. We however feel that the idea is rather intuitive, and will avoid the rather cumbersome formalism in the rest of the paper. For any subset $V' \subseteq V$, let $G[V']$ be the subgraph of G induced by V'. For a graph G, an agent $i \in N$, and a matching M, let $X_i(M)$ be the set of vertices in V_i that are not matched in M; if M is chosen randomly, then $X_i(M)$ is a random variable. Furthermore, let f^* be a mechanism that maps each graph G to a maximum cardinality matching of G. We say that a mechanism f is *strategyproof* if for every graph $G = (V, E)$ with $V = \bigcup_{i \in N} V_i$, for every $i \in N$, and for every $V'_i \subseteq V_i$ it holds that

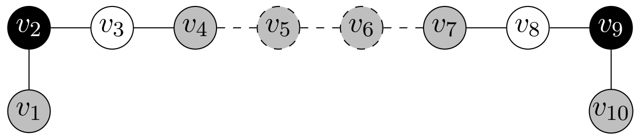
$$u_i(f(G)) \geq u_i(f(G[V \setminus V'_i])) + u_i(f^*(G[V'_i \cup X_i(f(G[V \setminus V'_i]))])).$$

In other words, a mechanism is strategyproof if an agent can never benefit by hiding some of his vertices. The agent's utility after hiding a subset V'_i of its vertices equals the (expected) number of its vertices that the mechanism matches given the subgraph induced by all vertices but those in V'_i, plus the (expected) number of vertices in a maximum cardinality matching of the subgraph induced by V'_i and the vertices not matched by the mechanism. In our model, individual rationality (IR) requires that an agent cannot benefit from the special case when $V'_i = V_i$, and is therefore implied by strategyproofness.

We are interested in mechanisms that, while being strategyproof, produce matchings that maximize social welfare, i.e., the sum of agent utilities. For any matching M, $\sum_{i \in N} u_i(M) = 2|M|$, so what we are looking for are matchings that are as large as possible. We say that a



(a) The original graph G, where the vertices of V_1 are white, the vertices of V_2 are gray, and the vertices of V_3 are black



(b) The graph G', agent 2 hides vertices v_5 and v_6

Figure 2: The naïve 3-agent mechanism is not strategyproof.

randomized mechanism f provides an α-approximation if for every graph G,

$$\frac{|f^*(G)|}{\mathbb{E}[|f(G)|]} \leq \alpha, \tag{1}$$

where once again $f^*(G)$ is a maximum cardinality matching of G. For deterministic mechanisms, the expectation in (1) can simply be dropped.

4 Lower Bounds

It may not be immediately apparent that the optimal mechanism is not strategyproof. Given a graph, the optimal mechanism simply returns a maximum cardinality matching (while employing a consistent tie-breaking rule to decide between different maximum cardinality matchings).

To see how this can fail to be strategyproof, consider the graph G in Figure 1(a). This graph has an odd number of vertices, so every matching leaves some vertex unmatched. However, each agent has a pair of vertices such that removing these vertices from the graph results in a graph with a unique maximum cardinality matching in which all of that agent's vertices are matched (Figures 1(b) and 1(c)). Thus, one of the agents must have an unmatched vertex in G, and this agent can hide two of his vertices to increase his utility. This simple example, which is due to Sönmez and Ünver (see also Ashlagi and Roth (2011)), can be used to derive lower bounds that will later turn out to be, at least in one case, tight.

Theorem 4.1. *If there are at least two agents,*

1. *no deterministic strategyproof mechanism can provide an α-approximation with respect to social welfare for $\alpha < 2$, and*

2. *no randomized strategyproof mechanism can provide an α-approximation with respect to social welfare for $\alpha < 8/7$.*

Proof. For the first part of the theorem, we consider the case where $N = \{1, 2\}$; the proof can easily be extended to the case where $n > 2$ by adding agents with vertices that are not incident to any edges. Let f be a deterministic mechanism, and consider the graph G given in Figure 1(a). Since G has an odd number of vertices, it does not have a perfect matching, so $f(G)$ must leave some $v \in V_1$ or some $v \in V_2$ unmatched. Thus, either $u_1(f(G)) \leq 3$ or $u_2(f(G)) \leq 2$.

We first deal with the case where $u_1(f(G)) \leq 3$. Consider the graph G' that is obtained when agent 1 hides vertices v_5 and v_6 (see Figure 1(b)). The unique maximum cardinality matching of this graph is $\{(v_1, v_2), (v_3, v_4)\}$, a matching of cardinality 2. However, agent 1 could internally match the pair (v_5, v_6) and obtain a utility of 4, contradicting strategyproofness. Therefore, $f(G')$ must have cardinality at most 1, meaning that its approximation ratio on G' cannot be smaller than 2.

The case where $u_2(f(G)) \leq 2$ can be handled similarly. Consider the graph G'' obtained when agent 2 hides vertices v_2 and v_3 (see Figure 1(c)). Once again there is a unique maximum matching of cardinality 2, but f cannot return this matching since it would yield a utility of 3 to agent 2, in contradiction to strategyproofness. As before the mechanism is forced to select a matching of cardinality at most 1.

The second part of the theorem can be derived using the same construction. Let f be a randomized strategyproof mechanism. Since G does not have a perfect matching, it must be the case that $f(G)$ either does not match some vertex of V_1 with probability at least 1/2, or it does not match some vertex of V_2 with probability at least 1/2, that is, either $u_1(f(G)) \leq 7/2$ or $u_2(f(G)) \leq 5/2$.

We now proceed as before. If $u_1(f(G)) \leq 7/2$, we consider the graph G'; by strategyproofness f can only match both of agent 1's pairs with probability at most 3/4, for a maximum of 7/4 pairs in expectation, but the optimum is 2. If $u_2(f(G)) \leq 5/2$, we use the graph G'' to show that f can only match 7/4 pairs in expectation, while the optimum is 2. □

5 Deterministic Mechanisms

Let us now consider deterministic mechanisms. We begin by designing a deterministic mechanism that is strategyproof for any number of agents, but may not provide a bounded approximation ratio. We then leverage this mechanism to obtain an optimal deterministic strategyproof mechanism for two agents. The more powerful application of our deterministic mechanism will only appear in the next section, when we discuss randomized mechanisms.

Let us first address the issue of designing strategyproof deterministic mechanisms without worrying, for now, about approximate efficiency or computational tractability. Consider the following mechanism for two agents. Given a graph G, the mechanism computes the set of all matchings on G that have maximum cardinality on V_1 and V_2, and among these selects a matching with maximum overall cardinality. Since every matching that this mechanism considers has maximum cardinality on V_1 and V_2, it clearly is individually rational. We will show momentarily that it is also strategyproof.

But let us first consider what this mechanism does when applied to the graph of Figure 1(a). Any matching that is a maximum cardinality matching on V_2 would have to match (v_2, v_3), and there are two maximum cardinality matchings on V_1: one can either match (v_4, v_5) or (v_5, v_6). If we match (v_5, v_6), no additional edges can be added. Hence, the unique matching of cardinality 3 that maximizes the number of internal edges is $\{(v_2, v_3), (v_4, v_5), (v_6, v_7)\}$. The only unmatched vertex in this matching is v_1. With the proof of Theorem 4.1 in mind, let us verify that agent 1 cannot benefit by hiding v_5 and v_6. Given the graph G' in Figure 1(b), the mechanism would simply return the matching (v_2, v_3), since this is the unique matching that is a maximum cardinality matching on V_2.

The two-agent mechanism suggested above seems promising from the perspective of strategyproofness. Let us extend it to an n-agent mechanism in the natural way, and consider the mechanism that selects a matching of maximum cardinality among the matchings that have maximum cardinality on each V_i, $i = 1, \ldots, n$. In addition, let us break ties *serially*: among all the matchings that meet the above criteria, we select a matching that maximizes the utility of agent 1; if there are several such matchings, we choose one that maximizes the utility of agent 2, and so on.

Interestingly enough, this n-agent mechanism is not strategyproof, even when $n = 3$. Consider the graph G given in Figure 2(a). Any matching that has maximum cardinality on V_2 must match (v_4, v_5) and (v_6, v_7); by the tie-breaking rule the mechanism then returns the matching $\{(v_2, v_3), (v_4, v_5), (v_6, v_7), (v_8, v_9)\}$. When agent 2 hides v_5 and v_6 we obtain the graph G' given in Figure 2(b). On this graph the mechanism returns a perfect matching $\{(v_1, v_2), (v_3, v_4), (v_7, v_8), (v_9, v_{10})\}$. After internally matching (v_5, v_6) agent 1 gains two

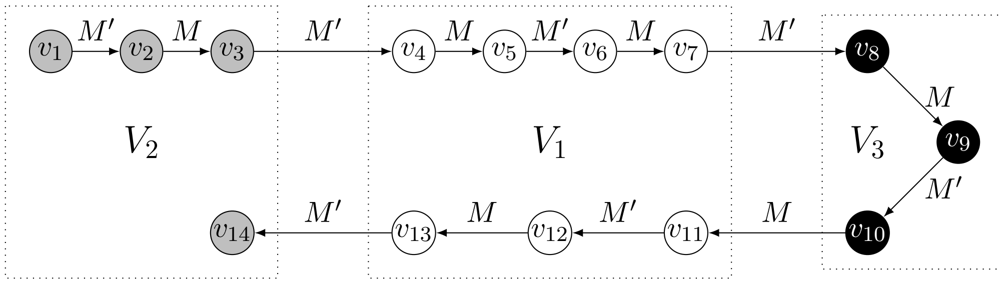


Figure 3: Illustration of Case 1 of the proof of Theorem 5.1, with $i = 1$ as the manipulator, and $\Pi = (\{1\}, \{2, 3\})$. $M \Delta M'$ is shown as a single directed path with alternating edges of M and M'. It holds that $3 = |M_{11}| > |M'_{11}| = 2$. Every subpath inside V_2 and V_3 has even length (those from v_1 to v_3 and from v_8 to v_{10}), but subpaths inside V_1 may not have (like that from v_4 and v_7). The subpath of $M \Delta M' \setminus (M_{11} \cup M'_{11})$ from v_1 to v_4 enters V_1 but does not exit it, while the subpath from v_{13} to v_{14} exits V_1 but does not enter it. This example satisfies (2) with equality.

additional matched vertices compared to the matching on G. Clearly this example can be modified to work if ties are broken in a different order.

The deeper reason why the above mechanism fails to be strategyproof is rather subtle, and has to do with the following observation. If one takes the union of the matchings generated on the graphs of Figures 2(a) and 2(b), and contracts each V_i to one vertex, one obtains an odd length cycle between V_1, V_2, and V_3, as the matching on G has an edge between V_1 and V_3, and the matching on G' has edges between V_1 and V_2, and V_2 and V_3. We proceed to refine the above mechanism in order to avoid such odd cycles; this turns out to be sufficient to guarantee strategyproofness. The following is in fact a family of mechanisms, parameterized by a fixed bipartition $\Pi = (\Pi_1, \Pi_2)$ of the set of agents.

$\textsc{Match}_\Pi$

1. Given a graph G, consider all the matchings that have maximum cardinality on each V_i and do not have any edges between V_i and V_j when $i, j \in \Pi_l$ for some $l \in \{1, 2\}$, i.e., those that maximize the number of internal edges and do not have any edges between sets on the same side of the bipartition.

2. Among these matchings select one of maximum cardinality, breaking ties serially in favor of agents in Π_1 and then agents in Π_2.

By letting $N = \{1, 2\}$, $\Pi_1 = \{1\}$, and $\Pi_2 = \{2\}$, we obtain the two-agent mechanism described above. The naïve generalization of this mechanism to three agents, on the other hand, is not an instance of $\textsc{Match}_\Pi$: for the example of Figure 2 showing that the mechanism is not strategyproof, the sets M_{12}, M_{13}, and M_{23} are all non-empty.

We proceed to show that $\textsc{Match}_\Pi$ is strategyproof for any bipartition of the set of agents. The main idea behind the proof of this theorem is rather subtle. It relies on the fact that if one takes the union of the two matchings produced by the mechanism before and after an agent hides some of its vertices, then this union cannot contains a cycle that visits the vertex sets of an odd number of agents. This property holds because the mechanism does not match edges between vertex sets of agents on the same side of the bipartition.

Theorem 5.1. *For any number of agents, and for any bipartition* Π *of the set of agents,* $\textsc{Match}_\Pi$ *is strategyproof.*

Proof. Fix some bipartition $\Pi = (\Pi_1, \Pi_2)$ of N. Consider a graph G, and let $M = \textsc{Match}_\Pi(G)$. Assume that agent $i \in N$ hides a subset of vertices, inducing a subgraph G', and let M' be the matching that results from applying the mechanism to G', along with the internal matching of agent 1 on its hidden and unmatched vertices, that is,

$$M' = \textsc{Match}_\Pi(G') \cup \hat{M},$$

where $\hat{M}$ is a maximum cardinality matching of agent i on its hidden and unmatched vertices.

The symmetric difference

$$M \Delta M' = M \cup M' \setminus (M \cap M')$$

then consists of vertex-disjoint paths (some of which may be cycles) with alternating edges of M and M'. For example, consider the two-agent version of $\textsc{Match}_\Pi$ applied to the graphs G and G' given in Figures 1(a) and Figure 1(b). It holds that

$$M = \textsc{Match}_\Pi(G) = \{(v_2, v_3), (v_4, v_5), (v_6, v_7)\},$$

whereas, say, $M' = \{(v_2, v_3), (v_5, v_6)\}$. Then, $M \Delta M'$ is the single path $\{(v_4, v_5), (v_5, v_6), (v_6, v_7)\}$ where the first and last edge are in M and the middle edge is in M'.

In order to simplify notation, we henceforth assume that $M \Delta M'$ consists of just one path. This assumption is made without loss of generality, because we show that each such path satisfies one of the following properties: either M matches at least as many vertices of V_i as M' for every $i \in N$, or one can derive a contradiction to the way M or M' were selected by switching between some (or all) of their edges on the path. Since the contradiction can be derived for each path separately, it follows that the first property holds on every path, that is, the overall utility of agent i for M is at least as large as its utility for M'.

If the path in $M\Delta M'$ is a cycle, then this cycle must be of even length, because otherwise there would be a vertex that is incident to two edges of the same matching. It follows that both M and M' match all the vertices on the cycle, hence agent i is indifferent between the two matchings. We may therefore assume that $M\Delta M'$ is not a cycle.

It will prove useful to arbitrarily fix a direction over the (undirected) edges of the single path in $M\Delta M'$. Since the path is not a cycle, this direction pinpoints two specific vertices as the start and the end of the path. We further say that the (directed) edge (u,v) *enters* V_j if $u \notin V_j$ and $v \in V_j$, and *exits* V_j if $u \in V_j$ and $v \notin V_j$.

We consider two cases.

Case 1: $|M_{ii}| > |M'_{ii}|$. We claim that

$$\sum_{j\in N\setminus\{i\}} |M_{ij}| \geq \sum_{j\in N\setminus\{i\}} |M'_{ij}| - 2. \tag{2}$$

Since both M and M' are maximum cardinality matchings on V_j for all $j \neq i$, it must hold that every subpath of $M\Delta M'$ on V_j has even length (see Figure 3); otherwise we would have, say, more edges of M than M' on the subpath, and by switching from M' to M on the subpath we would be able to increase the size of M' on V_j. This implies that for any $j \in N \setminus \{i\}$, any subpath entering V_j with an edge of M' must exit V_j with an edge of M, and any subpath entering V_j with and edge of M must exit V_j with an edge of M'.

The next part of the proof is crucial, and uses the main idea behind mechanism $\textsc{Match}_\Pi$. We argue that it also holds that a subpath exiting V_i with an edge of M' can only enter V_i with an edge of M. Assume without loss of generality that $i \in \Pi_1$. By the above argument the subpath enters V_{j_1}, $j_1 \in \Pi_2$, with an edge of M', and therefore exits it with an edge of M, entering some V_{j_2} in Π_1. If $j_2 \neq i$, and the subpath exits V_{j_2}, then it does so with an edge of M', and by the same arguments returns to the vertex set of an agent in Π_1 with an edge of M. If eventually the subpath enters V_i again, it must be with an edge of M. Analogously, if the subpath exits V_i with an edge of M, it can only enter V_i with an edge of M'. See Figure 3 for an illustration.

Now consider $(M\Delta M')\setminus(M_{ii}\cup M'_{ii})$, which again is a collection of vertex-disjoint subpaths. Some start and end in V_i, and it follows by the discussion above that such subpaths have exactly one edge in M_{ij} and one edge in M'_{ik}, for $k, j \in N \setminus \{i\}$. There can only be one subpath that starts in V_i but does not end in V_i, and at most one subpath that ends in V_i but does not start in V_i. Equation (2) directly follows.

We now have that

$$u_i(M) = 2|M_{ii}| + \sum_{j \in N\setminus\{i\}} |M_{ij}| \geq 2(|M'_{ii}| + 1) + \left(\sum_{j \in N\setminus\{i\}} |M'_{ij}| - 2 \right) = u_i(M'),$$

where the inequality follows from the fact that $|M_{ii}| > |M'_{ii}|$ and from (2).

Case 2: $|M'_{ii}| = |M_{ii}|$. Note that it holds that $|M_{jj}| = |M'_{jj}|$ for all $j \in N$, that is, $M\Delta M'$ has to be of even length inside every V_j. This includes M_{ii} and M'_{ii}, because the total number of internal edges for i is even. If some subpath of i's internal edges has odd length with more edges from M there must be another subpath with more internal edges from M'. Swapping the edges of M for those of M' in the second subpath results in a matching M'' such that $|M''_{ii}| > |M_{ii}|$, contradicting the construction of M to have maximum cardinality on each V_i. It follows that $|M| \geq |M'|$, since M is a maximum cardinality matching under the constraint that it has maximum cardinality inside each V_i.

We claim that if $|M| > |M'|$ then $\sum_j |M_{ij}| \geq \sum_j |M'_{ij}|$. Together with the assumption that $|M'_{ii}| = |M_{ii}|$ this implies that agent i cannot benefit. Indeed, in this case $M\Delta M'$ is a path of odd length that starts and ends with an edge of M. Recall that every subpath of $M\Delta M'$ consisting of i's internal edges has even length. This means that when the path enters V_i with an edge of M' it cannot end inside V_i, as otherwise it would end with an edge of M'. In other words, every time the path enters V_i with an edge of M' it must exit V_i with an edge of M. Similarly, every time the path exits V_i with an edge of M' it must have entered V_i with an edge of M, otherwise the path must start in V_i with an edge of M'. This proves our claim, so $|M| = |M'|$.

Since $|M| = |M'|$ we have that $M\Delta M'$ has even length, and moreover we know it has even length inside each V_i. Note that all the vertices on the path are matched under both M and M', except for the start and the end vertices. Hence, if agent i gains from the manipulation, it must be the case (when fixing a specific direction on the edges) that the start vertex is a vertex of V_i and the first edge is an edge of M', whereas the end vertex is in V_j, for some $j \in N \setminus \{i\}$, and the last edge is an edge of M.

Now, if tie-breaking favors i over j, then by switching the edges of M with those of M' we get a matching of equal size that has maximum cardinality on each V_i and is better for i, in contradiction to the tie-breaking rule. If tie-breaking favors j over i, consider the subpath of $M\Delta M'$ that starts with the last edge that exits V_i and ends with the last edge in $M\Delta M'$. This path must start with an edge of M'. To see why, note that $M\Delta M'$ starts in V_i with an edge of M'. This subpath has even length, so it exits with an edge of M'. By the same argument as in Case 1, the bipartition ensures that, if the path re-enters V_i, it does so with an edge from M. Since all subpaths of vertices in V_i are of even length, the path always

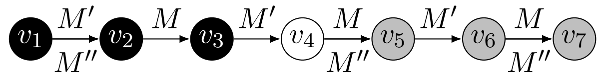


Figure 4: An illustration of the last argument in Case 2 of the proof of Theorem 5.1 with $i = 3$ and $j = 2$. The vertices of V_1 are white, the vertices of V_2 are gray, and the vertices of V_3 are black. By switching from M' to M'' we increase the utility of agent 2 and decrease the utility of agent 3, thereby obtaining a legal matching that contradicts the choice of M'.

exits V_i with an edge of M'.

By replacing all the edges of M' with the edges of M on this subpath, we can obtain a matching M'' that is identical to M' inside V_i, has maximum cardinality on V_k for each $k \in N$, is as large as M' overall, and satisfies $u_j(M'') = u_j(M') + 1$, $u_i(M') = u_i(M'') - 1$, and $u_k(M'') = u_k(M')$ for all $k \in N \setminus \{i, j\}$. By removing the edges of $\hat{M}$ (recall that this is the second stage internal matching of i) from both M' and M'' we get a contradiction to the way the mechanism broke ties when constructing M' (specifically, when constructing $\textsc{Match}_\Pi(G')$). See Figure 4 for an illustration. □

We next show that $\textsc{Match}_\Pi$ can be executed in polynomial time by a reduction to the maximum weighted matching problem (for a polynomial time algorithm for the latter see Gabow (1990)).

Theorem 5.2. $\textsc{Match}_\Pi$ *can be executed in polynomial time.*

Proof. Assume without loss of generality that $|E| > 1$, and let $\epsilon_i = 1/|E|^{i+1}$. We assign weights to edges as follows. An (internal) edge (u, v) such that $u, v \in V_i$ for some $i \in N$ receives weight $|E| + 3$. An (external) edge (u, v) such that $u \in V_i$ and $v \in V_j$ with $i \in \Pi_1$ and $j \in \Pi_2$ receives weight $1 + \epsilon_i + \epsilon_j/|E|^{n+1}$. An (external) edge (u, v) such that $u \in V_i$ and $v \in V_j$ with $i \neq j$ but $i, j \in \Pi_1$ or $i, j \in \Pi_2$ receives weight 0.

The sum of the weights of all external edges is at most $|E|(1+1/|E|^2+1/|E|^{n+3}) < |E|+3$, which is less than the weight of a single internal edge. Thus a maximum weight matching of this graph maximizes the number of internal edges. All edges between sets on the same side of the bipartition have weight zero, so no such edges will be included.

To complete the proof we need to verify that the maximum weight matching has maximum cardinality among those with a maximum number of internal edges and no edges across the bipartition, and that ties are broken appropriately. Each edge across the bipartition has weight at least 1 and at most $1 + 1/|E|^2 + 1/|E|^{n+3}$. Thus, given two matchings M and M' satisfying the above constraints such that $|M| > |M'|$, the difference in their weights is at least

$$
\begin{aligned}
1 - |M'|(1/|E|^2 + 1/|E|^{n+3}) &\geq 1 - |E|(1/|E|^2 + 1/|E|^{n+3}) \\
&= 1 - 1/|E| - 1/|E|^{n+2} > 0.
\end{aligned}
$$

The maximum weight matching thus has maximum cardinality subject to the constraints. For tie-breaking, observe that $\epsilon_i \geq |E|\epsilon_j$ if $i < j$, meaning that among agents on the same side of the bipartition those with smaller indices have higher priority. The factor of $1/|E|^{n+1}$ finally ensures that agents in Π_1 have priority over agents in Π_2. □

Recall that by Theorem 4.1 no deterministic strategyproof mechanism can have an approximation ratio smaller than 2, even when there are only two agents. We will see momentarily that Match_Π provides an approximation ratio of 2 when $N = \{1, 2\}$ and $\Pi = (\{1\}, \{2\})$, i.e., it is the best possible deterministic strategyproof mechanism for the case of two agents. Indeed, consider a graph G, let M^* be an optimal matching of G, and M the matching returned by $\text{Match}_{(\{1\},\{2\})}$. M is inclusion-maximal. Therefore, for every $(u, v) \in M^*$, either u is matched by M or v is matched by M. We conclude that $|M| \geq |M^*|/2$. Strategyproofness is obtained from Theorem 5.1.

Corollary 5.3. *Let $N = \{1, 2\}$. Then, $\text{Match}_{(\{1\},\{2\})}$ is strategyproof and provides a 2-approximation with respect to social welfare.*

Unfortunately, when $n \geq 3$, Match_Π does not provide a finite approximation ratio for any fixed bipartition. To see this, let $\Pi = (\Pi_1, \Pi_2)$ be a bipartition of the set of agents. Then there must be two distinct agents $i, j \in N$ such that $i, j \in \Pi_l$ for some $l \in \{1, 2\}$. Now consider a graph where the only edge is an external edge between V_i and V_j; given this graph Match_Π returns an empty matching, whereas the optimum is a matching of cardinality 1.

We believe that in general deterministic strategyproof mechanisms can only provide a bad approximation ratio, even for the case of three agents. The following conjecture makes this precise.

Conjecture 5.4. *If there are more than two agents, no deterministic strategyproof mechanism can provide an α-approximation with respect to social welfare for any constant α.*

6 Randomized Mechanisms

We have seen above that Match_Π does not provide a bounded approximation ratio for any fixed bipartition Π. The natural next step is to choose the bipartition uniformly at random. This leads to the eponymous Mix-and-Match mechanism.

Mix-and-Match

1. **Mix:** Construct a random bipartition $\Pi = (\Pi_1, \Pi_2)$ of the agents by independently flipping a fair coin for each agent to determine whether the agent is in Π_1 or in Π_2.

2. **Match:** Apply Match_Π to the given graph, where Π is the bipartition constructed in Step 1.

It immediately follows from Theorem 5.1 that Mix-and-Match is strategyproof, and in fact in a stronger sense than the one defined in Section 3, namely *universal strategyproofness*. A randomized mechanism is called universally strategyproof if agents cannot gain by lying regardless of the random choices made by the mechanism, i.e., if the mechanism is a distribution over strategyproof deterministic mechanisms.

A naïve analysis of Mix-and-Match would yield a rather unimpressive approximation ratio. Indeed, the reason why $\text{Match}_{(\{1\},\{2\})}$ does not provide a better approximation ratio than two is that it may have to sacrifice two external edges for one internal edge. The fact that Mix-and-Match will not be able to match many of the edges in the graph because they are not between the two elements of the constructed bipartition would seem to cause the approximation ratio to deteriorate further. Fortunately, these two problems effectively cancel out: sacrificing two external edges for an internal edge is less of a problem when each of those external edges is allowed to be part of the matching for only half of the bipartitions. Formally, we prove the following result.

Theorem 6.1. *For any number of agents,* Mix-and-Match *is (universally) strategyproof and provides a* 2*-approximation with respect to social welfare.*

Proof. We prove the theorem by taking a maximum cardinality matching M^* and constructing a matching M' that, when restricted according to a random bipartition (by removing edges between agents on the same side of the bipartition), has at least half the size of M^* in expectation. The matching produced by Match_Π then always is at least as large as M' restricted according to Π.

Consider a graph G, and let M^* be a maximum cardinality matching of G. For each $i \in N$, let M_i^{**} be a maximum cardinality matching on V_i, and let $M^{**} = \bigcup_{i \in N} M_i^{**}$.

We construct a matching M' by considering the symmetric difference $M^* \Delta M^{**}$. As in the proof of Theorem 5.1, it consists of a set of paths with alternating edges of M^* and M^{**}. For each path, if there are more internal edges among the edges from M^{**}, we put those edges in M'. Otherwise, we put the edges from M^* in M'.

Since M^{**} has maximum cardinality on each V_i and M' has the same number of internal edges from each path as M^{**}, M' has maximum cardinality on each V_i. Furthermore, since

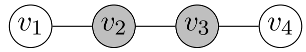


Figure 5: Graph illustrating that Mix-and-Match cannot provide an approximation ration smaller than two. V_1 is shown in white, V_2 is shown in gray. Mix-and-Match returns the matching (v_2, v_3).

M^* is a maximum cardinality matching, each path has either the same number of edges from M^* and M^{**} or one extra edge from M^*. All external edges on the path are from M^*, so if the edges from M^{**} are taken for M' then the number of internal edges gained relative to M^* is at least the number of external edges lost minus one. In the worst case M' has two fewer external edges for each extra internal edge relative to M^*. Thus M' satisfies

$$\sum_{i \in N}(|M'_{ii}| - |M^*_{ii}|) \geq \frac{1}{2}\sum_{i \in N}\sum_{j > i}(|M^*_{ij}| - |M'_{ij}|),$$

where we sum over $j > i$ so as not to count the same edges twice. Rearranging, we get

$$\sum_{i \in N}|M'_{ii}| + \frac{1}{2}\sum_{i \in N}\sum_{j > i}|M'_{ij}| \geq \sum_{i \in N}|M^*_{ii}| + \frac{1}{2}\sum_{i \in N}\sum_{j > i}|M^*_{ij}|. \tag{3}$$

Now let M^Π be the matching produced by Match$_\Pi$ for the fixed bipartition Π. Since M^Π has maximum cardinality under the constraints, we have

$$|M^\Pi| = \sum_{i \in N}|M^\Pi_{ii}| + \sum_{i \in \Pi_1}\sum_{j \in \Pi_2}|M^\Pi_{ij}| \geq \sum_{i \in N}|M'_{ii}| + \sum_{i \in \Pi_1}\sum_{j \in \Pi_2}|M'_{ij}|.$$

Since each pair of agents appears on opposite sides in exactly half of the bipartitions, the expected size of the matching produced by Mix-and-Match is

$$\begin{aligned}\sum_{\Pi}\left(\frac{1}{2^n}\cdot|M^\Pi|\right) &\geq \sum_{i \in N}|M'_{ii}| + \frac{1}{2}\sum_{i \in N}\sum_{j > i}|M'_{ij}| \\ &\geq \sum_{i \in N}|M^*_{ii}| + \frac{1}{2}\sum_{i \in N}\sum_{j > i}|M^*_{ij}| \geq \frac{1}{2}\cdot|M^*|,\end{aligned}$$

where the second inequality follows from (3). □

The graph in Figure 5 shows that the analysis of Mix-and-Match is tight even for $n = 2$. Still one might hope to do better, given that Theorem 4.1 only provides a randomized lower bound of 8/7, and indeed recently Caragiannis et al. (2011) were able to provide an upper bound of 3/2 for the case of $n = 2$ using the following mechanism.

Weight-and-Match

1. Given a graph G, assign a weight of 1 to internal edges and a weight of $1/2$ to external edges.

2. Flip a fair coin.

3. If the outcome is heads, return a maximum-cardinality matching among all maximum-weight matchings.

4. If the outcome is tails, return a minimum-cardinality matching among all maximum-weight matchings.

Despite the improvement over Mix-and-Match for the case of two agents, this mechanism still leaves a small gap between the randomized upper and lower bounds.

7 Average-Case Performance

Mix-and-Match achieves only half of the maximum social welfare in the worst case, and in fact no strategyproof mechanism can do much better. It is therefore natural to ask how Mix-and-Match performs in practice, where the occurrence of a worst-case instance might be very unlikely. We therefore simulate the practical performance of Mix-and-Match using incompatible donor-patient pairs drawn at random according to realistic parameters, and compare it to the optimal outcome without manipulation and to the outcome obtained when agents match their vertices internally and reveal only their unmatched vertices.

We begin by generating patients and donors until we reach a desired number k of incompatible pairs (currently compatible pairs simply have the operation performed and do not participate in the match). Each patient is assigned a blood type and a number $[0, 1]$ representing the likeliness of a tissue-type incompatibility with a random donor. Both are drawn from realistic distributions: we use probabilities of 48%, 34%, 14%, and 4% for blood types 0, A, B, and AB; the probability for tissue-type incompatibility is set to .05 with probability 70%, to .45 with probability 20%, and to .9 with probability 10% (cf. Roth et al., 2007). For each donor, we draw a blood type and a uniform random number in $[0, 1]$. If a patient and its corresponding donor have incompatible blood types or if the number of the donor is smaller than that of the patient (corresponding to a negative outcome of the so-called Panel Reactive Antibody test), they are added to the pool of incompatible pairs. Otherwise they are considered compatible and are discarded. Compatibility between any pair of vertices,

m	opt	mm	hide	mm/ opt	mm/ hide
2	10.00	9.84	9.84	0.98	1.00
4	26.81	25.91	25.80	0.97	1.00
6	45.64	43.66	43.16	0.96	1.01
10	86.34	82.17	80.40	0.95	1.02
15	141.72	134.00	130.45	0.95	1.03
20	196.78	185.86	179.96	0.94	1.03
25	252.86	237.42	229.18	0.94	1.04
30	311.38	290.79	280.38	0.93	1.04
40	430.04	397.20	381.28	0.92	1.04
50	548.68	503.38	483.27	0.92	1.04

Table 1: Performance of Mix-and-Match and the optimal mechanism in a situation with 20 agents and m vertices per agent, for different values of m

each corresponding to an incompatible pair thus drawn, is determined analogously, resulting in a random graph with k vertices. Finally, the vertices are independently assigned to one of n sets, each corresponding to an agent.

Table 1 shows the results for a setting with $n = 20$ and different numbers of vertices per agent, each of them averaged over 400 random instances. Columns "opt" and "mm" show the number of vertices in a maximum cardinality matching and the number of vertices matched by Mix-and-Match, respectively. Column "hide" shows the number of vertices matched when each agent matches vertices internally according to a random maximum cardinality matching and the remaining vertices are matched as well as possible. Columns "opt" and "hide" thus provide upper and lower bounds on the performance of the optimal mechanism. The last two columns illustrate the relative performance of Mix-and-Match compared to these bounds.

Table 2 shows the analogous results for $k = 300$. Note that the performance of Mix-and-Match now improves as n increases and k/n decreases. We also ran simulations for other values of k and n, with comparable results.

These results indicate that the practical performance of Mix-and-Match will often be much closer to the optimum in a setting without manipulation than suggested by the worst-case results, and will always be better than the optimum achievable under what is arguably

n	opt	mm	hide	mm/ opt	mm/ hide
5	141.54	133.25	127.21	0.94	1.05
10	139.70	131.17	125.92	0.94	1.04
15	140.05	132.04	127.59	0.94	1.03
20	140.29	132.90	129.22	0.95	1.03
25	140.54	133.29	129.99	0.95	1.03
30	139.88	133.06	130.46	0.95	1.02
35	139.67	133.51	131.18	0.96	1.02
40	140.26	134.34	132.35	0.96	1.01

Table 2: Performance of Mix-and-Match and the optimal mechanism in a situation with n agents, for different values of n, 300 vertices, and a variable number of vertices by agent

the most natural type of manipulation.[7] In addition to the margin it has over the optimal mechanism when agents manipulate, which sometimes is relatively small, Mix-and-Match also provides other advantages compared to mechanisms that are not strategyproof. These will be discussed briefly in the following section.

8 Discussion and Future Work

We have seen that Mix-and-Match provides near optimal worst-case guarantees: the outcome it achieves is always within a factor of two of the optimal two-way matching, which matches the lower bound for deterministic mechanisms and is close to the lower bound for randomized mechanisms. While a factor of two might not be acceptable in practice, in particular in the context of kidney exchanges, simulations suggest a practical performance that is much closer to optimal and better, if only by a relatively small amount, than that of mechanisms that incentivize agents to hide vertices and match them internally. More importantly, what distinguishes Mix-and-Match from mechanisms that are not strategyproof[8] is that it is robust against information asymmetries, has zero deliberation cost, and zero ex-post

[7]Observe that for the case of $n = 2$, Match_Π with $\Pi = \{\{1\}, \{2\}\}$ is in fact provably at least as good as the optimum achievable when manipulation takes place, because under Match_Π the maximum cardinality internal matchings are chosen to admit an optimal matching on the remaining vertices.

[8]This includes mechanisms that are not incentive compatible, but also mechanisms satisfying weaker notions of incentive compatibility like the one proposed by Ashlagi and Roth (2011).

regret. Arguably, all of these properties are important in the context of kidney exchanges.

An aspect of Mix-and-Match that might be problematic in practice is that it disallows matches between agents on the same side of the bipartition: it may be hard to convince hospitals that they best serve their patients by refusing to match them with patients from about half the other hospitals, despite the fact that this does not have a negative impact on the overall performance, neither in the worst case, nor on average assuming there are sufficiently many patients. One might therefore ask to what extent this property is necessary to guarantee strategyproofness and high welfare, or one could more generally try to characterize the set of strategyproof mechanisms. Our results suggest that there probably is no simple characterization. Quite a few straightforward mechanisms are instances of Match_{Π}: for example, only taking edges inside hospitals corresponds to a bipartition with all agents on the same side. On the other hand, a mechanism that selects two agents and runs the two-agent mechanism on them does not correspond to any bipartition.[9]

Several gaps still remain between our upper and lower bounds. The most enigmatic one concerns deterministic mechanisms when the number of agents is at least three. While Theorem 4.1 provides a deterministic lower bound of 2, we were unable to design a deterministic strategyproof mechanism with a constant approximation ratio, and indeed we conjecture that such a mechanism does not exist (Conjecture 5.4). For randomized mechanisms, there is a gap between the lower bound of 8/7 and the upper bound of 2 provided by Mix-and-Match. Our unfounded guess is that the randomized lower bound for more than two agents is 2. For the case of two agents, Caragiannis et al. (2011) recently provided an strategyproof 3/2-approximation mechanism, but it is still unclear whether this improved upper bound is tight.

There also exist a number of possible extensions to our work, of which we briefly point out a few. As we assume that agents wish to maximize the number of their clients being matched, a natural and realistic extension would be to incorporate weights into the model. For example, different exchanges involving the same vertex may be valued differently, or one vertex may be more important than the other. Another direction would be to allow exchanges of length greater than two; in this case one would have to consider directed graphs and look for sets of disjoint cycles that cover many vertices. This direction is important, as it has been shown that the number of exchanges could would increase substantially already through three-way exchanges. Finally, one could ask for the stronger requirement of group-strategyproofness, such that no group of agents would have an incentive to deviate in a

[9]While these examples are fairly close to Match_{Π}, we also know of a (relatively complex) mechanism that works quite differently.

coordinated fashion. A related approach would be to consider solution concepts like the core, to ensure that no group of agents would want to leave and form a smaller pool.

References

D. Abraham, A. Blum, and T. Sandholm. Clearing algorithms for barter exchange markets: Enabling nationwide kidney exchanges. In *Proceedings of the 8th ACM Conference on Electronic Commerce*, pages 295–304, 2007.

I. Ashlagi and A. E. Roth. Free riding and participation in large scale, multi-hospital kidney exchange. Working paper, 2011.

I. Ashlagi, F. Fischer, I. Kash, and A.D. Procaccia. Mix and match. In *Proceedings of the 11th ACM Conference on Electronic Commerce*, pages 295–304, 2010.

P. Biró, D. F. Manlove, and R. Rizzi. Maximum weight cycle packing in directed graphs, with application to kidney exchange programs. *Discrete Mathematics, Algorithms and Applications*, 1(4):499–517, 2009.

I. Caragiannis, A. Filos-Ratsikas, and A. D. Procaccia. An improved 2-agent kidney exchange mechanism. In *Proceedings of the 7th International Workshop on Internet and Network Economics*, 2011. Forthcoming.

G. Christodoulo, E. Koutsoupias, and A. Vidali. A lower bound for scheduling mechanisms. *Algorithmica*, 55(4):729–740, 2009.

O. Dekel, F. Fischer, and A. D. Procaccia. Incentive compatible regression learning. *Journal of Computer and System Sciences*, 76(8):759–777, 2010.

S. Dughmi and A. Ghosh. Truthful assignment without money. In *Proceedings of the 11th ACM Conference on Electronic Commerce*, pages 325–334, 2010.

H. N. Gabow. Data structures for weighted matching and nearest common ancestors with linking. In *Proceedings of the 1st Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 434–443, 1990.

R. Lavi and C. Swami. Truthful and near-optimal mechanism design via linear programming. In *Proceedings of the 46th Symposium on Foundations of Computer Science*, pages 595–604, 2005.

D. Lehmann, L. I. O'Callaghan, and Y. Shoham. Truth revelation in rapid, approximately efficient combinatorial auctions. *Journal of the ACM*, 49(5):577–602, 2002.

N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic Behavior*, 35 (1–2):166–196, 2001.

A. D. Procaccia and M. Tennenholtz. Approximate mechanism design without money. In *Proceedings of the 10th ACM Conference on Electronic Commerce*, pages 177–186, 2009.

A. E. Roth, T. Sönmez, and M. U. Ünver. Kidney exchange. *Quarterly Journal of Economics*, 119:457–488, 2004.

A. E. Roth, T. Sönmez, and M. U. Ünver. Pairwise kidney exchange. *Journal of Economic Theory*, 125:151–188, 2005.

A. E. Roth, T. Sönmez, and M. U. Ünver. Efficient kidney exchange: Coincidence of wants in markets with compatibility-based preferences. *American Economic Review*, 97:828–851, 2007.

T. Sönmez and M. U. Ünver. Market design for kidney exchange. In Z. Neeman, M. Niederle, and N. Vulkan, editors, *Oxford Handbook of Market Design*. Oxford University Press. Forthcoming.

P. Toulis and D. C. Parkes. A random graph model of kidney exchanges: Efficiency, individual-rationality and incentives. In *Proceedings of the 12th ACM Conference on Electronic Commerce*, pages 323–332, 2011.