Dynamic Matching via Weighted Myopia with Application to Kidney Exchange

John P. Dickerson and **Ariel D. Procaccia** and **Tuomas Sandholm**
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213
{dickerson,arielpro,sandholm}@cs.cmu.edu

Abstract

In many dynamic matching applications—especially high-stakes ones—the competitive ratios of prior-free online algorithms are unacceptably poor. The algorithm should take distributional information about possible futures into account in deciding what action to take now. This is typically done by drawing sample trajectories of possible futures at each time period, but may require a prohibitively large number of trajectories or prohibitive memory and/or computation to decide what action to take. Instead, we propose to learn *potentials* of elements (e.g., vertices) of the current problem. Then, at run time, we simply run an offline matching algorithm at each time period, but subtracting out in the objective the potentials of the elements used up in the matching.

We apply the approach to kidney exchange. Kidney exchanges enable willing but incompatible patient-donor pairs (vertices) to swap donors. These swaps typically include cycles longer than two pairs and chains triggered by altruistic donors. Fielded exchanges currently match myopically, maximizing the number of patients who get kidneys in an offline fashion at each time period. Myopic matching is sub-optimal; the clearing problem is dynamic since patients, donors, and altruists appear and expire over time. We theoretically compare the power of using potentials on increasingly large elements: vertices, edges, cycles, and the entire graph (optimum). Then, experiments show that by learning vertex potentials, our algorithm matches more patients than the current practice of clearing myopically. It scales to exchanges orders of magnitude beyond those handled by the prior dynamic algorithm.

Introduction

Kidney failure is a life-threatening health issue that affects hundreds of thousands of people worldwide. In the US alone, the waitlist for a kidney transplant had 90,563 patients as of January 23, 2012. This list is growing: demand far outstrips supply.

A recent innovation, kidney exchange, allows patients to bring an (incompatible) donor to a large pool where they can swap donors with other patients. Currently, fielded kidney exchanges match patients to donors in a myopic fashion, maximizing the number of candidates who get kidneys (on a weekly, monthly, or bimonthly basis) in an offline fashion. However, this is sub-optimal, since patients and donors arrive and leave the pool over time. Recent work shows that more candidates can be matched by considering the future (Awasthi and Sandholm 2009; Ünver 2010), but those approaches are either overly simplified or do not scale.

We introduce a method for informing myopic matching about the future in dynamic applications. It automatically learns *potentials* of structural elements of the problem (e.g., of vertices or edges in a graph) in a one-time offline fashion, then uses these potentials to guide myopic matching at run time. We apply these techniques to kidney exchange by first proving bounds on the power of learning potentials on vertices, edges, cycles, and the entire graph (which is optimal). Then, we learn potentials for vertices using a state-of-the-art parameter learning package and kidney exchange instance generator. We show that these learned potentials allow for a greater number of matches than simple myopic matching at nearly no run time cost across all the settings tested.

Static matching

One can encode a static n-patient kidney exchange as a directed graph $G(n)$ by constructing one vertex for each patient-donor pair. Add an edge e from v_i to v_j if the patient in v_j wants the item (donor kidney) of v_i. Each donor is willing to give a kidney if and only if the patient in his vertex v_i receives a kidney. The weight w_e of edge e represents the utility to v_j of obtaining v_i's kidney. In current practice, those weights are set by design committees, and they are typically equal or close to equal. A cycle c in this graph represents a possible swap, with each vertex in the cycle obtaining the kidney of the previous vertex. The weight w_c of a cycle c is the sum of its edge weights. A *matching* is a collection of disjoint cycles; they have to be disjoint because no donor can give more than one of his two kidneys.

The vanilla version of the *clearing problem* is to find a maximum-weight matching consisting of cycles with length at most some small constant L. Typically, $2 \leq L \leq 5$, and in most fielded kidney exchanges, including the UNOS nationwide kidney exchange with which we are closely involved, $L = 3$. The cycle length constraint L is crucial, since all operations in a cycle must be performed simultaneously. If they were not, a donor might back out after the patient in his ver-

tex has received a kidney.[1] In current practice, this matching is conducted using a single-shot optimization, with no concern for future matches. We call such matching *myopic*.

The clearing problem is NP-complete for $L > 2$ (Abraham, Blum, and Sandholm 2007). Significantly better solutions can be obtained by allowing cycles of length 3 instead of allowing 2-cycles only (Roth, Sönmez, and Ünver 2007). Using an integer linear program (ILP) where there is a decision variable for each cycle no longer than L, and constraints that state that accepted cycles are vertex disjoint, combined with specialized branch-and-price ILP solving code, the problem, with $L = 3$, is solvable to optimality in practice at the projected steady-state nationwide scale of 10,000 patients (Abraham, Blum, and Sandholm 2007). In our experiments, we use that algorithm as a subroutine whenever we have to solve the batch optimization problem.

A recent innovation in kidney exchange is *chains* (Roth et al. 2006; Montgomery et al. 2006; Rees et al. 2009). Each chain starts with an *altruistic* donor—that is, a donor who enters the pool, without a patient, offering to donate a kidney to any needy candidate in the pool, and expecting nothing in return. Chains start with an altruist donating a kidney to a candidate, whose paired donor donates a kidney to another candidate, and so on. Chains can be longer than cycles in practice because it is not necessary to carry out all the transplants in a chain simultaneously.[2]

Dynamic matching

In dynamic matching, the problem changes over time. In kidney exchange, patient-donor pairs and altruists enter and expire. While fielded kidney exchanges currently operate under the static paradigm described above, recent work in the kidney exchange community has shown that optimizing dynamically leads to higher cardinality matching overall.

From a theoretical standpoint, Ünver (2010) derives efficient dynamic mechanisms for general exchanges such that the total exchange surplus is maximized. These results are a generalization of an abstracted kidney exchange. He derives dynamic dispatch policies for kidney exchange in this abstract model; however, the model itself does not accurately reflect real-world kidney exchange. The results hinge on the assumption that one pair's candidate will be compatible with another pair's donor if they are blood type (aka ABO) compatible, ignoring other aspects of the potential match, most critically tissue type. His model also does not have chains.

Very few empirical results on non-myopic matching in dynamic kidney exchange are known. Most notable is a recent paper by Awasthi and Sandholm (2009) that uses trajectory-based online stochastic optimization algorithms to inform the matching algorithm of possible futures, thus potentially holding off matching some candidates and donors in an effort to increase overall matches later. Their results are promising, but the algorithm does not scale beyond very small exchanges due to the empirical complexity of sampling a large number of future world states, and the memory requirements associated with storing those trajectories and optimizing what to do in the present in light of them.

Many papers include experimental results on dynamic kidney exchange using myopic clearing (Gentry et al. 2009; Gentry and Segev 2011; Ashlagi et al. 2011; Ashlagi, Gamarnik, and Roth 2011; Dickerson, Procaccia, and Sandholm 2012). This is useful due to the ubiquity of myopic matching in fielded kidney exchanges, but, as we show in this paper, ultimately the strategy is flawed. Myopic matching that is *informed about the future* results in more matches.

Using *potentials* to inform myopia

In this paper, we introduce the idea of *potentials* to capture the future into myopic matching. Given a structural element (e.g., vertex, edge, cycle type, etc.) of the problem, a potential $P \in \mathbb{R}$ quantifies the future expected usefulness to the exchange of that element.

For example, consider potentials on vertices. In terms of ABO compatibility, an O-type donor can give to O-, A-, B-, and AB-type patients, an A-donor can give to A or AB, a B-donor can give to B or AB, and an AB-donor can only give to AB. Therefore, an O-type altruistic donor typically leads to more matched patients (i.e., has higher potential) than other types of altruist. Similarly, an altruist with a given blood type tends to have higher potential than a patient-donor vertex with that same donor type because the pair is harder to match, since it expects a kidney in return and that kidney needs to be compatible with the pair's patient. Intuitively, then, it might make sense to not match an O-altruist until he or she can trigger a long chain of patient-donor pairs in the pool. Similarly, if a feasible match for a hard-to-match patient-donor pair exists at the present time, it might make sense to immediately match this pair since saving the pair for later would likely yield no benefit (and would risk that pair never being matchable in the future). So, the altruist would be given a high potential: by saving the O-type altruist until she triggered a long chain, more lives would be saved overall. Similarly, the hard-to-match patient-donor pair would be given a low potential to incentivize immediate matching.

As described in the introduction, the exchange clearing problem finds a maximum-weight exchange of disjoint cycles. Given the potential for a vertex, edge, or cycle type (where type is defined by the ABO blood types of the donors and the patients in the vertices), we can easily translate this information into a language the matching algorithm understands. For example, with vertex potentials, the translation works as follows. Given vertex potentials P_X and P_Y representing the potentials of patient-donor pairs of ABO type X and Y, respectively, any edge e between vertices of type X and Y receives weight $w_e = f(P_X, P_Y)$, for some function $f : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. Cycles in the exchange are then assigned weights as usual, as the sum of their edge weights. In this way, the potentials assigned to specific elements affect the final maximum-weight exchange of disjoint cycles.

In our example above, say $P_{-O} = 2.1$ and $P_{O-AB} = 0.1$, representing the vertex potentials of an O-type altruist and an O-AB type patient-donor pair, respectively. Furthermore, define $f(P_X, P_Y) = 1 - \frac{1}{2}(P_X + P_Y)$. (This formula

[1] This reneging cannot be prevented by legal means because it is illegal to contract for organs in most countries.

[2] If a chain breaks by some donor reneging, the chain merely stops, but no pair is out their "bargaining chip" (donor kidney).

assumes that edge weights before taking potential into account are 1, as is the case in the state-of-the-art kidney exchange simulator (Saidman et al. 2006), which we use. Some fielded kidney exchanges also set all edge weights to 1, and others set them roughly equally. The methodology in this paper applies to unequal edge weights as well.) Then, any edge e between an O-type altruist and an O-AB type patient-donor pair will receive weight $1 - 0.5(2.1 + 0.1) = -0.1$. Informally, this is telling the matching algorithm that any chain c including edge e—triggered by the extremely valuable O-type altruist—will need to be long (i.e., high weight) enough to offset the negative weight of e.

In the ABO model of kidney exchange, there are 20 possible vertex types: 4 types of altruists (O-, A-, B-, and AB-type), and 16 types of patient-donor pairs (O-O, O-A, ..., AB-AB-type). If we consider edge types, this number jumps to 244: 208 possible edges originating from patient-donor pairs, and 36 originating from altruists. Allowing potentials to be learned for variable-length cycles increases this number dramatically. Intuitively, there is a tradeoff between the expressive power of the potentials (allowing potentials for larger structural elements such as edges, or cycles, or even beyond, having more expressiveness) and the computational power needed to learn the potentials (the hypothesis space being larger the more variables there are). To lend insight to this, in the next section, we prove bounds that compare associating potentials with vertices, edges, cycles, or even higher-level elements such as the entire graph.

How much can associating potentials to larger elements help?

In this section, we prove bounds on the best-case benefit from associating potentials on larger elements compared to associating them with smaller elements. (However, as we will experimentally show later in the paper, even properly setting vertex potentials works very well in practice.)

First, in Theorem 1 we compare the application of potentials to vertices and to edges and show that edge potentials can do notably better. Second, Theorem 2 considers allowing potentials to be applied to edges and cycles; again, we show that the finer-grained resolution of cycles can allow better overall results than just edges. Finally, Theorem 3 shows that, in certain pathological cases, even applying potentials to cycles can perform poorly compared to potentials on unlimited graph elements (which is equivalent to comparing against an omniscient algorithm with perfect foresight).

The constructions in the proofs are in the two- and three-stage setting and work even if the clearing engine is omniscient about the future. To evaluate the quality of a choice of potentials, we compute the number of matched pairs as follows: in the first stages the number of matched pairs plus the potentials of leftover elements is maximized, and in the final stage the number of matched pairs is maximized.

Vertex versus edge potentials. We first consider associating potentials only to the vertices, in the two-stage model.

Theorem 1 (Vertex potentials vs edge/graph potentials)**.**

1. *For every $k \in \mathbb{N}$ there exists an input with cycles of length at most $2k + 4$ and no chains such that for any choice of vertex potentials the number of matched patients is at most a $(4k + 4)/(6k + 4)$-fraction of the optimum.*
2. *For every $k \in \mathbb{N}$ there exists an input with 2-cycles and chains of length at most $2k + 5$ such that for any choice of vertex potentials the number of matched patients is at most a $(4k + 5)/(6k + 5)$-fraction of the optimum.*[3]
3. *There exists an input with cycles of length at most 2 and no chains such that for any choice of vertex potentials the match size is at most a $(5/6)$-fraction of the optimum.*

In each of the three cases, the construction is such that the optimum is achievable using edge potentials.

Proof. For part 1, given $k \in \mathbb{N}$, consider the input illustrated in Figure 1. In stage 1 there is a 2-cycle between AB-O and AB-A, a 2-cycle between A-A and A-O, k 2-cycles between AB-O and A-O, and k 2-cycles between AB-A and A-A. In stage 2 the gray vertices disappear, and $2k$ AB-AB vertices appear, so that a $(2k+4)$-cycle is formed between the white and dashed vertices. Note that an edge exists between two donor-patient pairs only if they are blood type compatible.

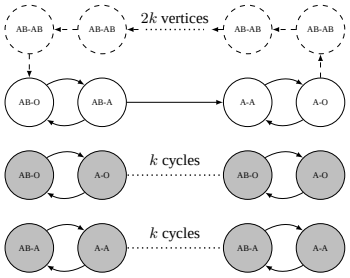

Figure 1: Example of Theorem 1. Vertices present in both stages are white. Vertices present only in stage 2 are dashed. Vertices present only in stage 1 are gray.

The optimal solution matches all gray cycles in the first stage, and the long cycle in the second stage, for a total of $6k+4$ pairs matched. This can easily be accomplished using edge potentials but, as we will show, not vertex potentials.

To analyze the quality of the solution when using vertex potentials, first consider the case where the 2-cycle between AB-O and AB-A and the 2-cycle between A-A and A-O both remain unmatched at the end of the first stage. Because these cycles are disjoint from other cycles in the graph, they can remain unmatched only if the potential of their vertices is greater than their length; formally, $P_{\text{AB-O}} + P_{\text{AB-A}} > 2$ and $P_{\text{A-A}} + P_{\text{A-O}} > 2$. By summing these two inequalities we obtain the inequality

$$P_{\text{AB-O}} + P_{\text{AB-A}} + P_{\text{A-A}} + P_{\text{A-O}} > 4. \quad (1)$$

It follows that $P_{\text{AB-O}} + P_{\text{A-O}} > 2$ or $P_{\text{AB-A}} + P_{\text{A-A}} > 2$. Indeed, otherwise by summing we would obtain that $P_{\text{AB-O}} + P_{\text{AB-A}} + P_{\text{A-A}} + P_{\text{A-O}} \leq 4$, in contradiction to Equation (1). Assume without loss of generality that $P_{\text{AB-O}} + P_{\text{A-O}} > 2$.

[3] As is common practice, we say that the last donor in a chain donates to the deceased donor waiting list (not shown in our illustrations). That is included in all our analyses and experiments. For example, if an altruist donates to a pair that donates to the waiting list, the chain length is 2 and the number of patients saved is 2.

Therefore, in stage 1 the k cycles between AB-O and A-O pairs are not matched. It follows that the number of pairs that are matched can be at most $4k + 4$.

We now consider the case where the first-round matching includes either the 2-cycle between AB-O and AB-A or the 2-cycle between A-A and A-O. In stage 2, the $2k$ AB-AB vertices in the long cycle cannot be matched. Thus, the number of matched pairs is at most $4k + 4$. We see that the ratio between the number of patients matched under optimal vertex potentials and the optimum is at most $(4k+4)/(6k+4)$.

The input for part 2 is almost identical, and is obtained by removing the edge from AB-AB to AB-O, and adding an altruistic donor (say with blood type O), who appears in stage 2, with an edge to the white AB-O pair. The proof of part 3 is omitted due to lack of space. □

As k grows in Theorem 1, the ratio of vertex to edge (and optimal) potentials in parts 1 and 2 tend toward $2/3$. This is a negative result in the high-stakes world of kidney exchange, where losing $1/3$ of the possible matches is highly undesirable. However, the subsection below shows that worst-case performance is poor even if we allow potentials on edges.

Edge versus cycle potentials. We now show that edge potentials can have poor performance compared to cycle potentials, tending toward matching 1/2 of the matchable patients in a two-stage model. We allow the cycle potential to be a function of all the ABO types of the vertices of the cycle.

Theorem 2 (Edge potentials vs cycle/graph potentials)**.**

1. *For every $k \in \mathbb{N}$ there exists an input with cycles of length at most $3k + 2$ and no chains such that for any choice of edge potentials the number of matched patients is at most a $(3k + 2)/(6k + 2)$-fraction of the optimum.*
2. *For every $k \in \mathbb{N}$ there exists an input with 2-cycles and chains of length at most $3k+3$ such that for any choice of edge potentials the number of matched patients is at most a $(3k + 3)/(6k + 3)$-fraction of the optimum.*

In both of these cases, the construction is such that the optimum is achievable using cycle potentials.

Proof sketch. The construction in Figure 2 proves case 1. Cycle potentials match the gray nodes and the long cycle,

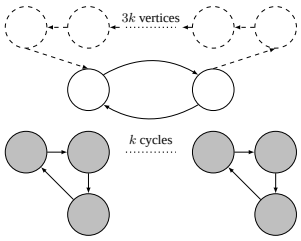

Figure 2: Example of Theorem 2.

while edge potentials can match either the gray and solid white vertices or only the long cycle. In this construction, every vertex has the same ABO-type, so each edge must have the same potential P. Assume that the edge weights before potential is taken into account are all 1. Now, if $P > 1$, the algorithm must match nothing in stage 1 while if $P < 1$, it must match the 2-cycle in stage 1 (along with the 3-cycles), which precludes it from matching the long cycle in stage 2. The adjustment for case 2 is the same as in Theorem 1. □

Cycle versus graph potentials. We now show that we cannot get optimal matching in the worst case even if we associate potentials with entire cycles, in a three-stage model (where all vertices expire after the second stage).

Theorem 3 (Cycle potentials vs graph potentials)**.** *Denote by L the cap on cycle length. There exists an input with cycles of length at most L (even with no altruistic donors) such that for any choice of cycle potentials the number of matched patients is at most a $1/L$-fraction of the optimum.*

Proof. In Figure 3, the optimal solution is to pass on the L-cycle in stage 1, but to accept all the L-cycles except the central one in stage 2. The cycle-potential algorithm cannot

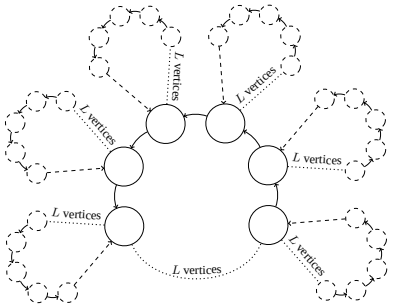

Figure 3: Example of Theorem 3.

accomplish that because it has to wait on all L-cycles or to take them all (since this construction assumes that all edges have the same pre-potential weight, and all L-cycles have the same potential because all the vertices have the same ABO type). It must either take the L-cycle in stage 1 or nothing in either stage; the former choice is better. □

While the theoretical results in this section show that there can be a significant benefit to associating potentials with larger elements, in the experiments that follow, we will use potentials on vertices. This is motivated by there being fewer weights to learn. As we will show, even learning that number of weights is challenging, but as we also show, the approach works well in practice.

Learning the values of potentials

In this section, we describe our technique for learning potentials for elements of kidney exchange. We use ParamILS (Hutter et al. 2009), a state-of-the-art algorithm configuration package, to intelligently search through the parameter space for an optimal (or near-optimal) instantiation of the potential variables. The method given is general enough to handle potentials on vertices, edges, cycles, and so on. However, due to the computational complexity of learning potentials, we focus on learning vertex potentials only.

Given an objective function and a parameter space over which to search, ParamILS takes an initial vector of parameter values and, using iterated local search, tries to optimize the objective by adjusting the parameters. In our case, the

parameters are 20 real-valued vertex potentials (16 patient-donor pair ABO types plus 4 altruist ABO types). Our objective function in ParamILS was to maximize the number of patients matched as a fraction of the optimal number of patients that could be matched in a full information model. This ratio is measured by running the myopic kidney exchange solver using the vertex potentials as parameters.

ParamILS requires discretization of the parameters. We let each potential live in the space $\{0, 0.2, \ldots, 3.0\}$. (The highest value that ParamILS ended up with was 1.4.)

To learn the parameter values, ParamILS uses a training set and an internal test set. Our sets contained 1500 and 300 kidney exchange graphs, respectively. Each graph had 95 patient-donor pairs and 5 altruists total arriving over 25 months, and was generated using the standard kidney exchange generator (Saidman et al. 2006). We amended the generator to generate altruists to correspond to fielded kidney exchanges. The expiration rate of vertices was set according to the reality that 12% of kidney patients survive 10 years (USRDS 2007). The expiration rate for altruists was set to be the same. The other ParamILS settings were: 1000 runs per random seed per graph, and 1000 random graphs per parameter vector. The authors of ParamILS recommend running ParamILS multiple times with different settings of the `numRuns` parameter; we ran 24 times with different values and ran for three days on each. Finally, we chose the parameter vector with highest score across all runs.

None of the ParamILS iterated local search runs terminated. This is due to (a) the high variability in running dynamic match runs on the same compatibility graphs, (b) the high variability between different compatibility graphs, and (c) the long run time required to solve each instance (the clearing problem is NP-hard). However, as our results in the next section show, the learned weights did, in fact, improve myopic matching significantly. Also, the relative sizes of the learned parameter values made sense, with easier-to-match vertices receiving greater potentials.

Weighted myopic matching at run time

We now present extensive experimental results using the potentials learned on our training and ParamILS-internal test data sets. We test on a new test set starting with the problem sizes used for training and then testing on much larger instances. We conclusively show that the learned vertex potentials increase the total number of matches made by the (currently fielded) myopic matching algorithm. Furthermore, once these potentials are learned, they do not seriously affect match run time. As such, the current state-of-the-art solving algorithm will still be able to clear exchanges at the estimated nationwide steady-state size of 10,000 pairs.

Comparing to optimal matching. We begin by comparing our weighted myopic algorithm and standard myopic algorithm to an optimal matching. The optimal matching is computed in a full-information model where the optimization algorithm is given access up front to exactly which patient-donor pairs and altruists will be in the pool at each period. It is impossible for any algorithm to match more candidates—and unlikely that any would be able to match equally many due to lack of full information about the future.

As in prior experiments on dynamic kidney exchange, each simulation was conducted over 51 time periods, representing 4 years and 3 months of actual time. We vary the number of patient-donor pairs from 110 to 710 and add 5% as many altruists as there are patient-donor pairs. For example, with 510 pairs, there are 25 altruists in the pool.[4] This number is motivated by the UNOS pilot nationwide kidney exchange that we have been working to establish. Altruistic donors were incorporated into that exchange in April 2011. Recently, their number has been roughly 5% of the number of patient-donor pairs.

Figure 4 shows the improvement gained by our weighted myopic matching algorithm, relative to the difference between plain myopic matching and the optimal match size. For example, if the standard myopic algorithm matched 300 candidates, the optimal matched 360, and the weighted myopic algorithm matched 315, we report $(315 - 300)/(360 - 300) = 25\%$. Clearly, vertex potentials help. Interestingly, as the number of patient-donor pairs and altruists increases, the weighted myopic algorithm tends toward the optimal solution more quickly than standard myopic matching.

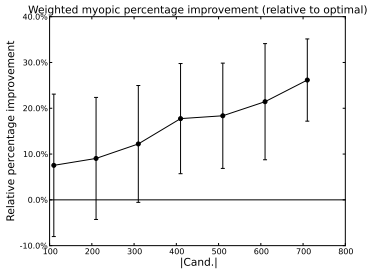

Figure 4: Percentage gain of weighted over unweighted myopic matching relative to the optimal match cardinality.

The table in Figure 5 gives the statistical significance of our results. Statistical significance testing was done using a Wilcoxon signed-rank test. The p-values and Z-test statistics (which are more useful than p-values for large-sample experiments like ours) clearly show the significance of the weighted matching algorithm's gains.

patient-donor pairs	%improvement	samples	p-value	Z-statistic
110	**7.54**	240	2.970e-204	1972.5
210	**9.04**	240	6.240e-40	2371.5
310	**12.21**	240	4.515e-37	1722.5
410	**17.74**	240	5.994e-41	321.0
510	**18.36**	239	1.468e-41	185.5
610	**21.44**	66	1.400e-12	8.5
710	**26.16**	15	5.320e-04	0.0

Figure 5: Statistical significance tests. The %improvement over unweighted is relative to the optimal matching.

Scaling to larger graphs. We now study how the approach scales to larger problems. Because the parameter learning was complex already at a smaller problem size, we do not attempt to re-learn the parameters for the larger problems.

Furthermore, calculating the optimal match size quickly becomes intractable, since the clearing algorithm must con-

[4] At each time period, the number of pairs and number of altruists are drawn from a normal distribution; this is retried until the total number of each of the two across all 51 time periods is correct.

sider an unrealistically large pool over all 51 time periods at once. Therefore, we only evaluate the efficacy of our algorithm against the myopic matching algorithm. We maintain the test setup from the last section, with 5% as many altruists as there are patient-donor pairs.

Figure 6 gives two sample cumulative distribution functions of the weighted myopic gains over unweighted myopic in terms of total match size. White bars correspond to the weighted algorithm matching more than unweighted. Black bars represent when our method was beaten by unweighted myopic. Losses such as these are unavoidable in any non-full-information model; however, in our experiments losses were rare and significantly smaller than respective gains.

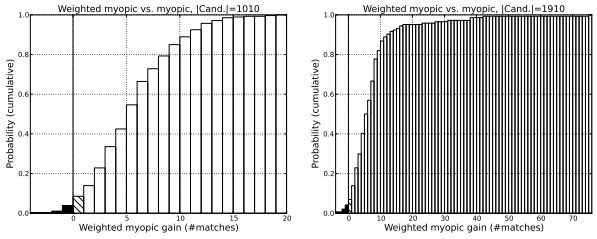

Figure 6: CDFs of absolute gains of weighted versus unweighted myopic matching. White bars correspond to the weighted approach outperforming the unweighted.

Figure 7 shows the percentage gain of the weighted myopic algorithm relative to the match cardinality of the unweighted myopic matching algorithm. While these percentages are relatively small, it is important to note that (1) the full-information upper bound on the amount of room for improvement is rather low and our algorithm captures a large percentage of that (Figure 4), (2) these are improvements in real lives saved, so even small improvements are important, and (3) the improvements are statistically significant as discussed above. Furthermore, some of the decline in absolute percentage gain may be due to our learning vertex potentials on significantly smaller graphs.

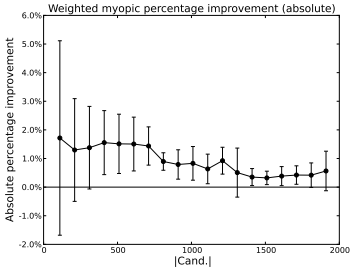

Figure 7: Percentage gain of weighted over unweighted myopic matching relative to the myopic match cardinality.

The role of altruistic donors

One significant unknown in kidney exchange is the ratio of altruistic donors to patient-donor pairs. More altruists in the pool drastically increases the fraction of patients matched (see, for example, recent work by Dickerson et al. (2012)). In this section, we vary the percentage of altruists in the pool relative to the number of patient-donor pairs. It turns out that our method works well when there are no altruists, few altruists, or many altruists. This is despite the fact that we do not re-learn potentials for a specific density of altruists.

For these experiments, we held the number of patient-donor pairs in the pool constant at 510, and varied the number of altruists from 0 to 120. Figure shows the absolute gain in the number of candidates matched by our weighted myopic algorithm over the standard myopic algorithm. As above, the absolute gain is 1–1.5%. Interestingly, the absolute gain percentage decreases as the number of altruists increases; this can be explained by myopic matching being closer to optimal when the number of altruists is large. Figure shows the percentage gain of weighted myopic matching relative to the gap between optimal matching and traditional myopic matching. We see that using the vertex potentials learned earlier results in matching 10–25% of the candidates left unmatched by unweighted myopic matching, *regardless of the number of altruists in the pool*.

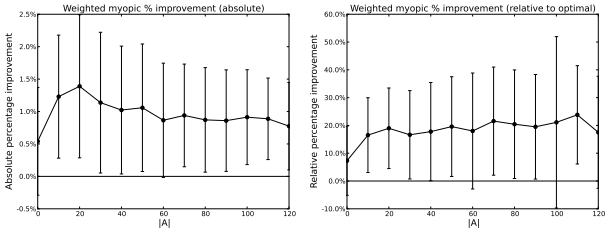

Figure 8: Percentage gain of weighted myopic over unweighted myopic matching relative to the myopic match cardinality (left) and optimal match cardinality (right).

Figure 9 shows that the vertex potentials improve the matching with extremely high statistical significance. Again, Wilcoxon signed-rank test methodology was used.

altruists	%improvement	samples	p-value	Z-statistic
0	**7.29**	369	2.507e-43	9098.5
10	**16.51**	280	4.759e-45	1003.5
20	**18.97**	280	1.304e-44	1196.0
30	**16.63**	280	1.572e-40	1725.0
40	**17.77**	280	1.863e-38	1808.0
50	**19.58**	280	8.005e-42	1355.5
60	**18.00**	280	1.906e-37	2231.0
70	**21.55**	280	4.829e-44	1253.5
80	**20.43**	280	2.407e-41	1483.0
90	**19.50**	270	1.763e-41	1438.5
100	**21.10**	160	5.974e-26	316.5
110	**23.81**	65	3.552e-12	13.5
120	**17.51**	49	1.769e-09	45.5

Figure 9: Statistical significance tests for graphs with 510 patient-donor pairs and varying number of altruistic donors.

Potentials on altruists only. We also ran experiments where we learned and tested potentials on altruists only. This significantly reduces the search space size to 4 parameters. ParamILS learned the following potentials for O-, A-, B-, and AB-altruists, respectively: 0.8, 0.6, 0.4, and 0.2. This makes qualitative sense because O-altruists are easiest to match and AB-altruists the hardest. Interestingly, the potential is less than 1 even for O-altruists. Perhaps surprisingly, this approach led only to a tiny improvement over unweighted myopic matching across graph sizes. For example,

averaged over 1500 runs with 410 vertices each, the relative gain over unweighted myopic was 0.06% while it was 16.3% when learning all 20 potentials (over patient-donor pairs and altruists).

Conclusions and future research

We introduced an automated, scalable method for informing myopic matching algorithms about the future. It learns *potentials* of elements of the problem offline and then uses the potentials to guide myopic matching at run time.

We applied these techniques to kidney exchange. We theoretically compared the power of using potentials on increasingly large elements: vertices, edges, cycles, and the entire graph. Then, experiments showed that by learning vertex potentials, our algorithm matches more patients than the current practice of clearing myopically—at nearly no run time cost. We experimented with a variety of graph types; weighted myopic matching helped on them all. It scales to the projected nationwide kidney exchange size.

A clear direction for future research would be an empirical comparison of potentials learned on vertices to larger elements. There is a tradeoff: on the one hand, as we proved, associating potentials with larger elements has more power, but on the other hand, there are more parameters to learn. ParamILS did not converge even on vertex potentials; we conjecture that a new learning method is required to move beyond vertex potentials. This could involve using domain knowledge of partial ordering of potentials.

Mechanism design results in dynamic kidney exchange are few and far between, with one notable exception (Ünver 2010). It would be interesting to extend these results along the lines of static kidney exchange mechanism results (Ashlagi et al. 2010; Ashlagi and Roth 2011; Caragiannis, Filos-Ratsikas, and Procaccia 2011; Toulis and Parkes 2011), and to perhaps tie in the algorithms presented in this paper.

Acknowledgments

This material is based on work supported by the National Science Foundation under grants IIS-0905390, IIS-0964579, and CCF-1101668, and TeraGrid resources provided by the Pittsburgh Supercomputing Center. We also acknowledge Intel Corporation and IBM for their machine gifts.

References

Abraham, D.; Blum, A.; and Sandholm, T. 2007. Clearing algorithms for barter exchange markets: Enabling nationwide kidney exchanges. In *ACM Conference on Electronic Commerce (EC)*.

Ashlagi, I., and Roth, A. 2011. Individual rationality and participation in large scale, multi-hospital kidney exchange. In *ACM Conference on Electronic Commerce (EC)*.

Ashlagi, I.; Fischer, F.; Kash, I.; and Procaccia, A. 2010. Mix and match. In *ACM Conference on Electronic Commerce (EC)*.

Ashlagi, I.; Gilchrist, D.; Roth, A.; and Rees, M. 2011. Nonsimultaneous chains and dominos in kidney-paired donation—revisited. *American Journal of Transplantation* 11(5):984–994.

Ashlagi, I.; Gamarnik, D.; and Roth, A. 2011. The need for chains in kidney exchange. In *NBER Market Design Workshop*.

Awasthi, P., and Sandholm, T. 2009. Online stochastic optimization in the large: Application to kidney exchange. In *Int. Joint Conference on Artificial Intelligence (IJCAI)*.

Caragiannis, I.; Filos-Ratsikas, A.; and Procaccia, A. 2011. An improved 2-agent kidney exchange mechanism. In *Workshop on Internet and Network Economics (WINE)*.

Dickerson, J.; Procaccia, A.; and Sandholm, T. 2012. Optimizing kidney exchange with transplant chains: Theory and reality. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*.

Gentry, S., and Segev, D. 2011. The honeymoon phase and studies of nonsimultaneous chains in kidney-paired donation. *American Journal of Transplantation* 11(12):2778–2779.

Gentry, S.; Montgomery, R.; Swihart, B.; and Segev, D. 2009. The roles of dominos and nonsimultaneous chains in kidney paired donation. *American Journal of Transplantation* 9(6):1330–1336.

Hutter, F.; Hoos, H.; Leyton-Brown, K.; and Stützle, T. 2009. ParamILS: An automatic algorithm configuration framework. *Journal of Artificial Intelligence Research* 36(1):267–306.

Montgomery, R.; Gentry, S.; Marks, W..; Warren, D.; Hiller, J.; Houp, J.; Zachary, A.; Melancon, J.; Maley, W.; Rabb, H.; Simpkins, C.; and Segev, D. 2006. Domino paired kidney donation: a strategy to make best use of live non-directed donation. *The Lancet* 368(9533):419–421.

Rees, M.; Kopke, J.; Pelletier, R.; Segev, D.; Rutter, M.; Fabrega, A.; Rogers, J.; Pankewycz, O.; Hiller, J.; Roth, A.; Sandholm, T.; Ünver, U.; and Montgomery, R. 2009. A nonsimultaneous, extended, altruistic-donor chain. *New England Journal of Medicine* 360(11):1096–1101.

Roth, A.; Sönmez, T.; Ünver, U.; Delmonico, F.; and Saidman, S. 2006. Utilizing list exchange and nondirected donation through 'chain' paired kidney donations. *American Journal of Transplantation* 6:2694–2705.

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

Saidman, S. L.; Roth, A.; Sönmez, T.; Ünver, U.; and Delmonico, F. 2006. Increasing the opportunity of live kidney donation by matching for two and three way exchanges. *Transplantation* 81(5):773–782.

Toulis, P., and Parkes, D. 2011. A random graph model of kidney exchanges: efficiency, individual-rationality and incentives. In *ACM Conference on Electronic Commerce (EC)*.

Ünver, U. 2010. Dynamic kidney exchange. *Review of Economic Studies* 77(1):372–414.

2007. United States Renal Data System (USRDS). http://www.usrds.org/.