
Kidney Exchange with Inhomogeneous Edge Existence Uncertainty

Hoda Bidkhori
University of Pittsburgh

John P Dickerson
University of Maryland

Duncan C McElfresh
University of Maryland

Ke Ren
University of Pittsburgh

Abstract

Patients with end-stage renal failure often find kidney donors who are willing to donate a life-saving kidney, but who are medically incompatible with the patients. Kidney exchanges are organized barter markets that allow such incompatible patient-donor pairs to enter as a single agent—where the patient is endowed with a donor “item”—and engage in trade with other similar agents, such that all agents “give” a donor organ if and only if they receive an organ in return. In practice, organized trades occur in large cyclic or chain-like structures, with multiple agents participating in the exchange event. Planned trades can fail for a variety of reasons, such as unforeseen logistical challenges, or changes in patient or donor health. These failures cause major inefficiency in fielded exchanges, as if even one individual trade fails in a planned cycle or chain, *all or most of the resulting cycle or chain fails*. Ad-hoc, as well as optimization-based methods, have been developed to handle failure uncertainty; nevertheless, the majority of the existing methods use very simplified assumptions about failure uncertainty and/or are not scalable for real-world kidney exchanges.

Motivated by kidney exchange, we study a stochastic cycle and chain packing problem, where we aim to identify structures in a directed graph to maximize the expectation of matched edge weights. All edges are subject to failure, and the failures can have nonidentical probabilities. To the best of our knowledge, the state-of-the-art approaches are only tractable when failure probabilities are identical. We formulate a relevant non-convex optimization problem and propose a tractable

mixed-integer linear programming reformulation to solve it. In addition, we propose a model that integrates both risks and the expected utilities of the matching by incorporating conditional value at risk (CVaR) into the objective function, providing a robust formulation for this problem. Subsequently, we propose a sample-average-approximation (SAA) based approach to solve this problem. We test our approaches on data from the United Network for Organ Sharing (UNOS) and compare against state-of-the-art approaches. Our model provides better performance with the same running time as a leading deterministic approach (PICEF). Our CVaR extensions with an SAA-based method improves the $\alpha \times 100\%$ ($0 < \alpha \leq 1$) worst-case performance substantially compared to existing models.

1 INTRODUCTION

Kidney exchange is a centralized barter market where patients with end-stage renal disease trade willing donors in cyclic or chain-like transactions [Abraham et al., 2007, Rapaport, 1986, Roth et al., 2004]. The aim of the kidney exchange clearinghouse is to find the “best” disjoint set of such swaps—i.e., to solve a cycle and chain packing problem. Exchanges already account for over 12% of living kidney donations in the US, and exchange programs are growing worldwide [Biró et al., 2019]—including via extensions to liver and lung [Ergin et al., 2017], and even multi-organ [Dickerson and Sandholm, 2017], exchange. Fielded exchanges face several sources of inefficiency, primarily due to pre-transplant “failure” [Leishman, 2019]; that is, most *planned* transplants never result in transplantation due to medical or logistical incompatibility [Agarwal et al., 2019, Alvelos et al., 2015, Anderson et al., 2015, Dickerson et al., 2016, 2018, Glorie et al., 2014, Goldberg and Poss, 2019, Klimen-

tova et al., 2016, Manlove and O’Malley, 2015]. In other words, the exchange program cannot be certain whether a compatible patient and donor will result in a transplant. Exchanges are often represented by directed graphs (see § 2), where edges indicate potential transplants and edge weights reflect the medical or social utility of the transplant. If a planned transplant (i.e., *edge*) fails, its effects can cascade through the exchange, causing other edges to fail (see § 2.1)—thus, edge failures can severely impact the overall utility of an exchange. Thus it is of interest for exchange coordinators to account for uncertainty when planning transplants. Kidney exchange with *edge existence uncertainty*—specifically, selecting transplants which minimize impact from edge failures—is the focus of this paper.

Many approaches have been proposed to deal with edge existence uncertainty, often using stochastic or robust optimization techniques. However most of these models either make the simplifying assumption that all transplants are equally likely to fail, or they require intractable algorithms that cannot be used on large exchanges.

We fill a major gap in prior work by proposing the first *scalable* algorithm (meaning it uses a number of variables polynomial in the input size) for maximizing expected matching weight, with *non-identical* failure probabilities. This is an important step forward, as failure probabilities are known to be inhomogeneous—some edges are inherently riskier than others [Dickerson et al., 2018]. We provide a mixed-integer linear program for our approach, which is compact and can be solved directly by a general-purpose integer programming solver (e.g., CPLEX, Gurobi, or SCIP). In computational experiments we demonstrate that accounting for inhomogeneous edge probabilities improves over state-of-the-art approaches, using data from the United Network for Organ Sharing.

Additionally, we propose a modified version of the kidney exchange problem which balances the *mean expected weight* with the *worst-case* weight (“risk”) of an exchange with known nonidentical edge failure probabilities; we achieve this balance using a conditional value-at-risk (CVaR) objective. This approach is motivated by the fact that expected weight can be misleading when the worst-case outcome can be arbitrarily bad (see § 2.1). We are not the first to propose a CVaR approach for kidney exchange; however, previous CVaR-based approaches do not allow for arbitrary length limits on cycles and chains—which are used by all fielded exchanges. With cycle and chain length limits, the kidney exchange problem with a CVaR objective is challenging, as there is no closed-form expression for the objective function. Thus, we propose a sample-average-approximation-based method

and develop an equivalent mixed-integer linear programming representation. Computational experiments show that our model improves the worst-case mean over state-of-the-art methods.

1.1 Uncertainty in Kidney Exchange

Many prior approaches address edge existence uncertainty in kidney exchange, often with the objective of maximizing *expected* matching weight, assuming all edges have identical failure probability. Dickerson et al. [2016] provides a scalable formulation in this case, and Dickerson et al. [2018] extends this to consider inhomogeneous edge probabilities; however the latter model can require enumeration of all feasible cycles and chains, which can be intractable for even small exchanges. Similar approaches have been proposed, but still assume that all edges have equal failure probability [Alvelos et al., 2015, Constantino et al., 2013]. Rather than maximizing expected edge weight, other approaches take the *risk-averse* perspective, aiming to maximize the worst-case matching weight [Carvalho et al., 2020, McElfresh et al., 2019]; these approaches are often *too* conservative, as the worst case in kidney exchange is often arbitrarily bad (i.e., in the worst case, all planned transplants fail). Zheng et al. [2015] propose a CVaR method that endogenously balances structure length with risk; however, their model is not amenable to length caps on cycles and/or chains, a requirement in all fielded kidney exchanges.

Several other optimization-based approaches have been proposed, using recourse [Anderson et al., 2015], forms of “fallback” options [Bartier et al., 2019, Manlove and O’Malley, 2015, Wang et al., 2019], and pre-match edge queries [Blum et al., 2013, 2015, McElfresh et al., 2020]. These methods involve additional decision stages, and are not directly comparable in our setting.

Next we describe the formal model of kidney exchange and edge existence uncertainty.

2 PRELIMINARIES

We represent a kidney exchange as a directed graph $G = (E, V)$ where each vertex $v_i \in V$ is an incompatible patient-donor pair, or a non-directed donor (NDD, i.e., a donor without a paired patient). Directed edges $e = (v_i, v_j)$ represent potential transplants from the donor of node i to the patient of node j ; edge weights $w_e > 0$ represent the medical or social utility of each potential transplant. We assume that edge failure probabilities $p_e \in [0, 1]$ are known in advance and are not necessarily homogeneous. That is, if edge $e = (v_i, v_j)$ is matched, then with probability p_e the patient of v_j would still fail to receive a kidney from v_i ’s donor.

Kidney exchanges consist of two types of swaps: *cycles* consist of several patient donor pairs, while *chains* begin with an NDD and continue through one or more patient pairs [Roth et al., 2005a]. The goal of the *kidney exchange clearing problem* (KEP) is often to select the set of vertex-disjoint cycles and chains in G which maximize overall edge weight. We refer to any set of vertex-disjoint cycles and chains as a *matching*. For example, let w denote the vector of weights for all cycles and chains in the graph, let x denote a vector of binary decision variables, and let \mathcal{M} denote the set of feasible matchings (i.e., binary vectors x corresponding to vertex-disjoint cycles and chains); in this case the KEP is expressible as $\max_{x \in \mathcal{M}} x^\top w$.

Cycles and chains are quite vulnerable to edge failure: if *any* edge in a cycle fails, then *none* of the transplants in the cycle can proceed, because at least one of the patients will be left without a compatible donor. If an edge participating in a chain fails, then none of the edges *following* that failed edge can proceed.¹

We consider modified versions of the KEP which account for edge failures, using known edge failure probabilities. Before describing our approach, we emphasize that the choice of *objective* is important in the KEP. We demonstrate this point with a small example.

2.1 Example: Edge Existence Uncertainty

The choice of the *objective function*—and, in particular, its treatment of uncertainty—can substantially impact the structure of the final matching. Consider the exchange in Figure 1, in which there are four possible matchings: 2-cycle (1, 2), 2-cycle (1, 3), 3-chain (n, 1, 2), and 3-chain (n, 1, 3).² All edges have integer weight w and failure probability p ; only the edge from n to pair 1 is guaranteed to succeed ($p = 0$). Any of the four feasible matching in this graph might be “optimal,” depending on the choice of objective.

An objective that maximizes overall matching weight (i.e., the objective used by many fielded exchanges [Anderson et al., 2015, UNOS]) would select 2-cycle (1, 2) with total weight 10. However this matching is likely to fail: at least one cycle edge will fail with probability 0.84—in which case the matching receives zero weight. Instead, we might maximize *expected* matching weight (e.g., as in Dickerson et al. [2018]), and select 2-cycle (1, 3). Indeed this matching achieves total expected weight 6.23, nearly twice the expected weight

¹We assume that chains can be *partially* executed. Some fielded exchanges cancel the entire chain if even one edge fails.

²The 2-chain (n, 1) is also a feasible exchange, though this chain has strictly lower weight than either of the 3-chains.

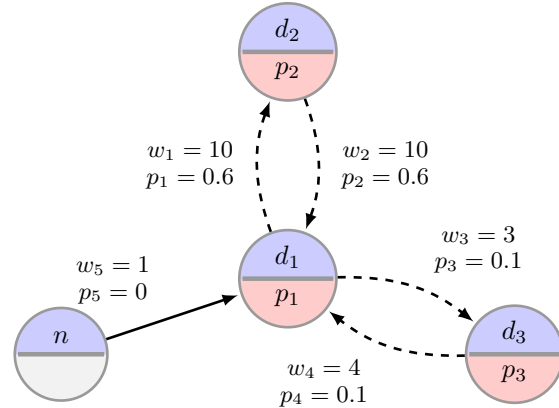


Figure 1: Example exchange graph with a single NDD n , and three patient-donor pairs; weights w and failure probabilities p are shown for each edge. The max-weight matching is the cycle between pairs 1 and 2; the max-expected-weight matching is the cycle between pairs 1 and 3, and the risk-averse/robust optimal matching is any the chain beginning with the edge from n to pair 1.

of cycle (1, 2). Of course, cycle (1, 3) has a significant ($\sim 10\%$) chance of failure, which may be unacceptable in a real setting. Thus, we might choose an objective that aims to maximize the matching weight under the *worst-case* outcome (e.g., as in McElfresh et al. [2019]). In this case, any chain beginning with edge $(n, 1)$ is optimal.

Next we describe our approach, beginning with a characterization of the *expected* matching weight.

3 MAXIMIZING EXPECTED MATCHING WEIGHT

We are primarily interested in maximizing the *expected* weight of a matching; indeed this is the focus of most prior work (see § 1.1). We refer to this as the *stochastic* KEP. First we characterize the objective of this problem—the expected matching weight. With known edge failure probabilities, the expected weight of a cycle or chain is expressible in closed form.

Discounted weight of a cycle. The discounted weight of a k -cycle c reflects the fact that the *whole* cycle will fail if any single transplant fails. We use w_e to denote the weight of edge e in the cycle, c .

$$u(c) = \left(\sum_{e \in c} w_e \right) \left[\prod_{e \in c} (1 - p_e) \right].$$

Discounted weight of a chain. The expected weight $u(\kappa)$ of the k -chain $\kappa \equiv (v_1, \dots, v_{k+1})$, where v_1 is a

Table 1: Comparison of stochastic and robust approaches to kidney exchange, which use a setting comparable to ours.³ Columns indicate the type of uncertainty considered in the problem (“Uunct.”: stochastic or worst-case/robust), whether or not edge failure probability is assumed to be *homogeneous* (“Homog.”), and the number of variables and constraints in each formulation.

Formulation	Uunct.	Homog.	# Vars.	# Constr.
PC-TSP [Anderson et al., 2015]	None	N/A	$O(E \cdot V + V ^2 + C)$	$O(V \cdot (E + 2^{ V } + C))$
PICEF [Dickerson et al., 2016]	Stoch.	Yes	$O(L \cdot E + C)$	$O(L \cdot V + L \cdot E + C)$
ROBUST [McElfresh et al., 2019]	Robust	N/A	$O(E \cdot V + V ^2 + C)$	$O(E \cdot V + V ^2 + C)$
DPS-18 [Dickerson et al., 2018]	Stoch.	No	$O(V ^L + C)$	$O(V)$
Our model (6)	Stoch.	No	$O(L \cdot E + C)$	$O(L \cdot V + L \cdot E + C)$

non-directed donor (NDD), is defined as

$$u(\kappa) = \sum_{i=2}^k p_i \left(\sum_{j=1}^{i-1} w_j \right) \prod_{j=1}^{i-1} (1 - p_j) + \left(\sum_{i=1}^k w_i \right) \prod_{i=1}^k (1 - p_i). \quad (1)$$

In the above, p_i and w_i denotes the failure probability and weight of edge (v_i, v_{i+1}) , respectively. The first term above is the sum of expected weights for the chain executing exactly $i-1 = \{1, \dots, k-1\}$ steps and then failing on the i th step. The second term is the resulting weight if the chain executes completely.

Using the above expressions, we can write the stochastic KEP as follows. With some abuse of notation, let $(\mathcal{C}, \mathcal{K}) \in \mathcal{M}$ denote a feasible matching consisting of cycles \mathcal{C} and chains \mathcal{K} . Problem 2 is an equivalent formulation of the stochastic KEP.

$$\max_{(\mathcal{C}', \mathcal{K}') \in \mathcal{M}} \sum_{c' \in \mathcal{C}'} u(c') + \sum_{\kappa' \in \mathcal{K}'} u(\kappa') \quad (2)$$

Next we describe our solution approach for Problem 2, and an equivalent compact mixed-integer linear program formulation.

3.1 Compact Formulation for Maximizing Expected Matching Weight

Here we present a new compact formulation to maximize the expected weight in the case of *non-identical* edge failure probabilities. *Compact* means that the counts of variables and constraints are polynomial in the size of the input. We compare the size of this model with other state-of-the-art approaches in § 3.2.

In [Dickerson et al., 2018], the authors propose a solution approach for Problem 2, which enumerates all feasible cycles and chains in the graph. However the number of cycles and chains grows exponentially with the size of

the graph, meaning this formulation is not compact. Further, it is intractable to even write this model in memory for large exchanges or long chain lengths.

We propose an exact, compact representation for Problem 2, using an equivalent expression for expected chain weight $u(\kappa)$ given in Lemma 1.

Lemma 1. *The discounted weight $u(\kappa)$ of the k -chain $\kappa = (v_1, \dots, v_{k+1})$ is*

$$u(\kappa) = \sum_{i=1}^k w_i \prod_{j=1}^i (1 - p_j),$$

where w_i and p_i are the edge weight and failur probability of the i^{th} edge, (v_i, v_{i+1}) , in the chain, for $i = 1, \dots, k$.

In other words, the discounted weight of a chain can be expressed as the sum of the “discounted weights” of each edge in the chain, i.e. $u(\kappa) = \sum_{i=1}^k w'_i$, where $w'_i \equiv w_i \prod_{j=1}^i (1 - p_j)$, where we refer to $\prod_{j=1}^i (1 - p_j)$ as the *discount factor*.

The objective of Optimization (4) uses Lemma 1 to express the total discounted weight of all matched cycles and chains, assuming non-uniform edge failure probabilities. This is achieved using two sets of variables, o_{ek} (the discount factor of edge e at position k in a chain) and v_c (the success probability of cycle c). Optimization (4) uses the following parameters:

- G : kidney exchange graph, consisting of edges $e \in E$ and vertices $v \in V = P \cup N$, including patient-donor pairs P and non-directed donors (NDDs) N
- \mathcal{C} : a set of cycles on exchange graph G
- L : chain cap (max. number of edges in a chain)
- w_e : edge weights for each edge $e \in E$
- w_c : cycle weights for each cycle $c \in \mathcal{C}$, defined as $w_c = \sum_{e \in c} w_e$

- $\delta^-(i)$: the set of edges into vertex i
- $\delta^+(i)$: the set of edges out of vertex i
- p_e : failure probability for edge $e \in E$

Edges between an NDD $n \in N$ and a patient-donor vertex $d \in P$ may only take position 1 in a chain, while edges between two patient-donor pairs may take any position $2, \dots, L$ in a chain. For convenience, we define the function \mathcal{K} for each edge e , such that $\mathcal{K}(e)$ is the set of all possible positions that edge e may take in a chain.

$$\mathcal{K}(e) = \begin{cases} \{1\}, & e \text{ begins in } n \in N, \\ \{2, \dots, L\}, & e \text{ begins in } d \in P. \end{cases} \quad (3)$$

The following decision variables are used.

- $z_c \in \{0, 1\}$: 1 if cycle c is used in the matching, and 0 otherwise
- $y_{ek} \in \{0, 1\}$: 1 if edge e is used at position k in a chain, and 0 otherwise
- $o_{ek} \in [0, 1]$: discount factor of edge e at position k in a chain

Our formulation is given in (4).

$$\max_{\mathbf{y}, \mathbf{z}, \mathbf{o}} \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e y_{ek} o_{ek} + \sum_{c \in C} w_c z_c v_c \quad (4a)$$

$$s.t. \quad \{\mathbf{y}, \mathbf{z}\} \in \mathcal{X}, \quad (4b)$$

$$\sum_{\substack{e \in \delta^-(i) \wedge \\ k \in \mathcal{K}(e)}} o_{ek} y_{ek} \geq \sum_{e \in \delta^+(i)} \frac{o_{e,k+1} y_{e,k+1}}{1 - p_e}, \quad (4c)$$

$$i \in P, k \in \{1, \dots, L-1\}, \quad (4c)$$

$$0 \leq o_{ek} \leq 1 - p_e, e \in E, k \in \mathcal{K}(e), \quad (4d)$$

$$v_c = \prod_{e \in c} (1 - p_e), c \in C, \quad (4e)$$

where \mathcal{X} denotes the set of feasible decision variables for the PICEF formulation of kidney exchange [Dickerson et al., 2016], defined as

$$\mathcal{X} = \begin{cases} \sum_{e \in \delta^-(i)} \sum_{k \in \mathcal{K}(e)} y_{ek} + \sum_{\substack{c \in C: \\ i \in c}} z_c \leq 1, & i \in P; \\ \sum_{e \in \delta^-(i) \wedge \\ k \in \mathcal{K}(e)} y_{ek} \geq \sum_{e \in \delta^+(i)} y_{e,k+1}, & i \in P, \\ & k \in \{1, \dots, L-1\}; \\ \sum_{e \in \delta^+(i)} y_{e1} \leq 1, & i \in N; \\ y_{ek} \in \{0, 1\}, & e \in E, k \in \mathcal{K}(e); \\ z_c \in \{0, 1\}, & c \in C; \end{cases} \quad (5)$$

The constraints of \mathcal{X} are interpreted as follows: 1) the first constraint in (5) requires that each patient-donor vertex $i \in P$ may only participate in one cycle or one chain; 2) the second constraint requires that each patient-donor vertex $i \in P$ can only have an outgoing edge at position $k+1$ in a chain if it has an incoming edge at position k ; 3) the third constraint requires that each NDD $i \in N$ may only participate in one chain.

Constraints (4c), (4d), and (4e) define the discounted weight of chains and cycles. We briefly describe how the discounted weight of cycles and chains are represented in this formulation:

- For a *cycle*, the success probability is $v_c = \prod_{e \in c} (1 - p_e)$. Thus the discounted weight of all cycles is expressed as $\sum_{c \in C} w_c z_c v_c$.

- For a *chain*, the discounted weight is expressed using Lemma 1. Consider the following example: suppose a k -chain consists of edges e_1, \dots, e_k . Suppose that i is the *first* patient-donor pair in this chain—so e_1 is the edge *into* i , and e_2 is the edge *out of* i ; that is, $e_1 \in \delta^-(i)$ and $e_2 \in \delta^+(i)$. From constraints (4c) we have $o_{e_1,1} \geq \frac{1}{1-p_{e_2}} o_{e_2,2}$ for vertex i . The sums in constraint (4c) contain no other terms, because \mathcal{X} requires that only one edge into vertex i and one edge out of vertex i can be matched. Therefore, $(1 - p_{e_2}) o_{e_1,1} \geq o_{e_2,2}$.

Similarly, $(1 - p_{e_{j+1}}) o_{e_j,j} \geq o_{e_{j+1},j+1}$ for $j = 2, \dots, k-1$. Since Optimization (4) is a maximization problem, the optimal values of variables $o_{e,j}$ will satisfy $o_{e_j,j} = \prod_{i=1}^j (1 - p_{e_i})$, for $1 \leq j \leq k$. Accordingly, $\sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e y_{ek} o_{ek}$ represents the total discounted weight of all chains according to Lemma 1.

3.2 MIP Reformulation of Optimization (4)

Although Optimization (4) exactly maximizes expected edge weight under non-identical edge failure probabilities, it is a nonconvex optimization problem. In this section, we reformulate it as a mixed-integer linear program which can be solved using general-purpose solvers. Proposition 1 concludes our results; the main idea is to define a set of new variables O_{ek} to replace $y_{ek} o_{ek}$ in Optimization (4).

Proposition 1. *Optimization (4) is equivalent to*

$$\begin{aligned} \max_{\mathbf{y}, \mathbf{z}, \mathbf{O}, \mathbf{o}} \quad & \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e O_{ek} + \sum_{c \in C} w_c z_c \left(\prod_{e \in c} (1 - p_e) \right) \\ s.t. \quad & \{\mathbf{y}, \mathbf{z}\} \in \mathcal{X}, \\ & \{\mathbf{y}, \mathbf{O}, \mathbf{o}\} \in \mathcal{X}', \end{aligned} \quad (6)$$

where \mathcal{X} follows the definition in (4), and \mathcal{X}' is defined as

$$\mathcal{X}' = \left\{ \begin{array}{l} \sum_{e \in \delta^-(i) \wedge k \in \mathcal{K}(e)} O_{ek} \geq \sum_{e \in \delta^+(i)} \frac{O_{e,k+1}}{1-p_e}; \\ i \in P, k \in \{1, \dots, L-1\}, \\ O_{ek} \leq y_{ek}, e \in E, k \in \mathcal{K}(e); \\ O_{ek} \leq o_{ek}, e \in E, k \in \mathcal{K}(e); \\ O_{ek} \in [0, 1], e \in E, k \in \mathcal{K}(e); \\ 0 \leq o_{ek} \leq 1 - p_e, e \in E, k \in \mathcal{K}(e). \end{array} \right\} \quad (7)$$

Appendix B gives an explicit example of how to use the reformulation in Proposition 1 to model a kidney exchange graph. Optimization (6) can be solved using standard solvers such as CPLEX and Gurobi.

Scalability

We compare our model size with state-of-the-art approaches in literature. We summarize all approaches in Table 1. The size of each model (the number of variables and constraints) is expressible in terms of the chain cap L , and the number of edges ($|E|$), cycles ($|C|$), total vertices ($|V|$), NDD vertices ($|N|$), and patient-donor pair vertices $|P|$. For ease of exposition we assume $|N| = O(|V|)$ and $|P| = O(|V|)$.

Our size is comparable with PICEF, while accounting for non-identical failure probabilities. DPS-18 [Dickerson et al., 2018] considers non-identical failure probabilities at the cost of representing every single chain and cycle as a decision variable, and thus this model grows exponentially with the chain cap L ; in contrast, the number of variables in our formulation is polynomial in L . Real exchanges often use a cycle cap of 3, which is sufficiently small that all cycles can be enumerated in practice—even on realistic graphs with hundreds of vertices. If exchanges grow much larger in the future (e.g., thousands of vertices), or if cycle lengths are increased substantially, we further propose a branch-and-price implementation to solve the corresponding problems brought by huge $|C|$ in Appendix D.

4 EXTENSIONS TO MEAN-RISK KIDNEY EXCHANGE MODEL

Next we introduce a kidney exchange model which balances both the *mean expected weight* and the *worst-case weight* (“risk”) of a matching, using known non-identical edge failure probabilities. We achieve this balance using a conditional value-at-risk (CVaR) objective. This approach is motivated by the fact that the *expected weight* of a matching can be misleading when the *worst-case* outcome can be arbitrarily bad (see § 2.1). This is especially true in kidney exchange, where a single edge failure can impact an entire cycle or chain.

4.1 Mean-risk Model

At a high level, the CVaR objective for kidney exchange is expressed as

$$\mu + \gamma \times \mu_\alpha,$$

where μ is the expected matching weight and μ_α is the $\alpha \times 100\%$ ($\alpha \in (0, 1]$) *worst-case* mean weight—that is, the mean matching weight in the worst $\alpha \times 100\%$ of all outcomes. The parameter γ is set by the user, and controls the trade-off between average performance and the *risk* of the solution.

For tractability and simplicity, we define \mathbf{W} as an $|E|$ -dimensional vector with

$$W_e = - \sum_{k \in \mathcal{K}(e)} y_{ek} - \sum_{c \in C} \mathbf{1}(e \in c) z_c, \quad \forall e \in E.$$

That is, $W_e = -1$ if edge e is used, and $W_e = 0$ otherwise. We use $\mathbf{w} \in \mathbb{R}^{|E|}$ to represent the random discounted edge weights under known edge failure probabilities. Correspondingly, $\langle \mathbf{w}, \mathbf{W} \rangle$ represents the loss (negative weight) of a matching. The $\alpha \times 100\%$ worst-case (highest) mean loss is equivalent to the CVaR objective [Rockafellar et al., 2000] at level α . The corresponding optimization problem is expressed in (8), by introducing an auxiliary variable d . We use $(x)^+$ to denote $\max(0, x)$, and the expectation in (8) is taken over the distribution of random edge weights under the known edge failure probabilities. As before, \mathcal{X} denotes the set of feasible matchings using the PICEF formulation.

$$\begin{array}{ll} \min_{\mathbf{y}, \mathbf{z}, d} & \mathbb{E}(\langle \mathbf{w}, \mathbf{W} \rangle) + \gamma \left[d + \frac{1}{\alpha} \mathbb{E} [(\langle \mathbf{w}, \mathbf{W} \rangle - d)^+] \right] \\ \text{s.t.} & \{\mathbf{y}, \mathbf{z}\} \in \mathcal{X}. \end{array} \quad (8)$$

4.2 An SAA-based Approach for Optimization (8)

The main difficulty in solving Optimization (8) is that term $\mathbb{E} [(\langle \mathbf{w}, \mathbf{W} \rangle - d)^+]$ does not have a simple closed-form reformulation under the known edge failure probabilities. Thus, we propose an approach based on Sample Average Approximation (SAA) [Anderson et al., 2015] to solve (8). The main idea is to first sample N realizations of edge existence according to the known edge failure probabilities; for each realization we formulate a mixed-integer linear program representing the matching weight under this realization. Finally, we combine all N models to obtain an optimization problem that is (approximately) equivalent to Optimization (8) based on these N realizations. Algorithm 1 gives a pseudocode description of this approach.

Algorithm 1 SAA

- 1: **Initialization:** N ;
 - 2: **STEP 1:**
 - 3: Sample N edge existence realizations $\{\hat{r}_{en} \in \{0, 1\}, \forall e \in E, n = 1, \dots, N\}$;
 - 4: **STEP 2:**
 - 5: Solve Optimization (9).
-

This algorithm has only two steps: first it samples N *realizations* of edge existence from the known edge failure probabilities, where \hat{r}_{en} is 1 if edge e succeeds in realization n , and 0 if it fails. These realization variables are used as input to Optimization (9), which uses decision variables $\hat{\mathbf{W}}_n$ to represent the *realized* edge discount factor for realization n —that is, \hat{W}_{en} is 1 if edge e is matched and succeeds in realization n and 0 otherwise (see Appendix C for details). Using these decision variables, the objective of Optimization (9) includes two terms: the mean matching weight, and the CVaR objective—both approximated using all N samples (i.e., the sample-average approximation). Thus, Optimization (9) represents the SAA of (8) under the N sampled realizations.

Proposition 2. *Optimization (9) is equivalent to the SAA of (8) under N edge existence realizations represented by \hat{r}_{en} , with*

$$\begin{aligned}
 \min \quad & \frac{1}{N} \sum_{n=1}^N \langle \mathbf{w}, \hat{\mathbf{W}}_n \rangle + \gamma \left(d + \frac{1}{\alpha} \frac{1}{N} \sum_{n=1}^N \Pi_n \right) \\
 \text{s.t.} \quad & \hat{W}_{en} = - \sum_{k \in \mathcal{K}(e)} O_{ekn} - \sum_{c \in C} \mathbf{1}(e \in c) z_c v_{cn}, \forall e, n, \\
 & \Pi_n \geq 0, \forall n, \\
 & \Pi_n \geq \langle \mathbf{w}, \hat{\mathbf{W}}_n \rangle - d, \forall n, \\
 & \{\mathbf{y}, \mathbf{z}\} \in \mathcal{X}, \\
 & \{\mathbf{y}, \mathbf{z}\} \in \mathcal{X}', \\
 & o_{ekn} \leq \hat{r}_{en}, \forall e, k, n, \\
 & v_{cn} = \min_{e \in c} \{\hat{r}_{en}\}, \forall c, n,
 \end{aligned} \tag{9}$$

where \mathcal{X} follows the definition in (4), and \mathcal{X}' is defined as

$$\left\{ \begin{array}{l}
 \sum_{e \in \delta^-(i) \wedge k \in \mathcal{K}(e)} O_{ekn} \geq \sum_{e \in \delta^+(i)} O_{e,k+1,n}, \\
 \forall i \in P, k \in \{1, \dots, L-1\}, n \in \mathcal{N}; \\
 O_{ekn} \leq y_{ek}, e \in E, k \in \mathcal{K}(e), n \in \mathcal{N}; \\
 O_{ekn} \leq o_{ekn}, e \in E, k \in \mathcal{K}(e), n \in \mathcal{N}; \\
 o_{ekn}, O_{ekn} \in [0, 1], e \in E, k \in \mathcal{K}(e), n \in \mathcal{N}; \\
 \mathcal{N} = \{1, \dots, N\}.
 \end{array} \right.$$

Optimization (9) can be understood by viewing $\langle \mathbf{w}, \hat{\mathbf{W}}_n \rangle$ as the realized edge weight under the n -th realization with matching $\{\mathbf{y}, \mathbf{z}\}$.

5 COMPUTATIONAL EXPERIMENTS

First, we benchmark our tractable model for non-identical edge failure probabilities (6) (“KEP-NP”) against previous approaches, with the *stochastic* (i.e., max-expected-weight) objective. We find that our approach outperforms two leading previous methods: PICEF without edge failure probabilities [Dickerson et al., 2016] (“KEP”), and PICEF with identical edge failure probability [Dickerson et al., 2018] (“KEP-IP”). Second, we compare our CVaR model (9) (“CVAR”) against KEP, KEP-IP, and KEP-NP; to our knowledge, there are no other tractable approaches using the CVaR objective in our setting. Finally, we briefly present the running time of all implemented approaches.

5.1 Stochastic Objective

We use two sets of 32 randomly-generated graphs, one with 64 nodes each and one with 128 nodes each. These graphs resemble the structure of real exchanges, and are generated using anonymized data from the United Network for Organ Sharing (UNOS) US-wide kidney exchange.⁴ We simulate edge existence uncertainty by randomly assigning each edge in each graph a failure probability, independently uniformly distributed on $[0.1, 0.9]$; for simplicity, we set all edge weights to 1. We use cycles of length 2 and 3, and chains up to length 4—which are the standard limits in fielded exchanges (including UNOS). For KEP-IP, we assume $p_e = 0.5$ for all edges (the correct mean edge failure probability). For each random exchange graph we first find the optimal matching according to each approach (KEP, KEP-IP, and KEP-NP). We then generate 200 *realizations* of the exchange graph, according to each edge’s (randomly generated) failure probability. We then calculate the *realized* weight of the optimal matching for each method, accounting for failed edges (cycles with any failed edges receive zero weight, and chains only receive weight for consecutive successful edges, beginning with the first). We also calculate the *omniscient* matching for each realization, i.e., the maximum matching weight *after* observing edge failures.

Metric: Percentage of Omniscient Weight. We compare all approaches against the omniscient matching weight, which is a strict upper bound on performance for any matching approach. Let W_{OPT} be the omniscient-optimal matching weight for a particular exchange, and a particular realization; let W_M be the realized matching weight for a non-omniscient method. We calculate the percentage of W_{OPT} achieved by each matching method, for a particular realization, as $\%OPT \equiv$

⁴See <https://optn.transplant.hrsa.gov/data> for more information on UNOS.

$100 \times W_M / W_{OPT}$. Figure 2 (left column) shows %OPT for all exchange graphs, over all 200 realizations, for 64-node graphs (top) and 128-node graphs (bottom). Our method (KEP-NP) improves expected matching weight compared to previous methods KEP and KEP-IP.

5.2 CVaR Objective

We implement CVAR (§ 5.1) using $N = 10$ simulated edge realizations, with $\gamma = 10$, and $\alpha = 0.5$.

Metric: $\alpha\%$ Worst-Case Mean. CVAR is designed to maximize the $\alpha\%$ worst-case mean matching weight; thus, we use this metric for each matching approach. For each graph, and each matching approach, we calculate the mean of the $\alpha\%$ lowest realized matching weights (over all 200 realizations). We compare each method to KEP, which assumes $p_e = 0$. Let μ_P^α be the $\alpha\%$ lowest-realized matching weights for KEP, and let μ_M^α be the same, for a different matching approach; we calculate a ratio as follows: $\Delta\alpha\% \equiv 100 \times (\mu_M^\alpha - \mu_P^\alpha) / \mu_P^\alpha$. Figure 2 (middle column) shows $\Delta\alpha\%$ for all 32 exchange graphs. CVAR clearly improves the $\alpha\%$ worst-case mean matching weight, over other methods (including our new formulation for inhomogeneous edge failure probabilities, KEP-NP).

Timing Figure 2 (right column) shows solver time required for each method. Our new formulation (KEP-NP), requires nearly the same runtime as KEP (a deterministic PICEF model). As expected CVAR requires more time—and it increases with the number of samples (N).

6 CONCLUSION

In fielded kidney exchanges, planned transplants *fail* for a variety of reasons. Due to the cycle- and chain-like swaps used by exchanges, a single failed transplant can “cascade” through an exchange, causing several other transplants to fail. These failures are common (UNOS estimates that about 85% of its planned transplants fail [Leishman, 2019]); failures cause patients to face longer waiting times, and incur the additional costs and burden of dialysis.

We consider a setting where the failure probability of each potential transplant (*edge*) is known, and the *kidney exchange clearing problem* is to select a set of transplants that maximize a mathematical objective subject to this uncertainty. The choice of objective is important, particularly in kidney exchange: a *deterministic* approach (which ignores potential failures) may naïvely select long cycles or chains, which have high likelihood of failure. On the other hand, a *robust* approach (which pro-

tections against the worst-case outcome) is often too conservative, because in kidney exchange, the worst-case outcome is often that *all transplants* fail. We consider two objectives: maximizing the *expected* weight, and maximizing the conditional value-at-risk (CVaR). We are not the first to investigate these objectives in this setting. However, state-of-the-art approaches either assume that all edges have identical failure probabilities, or their algorithms scale exponentially in the size of the input—and are intractable for realistic exchanges.

We propose the first *scalable* approaches for kidney exchange with non-identical edge failure probabilities, for both the stochastic and CVaR objectives. For the max-expected weight objective our approach is exact, and clearly outperforms prior approaches that assume identical edge failure probabilities—with marginally longer runtime. For the CVaR objective we use a sample-average-approximation-based method, which outperforms comparable state-of-the-art approaches, even with a small number of samples. We formulate both of our approaches as mixed integer linear programs, which are solvable with off-the-shelf commercial solvers such as CPLEX or Gurobi.

There are several areas for future work. Our model assumes perfect knowledge of edge failure probabilities—while in reality only rough estimates of these probabilities are available. Furthermore, slight over- or under-estimation of these probabilities can impact the matching weight [Dickerson et al., 2018]—something we did not address in this work.

We emphasize that the choice of objective is important in kidney exchange, as different objectives (or a different weighting of multiple objectives) can drastically change the outcome. Before implementing any of these approaches, it is necessary to understand the priorities of the relevant stakeholders, their appetite for risk, and whether these priorities align with our mathematical objectives [Freedman et al., 2020, Noothigattu et al., 2018].

Finally, each of the approaches discussed in this paper may negatively impact some exchange participants. For example, *highly-sensitized* patients are often sicker and harder to match than other patients; transplants involving highly-sensitized patients are thus often riskier than other transplants. A risk-averse or stochastic objective function would likely de-prioritize highly-sensitized patients, ignoring them for lower-risk matches. Thus, new objective functions and other modeling choices will likely raise concerns of *fairness* for different patients, or groups of patients, within an exchange [McElfresh and Dickerson, 2018, Roth et al., 2005b, Yilmaz, 2011].

Acknowledgments. Dickerson and McElfresh were sup-

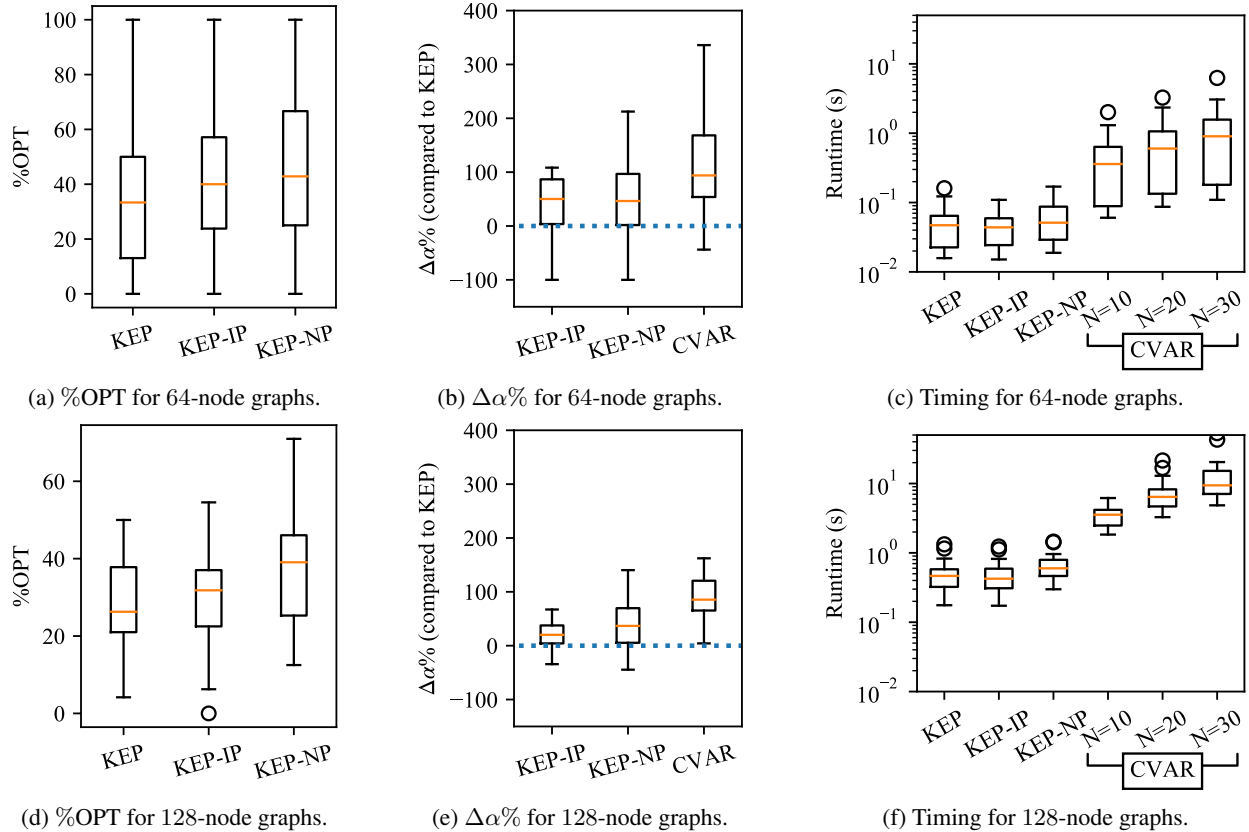


Figure 2: Boxplots of %OPT (left column), $\Delta\alpha\%$ (center column), and timing (right column) for each matching approach, over 32 random graphs with 64 nodes (top row) and 128 nodes (bottom row). The horizontal line at the center of each box plot indicates the median; the upper and lower edges of the box indicate the first and third quartiles; the whiskers extend 1.5 times the interquartile range beyond quartile 1 and 3.

ported in part by NSF CAREER IIS-1846237, DARPA GARD #HR112020007, DARPA SI3-CMD #S4761, DoD WHS #HQ3420F0035, NIH R01 NLM-013039-01, and a Google Faculty Research Award.

References

- David Abraham, Avrim Blum, and Tuomas Sandholm. Clearing algorithms for barter exchange markets: Enabling nationwide kidney exchanges. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 295–304, 2007.
- Nikhil Agarwal, Itai Ashlagi, Eduardo Azevedo, Clayton R Featherstone, and Ömer Karaduman. Market failure in kidney exchange. *American Economic Review*, 109(11):4026–70, 2019.
- Filipe Alvelos, Xenia Klimentova, Abdur Rais, and Ana Viana. A compact formulation for maximizing the expected number of transplants in kidney exchange programs. In *Journal of Physics: Conference Series*, volume 616. IOP Publishing, 2015.
- Ross Anderson, Itai Ashlagi, David Gamarnik, and Alvin E Roth. Finding long chains in kidney exchange using the traveling salesman problem. *Proceedings of the National Academy of Sciences*, 112(3):663–668, 2015.
- Valentin Bartier, Bart Smeulders, Yves Crama, and Frits CR Spieksma. Recourse in kidney exchange programs, 2019. Working paper.
- Péter Biró, Bernadette Haase-Kromwijk, Tommy Andersson, Eyjólfur Ingi Ásgeirsson, Tatiana Baltsová, Ioannis Boletis, Catarina Bolotinha, Gregor Bond, Georg Böhmig, Lisa Burnapp, et al. Building kidney exchange programmes in europe—an overview of exchange practice and activities. *Transplantation*, 103(7):1514, 2019.
- Avrim Blum, Anupam Gupta, Ariel D. Procaccia, and Ankit Sharma. Harnessing the power of two cross-matches. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 123–140, 2013.
- Avrim Blum, John P. Dickerson, Nika Haghtalab,

- Ariel D. Procaccia, Tuomas Sandholm, and Ankit Sharma. Ignorance is almost bliss: Near-optimal stochastic matching with few queries. In *Proceedings of the ACM Conference on Economics and Computation (EC)*, pages 325–342, 2015.
- Margarida Carvalho, Xenia Klimentova, Kristiaan Glorie, Ana Viana, and Miguel Constantino. Robust models for the kidney exchange problem. *INFORMS Journal on Computing*, 2020. To appear.
- Miguel Constantino, Xenia Klimentova, Ana Viana, and Abdur Rais. New insights on integer-programming models for the kidney exchange problem. *European Journal of Operational Research*, 231(1):57–68, 2013.
- John P. Dickerson and Tuomas Sandholm. Multi-organ exchange. *Journal of Artificial Intelligence Research*, 60:639–679, 2017.
- John P. Dickerson, David Manlove, Benjamin Plaut, Tuomas Sandholm, and James Trimble. Position-indexed formulations for kidney exchange. In *Proceedings of the ACM Conference on Economics and Computation (EC)*, 2016.
- John P. Dickerson, Ariel D. Procaccia, and Tuomas Sandholm. Failure-aware kidney exchange. *Management Science*, 2018. To appear; earlier version appeared at EC-13.
- Haluk Ergin, Tayfun Sönmez, and M Utku Ünver. Multi-donor organ exchange, 2017. Working paper.
- Rachel Freedman, Jana Schach Borg, Walter Sinnott-Armstrong, John P. Dickerson, and Vincent Conitzer. Adapting a kidney exchange algorithm to align with human values. *Artificial Intelligence (AIJ)*, 283: 103261, 2020.
- Kristiaan M. Glorie, J. Joris van de Klundert, and Albert P. M. Wagelmans. Kidney exchange with long chains: An efficient pricing algorithm for clearing barter exchanges with branch-and-price. *Manufacturing & Service Operations Management (MSOM)*, 16(4):498–512, 2014.
- Noam Goldberg and Michael Poss. Maximum probabilistic all-or-nothing paths. *European Journal of Operational Research*, 2019.
- Xenia Klimentova, João Pedro Pedroso, and Ana Viana. Maximising expectation of the number of transplants in kidney exchange programmes. *Computers & Operations Research*, 73:1–11, 2016.
- Ruthanne Leishman. Challenges in match offer acceptance in the OPTN kidney paired donation pilot program. Presentation at the INFORMS Annual Meeting, 2019. Head of UNOS (US-wide kidney paired donation program).
- David Manlove and Gregg O’Malley. Paired and altruistic kidney donation in the UK: Algorithms and experimentation. *ACM Journal of Experimental Algorithms*, 19(1), 2015.
- Duncan McElfresh, Hoda Bidkhori, and John P. Dickerson. Scalable robust kidney exchange. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2019.
- Duncan C. McElfresh and John P. Dickerson. Balancing lexicographic fairness and a utilitarian objective with application to kidney exchange. *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.
- Duncan C McElfresh, Michael Curry, Tuomas Sandholm, and John P Dickerson. Improving policy-constrained kidney exchange via pre-screening. *arXiv preprint*, 2020.
- Ritesh Noothigattu, Snehal Kumar Neil S. Gaikwad, Edmond Awad, Sohan D’Souza, Iyad Rahwan, Pradeep Ravikumar, and Ariel D. Procaccia. A voting-based system for ethical decision making. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.
- F. T. Rapaport. The case for a living emotionally related international kidney donor exchange registry. *Transplantation Proceedings*, 18:5–9, 1986.
- R Tyrrell Rockafellar, Stanislav Uryasev, et al. Optimization of conditional value-at-risk. *Journal of risk*, 2: 21–42, 2000.
- Alvin Roth, Tayfun Sönmez, and Utku Ünver. Kidney exchange. *Quarterly Journal of Economics*, 119(2): 457–488, 2004.
- Alvin Roth, Tayfun Sönmez, and Utku Ünver. A kidney exchange clearinghouse in New England. *American Economic Review*, 95(2):376–380, 2005a.
- Alvin Roth, Tayfun Sönmez, and Utku Ünver. Pairwise kidney exchange. *Journal of Economic Theory*, 125(2):151–188, 2005b.
- UNOS. United Network for Organ Sharing (UNOS). <http://www.unos.org/>.
- Wen Wang, Mathieu Bray, Peter XK Song, and John D Kalbfleisch. An efficient algorithm to enumerate sets with fallbacks in a kidney paired donation program. *Operations Research for Health Care*, 20:45–55, 2019.
- Özgür Yılmaz. Kidney exchange: An egalitarian mechanism. *Journal of Economic Theory*, 146(2):592–618, 2011.
- Qipeng P Zheng, Siqian Shen, and Yuhui Shi. Loss-constrained minimum cost flow under arc failure uncertainty with applications in risk-aware kidney exchange. *IIE Transactions*, 47(9):961–977, 2015.

A Proof of Lemma 1

Proof. The expected discounted weight of a chain with k edges is expressed as

$$u(k) = \sum_{i=2}^k p_i \left(\sum_{j=1}^{i-1} w_j \right) \prod_{j=1}^{i-1} (1 - p_j) \\ + \left(\sum_{i=1}^k w_i \right) \prod_{i=1}^k (1 - p_i).$$

The coefficient on weight w_i (the i^{th} edge in the chain), for any $1 \leq i \leq k$, is expressed as $\prod_{j=1}^i (1 - p_j)$. Thus,

$$u(k) = \sum_{i=1}^k w_i \prod_{j=1}^i (1 - p_j).$$

□

B Operationalizing Proposition 1

In this section, we give an explicit instantiation of the mixed-integer linear program of Proposition 1. We provide an example here for the compatibility graph shown in Figure 1, where there are two cycles, five edges, and several feasible chains. Suppose cycle 1 contains Edge 1 and 2, and cycle 2 contains Edge 3 and 4. Additionally, the chain capacity is $L = 2$ in Figure 1.

$$\begin{aligned} \max_{\mathbf{y}, \mathbf{z}, \mathbf{O}, \mathbf{o}} \quad & \sum_{e=1}^4 w_e O_{e2} + w_5 O_{51} \\ & + (w_1 + w_2) z_1 (1 - p_1) (1 - p_2) \\ & + (w_3 + w_4) z_2 (1 - p_3) (1 - p_4) \\ \text{s.t.} \quad & y_{51} + y_{22} + y_{42} + z_1 + z_2 \leq 1, \\ & y_{12} + z_1 \leq 1, \\ & y_{32} + z_2 \leq 1, \\ & y_{51} \geq y_{12} + y_{32}, \\ & y_{51} \leq 1, \\ & y_{e2} \in \{0, 1\}, e \in \{1, \dots, 4\}, \\ & y_{51} \in \{0, 1\}, \\ & z_c \in \{0, 1\}, c \in \{1, 2\}, \\ & O_{51} \geq \frac{O_{12}}{1 - p_1} + \frac{O_{32}}{1 - p_3}, \\ & O_{e2} \leq y_{e2}, e \in \{1, \dots, 4\}, \\ & O_{51} \leq y_{51}, \\ & O_{e2} \leq o_{e2}, e \in \{1, \dots, 4\}, \\ & O_{51} \leq o_{51}, \\ & O_{e2} \in [0, 1], e \in \{1, \dots, 4\}, \\ & O_{51} \in [0, 1], \\ & 0 \leq o_{e2} \leq 1 - p_e, e \in \{1, \dots, 4\}, \\ & 0 \leq o_{51} \leq 1 - p_5. \end{aligned} \tag{10}$$

C Optimization (8) under one realization of edge existence

Before showing the equivalence between the SAA of (8) and (9), we first obtain the objective value of (8) under one fixed realization of edge existence. The objective value of (8) is obtained in (11), where we assume the fixed realization is $r_e \in \{0, 1\}$, $e \in E$, where 1 means the edge exists, and 0 otherwise.

In (11), we use two sets of variables $o_{ek} \in \{0, 1\}$ and $v_c \in \{0, 1\}$, which indicate the validity of chains and cycles, respectively.

- For any cycle c , c is only valid ($v_c = 1$) if all edges in c exist. Therefore, we restrict $v_c = \min_{e \in c} \{r_e\}$.
- For any chain, an edge e at position k is only valid ($o_{ek} = 1$) if 1) this edge exists ($r_e = 1$) and 2) the prior edges in this chain are valid too. Therefore, we use $o_{ek} \leq r_e$ to guarantee this edge e exists. The constraint (11c) serves the goal to guarantee the prior edges are valid. To see this point, we consider the following example. Suppose edge $e_1 \in \delta^-(i)$ and $e_2 \in \delta^+(i)$. Both edges are selected in one chain with $y_{e_1, k} = 1$, $y_{e_2, k+1} = 1$. If edge e fails ($r_{e_1} = 0$), then $o_{e_1, k} = 0$ restricting $o_{e_2, k+1}$ to be zero too. Therefore, all the edges after position k in this chain will be invalid.

$$\min_{\mathbf{y}, \mathbf{z}, \mathbf{o}, \mathbf{v}, d} \quad \left(- \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e y_{ek} o_{ek} - \sum_{c \in C} w_c z_c v_c \right) \\ + \gamma \left[d + \frac{1}{\alpha} \left(- \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e y_{ek} o_{ek} - \sum_{c \in C} w_c z_c v_c - d \right)^+ \right] \tag{11a}$$

$$\text{s.t.} \quad \{\mathbf{y}, \mathbf{z}\} \in \mathcal{X}, \tag{11b}$$

$$\sum_{\substack{e \in \delta^-(i) \wedge \\ k \in \mathcal{K}(e)}} o_{ek} y_{ek} \geq \sum_{e \in \delta^+(i)} o_{e, k+1} y_{e, k+1}, \tag{11c}$$

$$i \in P, k \in \{1, \dots, L - 1\}, \tag{11d}$$

$$o_{ek} \leq r_e, e \in E, k \in \mathcal{K}(e), \tag{11e}$$

$$v_c = \min_{e \in c} \{r_e\}, c \in C; \tag{11e}$$

$$o_{ek} \in [0, 1], e \in E, k \in \mathcal{K}(e). \tag{11f}$$

Optimization (11) has a tractable reformulation as shown in Proposition 3.

Proposition 3. Optimization (11) is equivalent to

$$\begin{aligned}
\min_{\mathbf{y}, \mathbf{z}, \mathbf{o}, \mathbf{v}, d} \quad & \langle \mathbf{w}, \hat{\mathbf{W}} \rangle + \gamma \left[d + \frac{1}{\alpha} (\langle \mathbf{w}, \hat{\mathbf{W}} \rangle - d)^+ \right] \\
\text{s.t.} \quad & \{\mathbf{y}, \mathbf{z}\} \in \mathcal{X}, \mathcal{X}' \\
& \hat{W}_e = - \sum_{k \in \mathcal{K}(e)} O_{e,k} - \sum_{c \in C} \mathbf{1}(e \in c) z_c v_c, \forall e, \\
& o_{e,k} \leq r_e, \forall e, k, \\
& v_c = \min_{e \in c} \{r_e\}, \forall c.
\end{aligned} \tag{12}$$

where \mathcal{X}' is defined as

$$\mathcal{X}' = \left\{ \begin{array}{l} \sum_{e \in \delta^-(i) \wedge k \in \mathcal{K}(e)} O_{e,k} \geq \sum_{e \in \delta^+(i)} O_{e,k+1}, \\ i \in P, k \in \{1, \dots, L-1\}; \\ O_{e,k} \leq y_{ek}, e \in E, k \in \mathcal{K}(e); \\ O_{e,k} \leq o_{e,k}, e \in E, k \in \mathcal{K}(e); \\ o_{e,k}, O_{e,k} \in [0, 1], e \in E, k \in \mathcal{K}(e). \end{array} \right\}$$

By comparing Optimization (12) and (9), it is easy to see that the objective value of (9) equals the average realized weights of N realizations. Therefore, we get the conclusion that Optimization (9) is equivalent to the SAA of Optimization (8).

D A branch and price implementation

In this section, we present a method for scaling our model to graphs with high cycle capacities. Theoretically, the number of cycles of length at most M is $O(|P|^M)$, making explicit representation and enumeration of all cycles infeasible for large enough instances. To solve this problem, we propose a branch and price algorithm, which uses column generation to incrementally consider the possible cycles in a graph. Similar ideas in other kidney exchange problems have also been explored [Dickerson et al., 2016, Glorie et al., 2014]. We show that our formulation with non-identical failure probabilities also scales well with large cycle numbers.

The detailed procedure is introduced as follows; for convenience, we use a vector \mathbf{X} to denote the solution $\mathbf{X} = [\mathbf{y}, \mathbf{z}]$. First, we define a set \mathcal{X}_f that indicates the fixed components in the solution \mathbf{X} . For example, $\mathcal{X}_f = \{X_i = 0, X_j = 1\}$ means the i -th and j -th components in \mathbf{X} are fixed to 0 and 1, respectively. Our algorithm begins with $\mathcal{X}_f = \emptyset$. Next, an LP relaxation (13) based on a (random) subset of cycles C' ($C' \subset C$) is solved.

$$\max_{\mathbf{y}, \mathbf{z}, \mathbf{o}, \mathbf{v}} \quad \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e O_{ek} + \sum_{c \in C'} w_c z_c \left(\prod_{e \in c} 1 - p_e \right) \tag{13a}$$

$$\text{s.t.} \quad \sum_{e \in \delta^-(i)} \sum_{k \in \mathcal{K}(e)} y_{ek} + \sum_{c \in C': i \in c} z_c \leq 1, i \in P, \tag{13b}$$

$$\sum_{e \in \delta^-(i) \wedge k \in \mathcal{K}(e)} y_{ek} \geq \sum_{e \in \delta^+(i)} y_{e,k+1}, \tag{13c}$$

$$i \in P, k \in \{1, \dots, L-1\}, \tag{13d}$$

$$\sum_{e \in \delta^+(i)} y_{e1} \leq 1, i \in N, \tag{13e}$$

$$y_{ek} \in [0, 1], e \in E, k \in \mathcal{K}(e), \tag{13f}$$

$$z_c \in [0, 1], c \in C', \tag{13g}$$

$$\sum_{e \in \delta^-(i) \wedge k \in \mathcal{K}(e)} O_{ek} \geq \sum_{e \in \delta^+(i)} \frac{O_{e,k+1}}{1 - p_e}, \tag{13h}$$

$$i \in P, k \in \{1, \dots, L-1\}, \tag{13i}$$

$$O_{ek} \leq y_{ek}, e \in E, k \in \mathcal{K}(e), \tag{13j}$$

$$O_{ek} \leq o_{e,k}, e \in E, k \in \mathcal{K}(e), \tag{13k}$$

$$O_{ek} \in [0, 1], e \in E, k \in \mathcal{K}(e), \tag{13l}$$

$$0 \leq o_{ek} \leq 1 - p_e, e \in E, k \in \mathcal{K}(e). \tag{13m}$$

The following step is to find positive price cycles: cycles that have the potential to improve the objective value if included in the model. The price of a cycle c is defined as $[w_c \prod_{e \in c} (1 - p_e) - \sum_{i \in c} \lambda_i]$, where λ_i are the dual values corresponding to the constraints (13b). While there exist any positive price cycles, optimality of the reduced LP has not yet been proved. This can be evidenced from Proposition 4, which can be proved through the strong duality of linear programming.

Proposition 4. Suppose the dual variables corresponding to the constraints (13b) are $\lambda_i, i \in P$. Then the optimal $\lambda_i, i \in P$, satisfy

$$w_c \prod_{e \in c} (1 - p_e) - \sum_{i \in c} \lambda_i \leq 0.$$

Therefore, we incrementally add (one or more) cycles that have positive prices, i.e. $w_c \prod_{e \in c} (1 - p_e) - \sum_{i \in c} \lambda_i > 0$ into C' until no positive price cycles exist in C . Afterwards, if the optimal solutions of the relaxed LP, i.e. (13), are integral, then they are the desired optimal solutions. Otherwise, branching occurs by following the standard branch-and-bound tree search. For example, suppose the i -th component of \mathbf{X} is fractional, then we fix $X_i = 0$ or $X_i = 1$. We record these fixed components in set \mathcal{X}_f and repeat above procedures with the new set

\mathcal{X}_f . We conclude the above discussions in the following Algorithm 2. By running $\text{BranchAndPrice}(G, \emptyset)$, we obtain the optimal solution.

Algorithm 2 $\text{BranchAndPrice}(G, \mathcal{X}_f)$

- 1: Generate a subset $C' \subset C$;
 - 2: Solve LP relaxation (13) based on C' and fixed components in \mathcal{X}_f ;
 - 3: **while** $\max_{c \in C} w_c \prod_{e \in c} (1 - p_e) - \sum_{i \in c} \lambda_i > 0$ **do**
 - 4: Add $c^* = \arg \max_{c \in C} w_c \prod_{e \in c} (1 - p_e) - \sum_{i \in c} \lambda_i$ to C' ;
 - 5: **end while**
 - 6: $\mathbf{X} = [\mathbf{y}, \mathbf{z}] \leftarrow$ solve LP relaxation (13) based on C' and fixed components in \mathcal{X}_f ;
 - 7: **if** \mathbf{X} is fractional **then**
 - 8: Find fractional binary variable $X_i \in \mathbf{X}$ closest to 0.5;
 - 9: $\mathbf{X}_1 = \text{BranchAndPrice}(G, \mathcal{X}_f \cup X_i = 0)$;
 - 10: $\mathbf{X}_2 = \text{BranchAndPrice}(G, \mathcal{X}_f \cup X_i = 1)$;
 - 11: Return \mathbf{X}_1 or \mathbf{X}_2 that gives larger objective values in the original Problem (6).
 - 12: **else**
 - 13: Return \mathbf{X} .
 - 14: **end if**
-