ABSTRACT

Title of Dissertation: PRACTICAL ALGORITHMS FOR RESOURCE ALLOCATION AND DECISION MAKING

Duncan C. McElfresh
Doctor of Philosophy, 2021

Dissertation Directed by: Professor John P. Dickerson
Department of Computer Science

Algorithms are widely used today to help make important decisions in a variety of domains, including health care, criminal justice, employment, and education. Designing *practical* algorithms involves balancing a wide variety of criteria. Deployed algorithms should be robust to uncertainty, they should abide by relevant laws and ethical norms, they should be easy to use correctly, they should not adversely impact user behavior, and so on. Finding an appropriate balance of these criteria involves technical analysis, understanding of the broader context, and empirical studies "in the wild". Most importantly, practical algorithm design involves close collaboration between stakeholders and algorithm developers.

The first part of this thesis addresses technical issues of uncertainty and fairness in *kidney exchange*—a real-world matching market facilitated by optimization algorithms. We develop novel algorithms for kidney exchange that are robust to uncertainty in both the quality and the feasibility of potential transplants, and we demonstrate the effect of these algorithms using computational simulations with real kidney exchange data. We also study *fairness* for hard-to-match patients in kidney exchange. We close a previously-open theoretical gap, by bounding the price of

fairness in kidney exchange with chains. We also provide matching algorithms that bound the price of fairness in a principled way, while guaranteeing Pareto efficiency.

The second part describes two real deployed algorithms—one for kidney exchange, and one for recruiting blood donors. For each case we characterize an underlying mathematical problem, and theoretically analyze its difficulty. We then develop practical algorithms for each setting, and we test them in computational simulations. For the blood donor recruitment application we present initial empirical results from a fielded study, in which a simple notification algorithm increases the expected donation rate by 5%.

The third part of this thesis turns to human aspects of algorithm design. We conduct several survey studies that address several questions of practical algorithm design: How do algorithms impact decision making? What additional information helps people use complex algorithms to make decisions? Do people understand standard algorithmic notions of fairness?

We conclude with suggestions for facilitating deeper stakeholder involvement for practical algorithm design, and we outline several areas for future research.

# PRACTICAL ALGORITHMS FOR RESOURCE ALLOCATION AND DECISION MAKING

by

Duncan C. McElfresh

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2021

Advisory Committee:
Professor John P. Dickerson, Chair/Advisor
Professor Vincent Conitzer
Professor Michelle Mazurek
Professor Ariel Procaccia
Professor Subramanian Raghavan, Dean's Representative
Professor Neil J. Sehgal

# Dedication

To Hilary.

# Acknowledgments

Thank you John for your unwavering support. You taught me how to be an effective researcher, communicator, and collaborator. You taught me that research can be collegial, and that academia can be a supportive environment if we want it to be. I am inspired by your mentorship, and if I mentor students in the future I will look to you as a model. But most importantly, you taught me that my ideas are valuable. For this confidence I can't thank you enough.

Thank you to my many, many collaborators and mentors for helping me grow and learn. Vince Conitzer, Lok Chan, Jana Schaich Borg, Walter Sinnott-Armstrong, Kenzie Doyle, and Cassi Carley: thank you for including me in the Moral AI family. I am inspired by your model of interdisciplinary thinking. Michelle Mazurek, Candice Schumann, Debjani Saha, and Michael Carl Tschantz: you taught me the basics of HCI and fairness in machine learning. I miss our (hypothetical) hiring dilemmas a fair bit. Tuomas Sandholm, Michael Curry: you showed me how to write a high-quality research paper on a tight deadline. I learned a ton during our month in Pittsburgh. Ruthanne Leishman, Sarah Booker, Morgan Stuart, and Darren Stewart: thank you for teaching me about the practical challenges of kidney exchange. You helped me connect my research with reality, and I am grateful to have you as collaborators. Eric Sodomka, Sergey Pupyrev, Christian Kroer, Karthik Abinav Sankararaman, Zack Chauvin, Neil Dexter, and Caner Gocmen: thank you for helping me bridge the industry-academia divide. You showed me how to conduct research that is both innovative and also useful. I truly appreciate your mentorship and support. Neil Sehgal: your guidance is a big reason that I'm excited for a future in health care

and public health. I am grateful for your advice on my career and research, and for our conversations on advance care planning.

I am especially grateful for the AMSC faculty and staff. You have created an environment where students can explore nearly any research topic we wish, while providing stability and support from our "home" department. I feel very lucky to be an AMSC student. Jessica Sadler: without you I would be nowhere close to graduation. You helped me navigate the jungle of rules and requirements, deadlines and fees. I am very grateful for your support. Howard Elman, Konstantina Trivisa, Maria Cameron: you helped me navigate the AMSC program while encouraging my adventures in other academic departments. Thank you for cultivating a supportive and curious learning environment.

Finally, thank you to my family. To Hilary and Leia, to Michael, to the McElfreshes and Chisholms and Hursts and Mayers. I love you all.

# Table of Contents

# List of Tables

# List of Figures

# Chapter 1: Introduction

Algorithms are all around us, whether we like it or not. If you are reading this thesis on a screen, at least a few algorithms were involved. If you are admitted to a hospital, algorithms sound an alarm if your heart rate becomes erratic. If you apply for a loan, algorithms help with underwriting and setting an interest rate. In the past two decades, increasing availability of data computing power has enabled the development of complex machine learning (ML) and artificial intelligence (AI) algorithms. But not all algorithms are complex, and not all algorithms run on a computer.

**What is an algorithm?** In this thesis I use *algorithm* to mean a process that transforms some input into some output.[1] I focus on algorithms used to advise a decision maker (e.g., suggesting a drug dosage to a physician), or take an action on behalf of a decision maker (e.g., automatically rejecting a suspicious loan application). This definition of course includes complex AI and ML algorithms, which are in fact a focus of this thesis. However we should not neglect "simple" algorithms, which are not always as simple as they seem.

There are a variety of ways to characterize how a deployed algorithm behaves "in the wild": How does it behave in different environments? With different users? How is it used? How does it impact human behavior or judgment? When does it make mistakes? These questions help us decide whether an algorithm is *practical*

[1] In this definition, an algorithm is simply a mathematical *function*.

for a specific application. Importantly, these are questions of human judgment, and not mathematical analysis. In particular we should consider judgments from people who use, and who are impacted by, algorithmic systems; throughout this thesis I refer to these people as *stakeholders*. There is often a dichotomy between stakeholders and the researchers and engineers responsible for designing and deploying an algorithm—who I refer to as *technicians*. As a technician it is important to understand the needs and concerns of stakeholders, and to *characterize* our work in a way that is relevant and understandable. At the same time it is important for stakeholders to understand the nature of algorithms that impact their lives. This presents a challenge: most technicians are not stakeholders in their own creations, and most stakeholders are not trained computer scientists. One way to bridge this knowledge gap is using common standards—for example transparency or fairness—for algorithm design; another way is to involve stakeholders directly in the design process.

**Thesis Statement**

Designing practical algorithms requires technical rigor, consideration for context, and empirical analysis "in the wild"; this is best achieved through close collaboration between stakeholders and technicians.

Part I of this thesis focuses on algorithms for matching patients and donors in kidney exchange. This is an application where algorithms play an incredibly important role, and their behavior is scrutinized closely by stakeholders. This part focuses on *technical* aspects of algorithm design, and in particular robustness to uncertainty (Chapters 3, 4, 5), and algorithmic fairness (Chapter 6).

In Part II I focus less on technical aspects of algorithm design, and more on their underlying applications. I present two example projects involving direct collaboration with stakeholders. Chapter 8 describes a project in collaboration with the United Network for Organ Sharing (UNOS); we develop an algorithm to prioritize

kidney exchange donors for pre-screening, which is compatible with current UNOS policies. Chapter 9 describes a collaboration with Facebook for the Facebook Blood Donation tool, where we design an algorithm to automatically notify potential blood donors about relevant donation opportunities.

In Part III I address the human challenges of algorithm design. Chapter 11 investigates the impact of AI suggestions on decision making; Chapter 12 describes a method for finding *useful* explanations for stakeholders; Chapter 13 highlights the importance of indecision in modeling human decisions; Chapter 14 investigates whether people understand standard notions of algorithmic fairness.

Chapter 15 gives concluding remarks, and Chapter 16 outlines several directions for future work.

# Part I

# Optimization Algorithms for Kidney Exchange

# Chapter 2: Introduction to Kidney Exchange

Patients with end-stage renal disease (kidney failure) have two treatment options: a lifetime on dialysis, or kidney transplantation. Dialysis[1] is a continuous procedure is far more expensive and burdensome than transplantation, and life expectancy for dialysis patients is far shorter than patients who receive a transplant. Table 2.1 gives a rough sketch of the trade-offs between these two treatments. While transplantation is by far the more-preferred treatment among patients with end-stage renal disease, the need for donor organs far outstrips supply. At any given time, there are roughly 100,000 patients in the United States waiting for a kidney, and about 20 of these people die each day. Roughly 20,000 patients receive a transplant each year, and the waiting list continues to grow [253].

[1]There are two types of dialysis: hemodialysis (HD) and peritoneal dialysis (PD). HD is done in a specialized clinic, where blood is removed from the patient's body, filtered, and then returned. PD uses a catheter in the patient's abdomen to remove fluids and waste products, and can sometimes occur at home rather than in a clinic. HD is far more common than PD: a 2012 study [171] found that 11% of the global dialysis population used PD dialysis, and this varies widely across countries. In most countries the proportion of PD patients is very small ( Japan: 3%, Germany: 4.8%, USA: 7%), however in a few countries the proportion of PD patients is quite high (Hong Kong: 79.4%, El Salvador: 76.5%, Mexico: 65.8%).

TABLE 2.1: Basic trade-offs between hemodialysis and transplantation for patients with end-stage renal disease.

| | **Hemodialysis** | **Kidney Transplantation** |
|---|---|---|
| Five-year survival rate [279] | 66% | 90% |
| Annual Cost/QALY [31] | $72k | $40-60k |
| Required treatment[2] | 12 hours of treatment per week | Transplantation surgery, lifelong medication and checkups |

## 2.1 Kidney Transplantation

There are two sources of donor kidneys used in transplantation: *deceased* donors and *living* donors. Deceased donation is slightly more common: the World Health Organization (WHO) estimates [301] that about 58% of the roughly 90,000 kidney transplants in 2016 came from deceased donors; however there are very likely more living *donors* than deceased donors, since living donors can donate only one kidney, while deceased donors can donate two.[3] In the United States, between 20-30% of kidney transplants come from living donors [253].

The criteria for joining the deceased donor waiting list, and the prioritization rules for allocating organs to the list, vary widely around the world. Both eligibility and priority are often determined by public policy, which prioritizes patients by medical, logistical, and moral criteria. For example, in the United States this process is managed by the Organ Procurement and Transplantation Network (OPTN), and their policies are publicly available [252]. OPTN prioritization criteria reflect both medical and logistical compatibility (blood type, antigen mismatch, location of the donor and recipient, and so on), as well as moral criteria (patient age, time spent on the waiting list, whether they are a prior organ donor). However, relatively few patients receive an organ through the waiting list: according to OPTN, around 15% of patients on the waiting list receive a transplantation within two years [253]. Unequal access to the deceased donor waiting list is a major cause of disparity in outcomes for patients; several factors impact a patient's likelihood of joining the waiting lists, including socioeconomic status [331], race [242], geography [274], education [276], and language [300].

Patients can also receive a transplantation from a *living* kidney donor—often a

[3] For religious reasons, many countries only allow transplantation from living donors [143].

family member or friend of the patient, and occasionally a stranger found through chance meetings, advertisements, or social networks [165]. After identifying a willing, living kidney donor, there are multiple steps to take prior to transplantation. First, most countries require that donors are screened to ensure that they were not coerced or induced to donate. The WHO warns against undue coercion in their Guiding Principles [301]; similar concerns have led some developing countries to ban living transplants from unrelated donors [85]. Second, the patient and donor must be *medically* compatible. Since this medical compatibility is especially important to this thesis, it deserves its own paragraph.

**Patient-Donor Compatibility** There are two primary factors that determine whether a patient and donor are medically compatible for kidney transplantation: blood type (ABO) compatibility and tissue type compatibility. Many exchanges require that patients and donors are ABO-compatible, [4] and these compatibility requirements are identical to those for blood donation. Tissue type compatibility is typically measured by the human leukocyte antigen (HLA) typing. A perfect HLA match (zero-mismatch) between patient and donor is preferable, though some degree of HLA mismatch is acceptable (though not preferred) by transplant teams [326]. HLA type is also used to estimate how likely a patient is to find a compatible donor. The calculated panel reactive antibody score (CPRA) derived from HLA type data, and estimates the fraction of donors that a patient is incompatible with; CPRA ranges from 0 (very likely to find a compatible donor) to 100 (very unlikely to find a compatible

[4] ABO-incompatible transplantation is possible, though ABO-compatibility is preferred [227].

donor). Patients with high CPRA are considered *highly-sensitized*.[5] Furthermore, patients can *become* highly-sensitized during certain medical treatments (such as transplantation or transfusion); in 2018, roughly 20% of all patients on the US deceased donor waiting list were highly-sensitized [162].

## 2.2 Kidney Exchange: Swapping Donors

Kidney exchange is a process where patients who have willing (and possibly incompatible) living kidney donors "swap" their donors in order to find a better match. A simple kidney exchange is illustrated in Figure 2.1: two patients ("husband" and "daughter") each have a willing, but incompatible donor ("wife" and "mother"); however, each donor is compatible with the *other* donor's patient, and thus both patients can receive a transplant. Modern kidney exchanges involve more complex cycle- and chain-like structures involving several patient-donor pairs. Many kidney exchange programs exist around the world, accounting for roughly 8% of living donor transplants; exchanges continue to expand worldwide [51]. Exchanges vary widely in their structure and governance: several countries have a single, national or single-center exchange—including the Netherlands, the United Kingdom, Canada and Australia [130]. National exchanges benefit from coordination and standardization—and in particular, larger exchanges benefit from increased market "thickness" (i.e., more potential transplants). On the other hand, exchanges in the United States have developed in a fragmented manner: there are many US-based exchanges of varying sizes, who compete for patients and donors. The largest US-based exchanges include the National Kidney Registry, the Alliance for Paired Donation, Johns Hopkins University, and the Methodist Specialty and Transplant Hospital

[5] The CPRA threshold for highly-sensitized patients varies. For example, OPTN policy prioritizes patients on the deceased donor waiting list according to a sliding scale, however in kidney *exchange*, patients with CPRA above 80 are considered highly sensitized and receive additional prioritization [252].

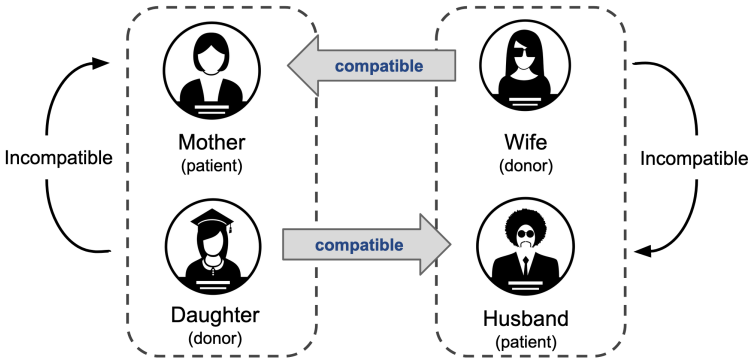



FIGURE 2.1: A simple pairwise exchange.

in San Antonio; each of these is—on its own—larger than many countries' national exchanges. The fragmented nature of US exchanges, paired with the privatization of US health care, leads to competition between exchanges that make it more challenging to match patients and donors [7].

### 2.2.1 Facilitating Organ Exchange via Cycles and Chains

A standard tenet of kidney exchange is that every patient-donor pair who *contributes* a donor organ to the exchange should also *receive* a donor organ. This means that any organ exchange between two or more patient-donor pairs is a **cycle**, which can be represented by an ordered list of patient-donor pairs: the first pair's donor donates to the second pair's patient, the second pair's donor donates to the third pair's patient (and so on); the final pair's donor donates to the *first* pair's patient, completing the cycle. A cycle involving $k$ patient-donor pairs is called a *$k$-cycle*. For example, the simple exchange in Figure 2.1 is a 2-cycle between the (Mother, Daughter) pair and the (Wife, Husband) pair. Exchanges usually require that all transplants in a cycle occur "simultaneously", meaning that all patients and donors undergo transplantation surgery at roughly the same time [50]. If cycle transplants occur non-simultaneously, the cycle may "break" if one of the transplants is canceled for medical reasons, or if a donor reneges. In this case, a patient may be left with no compatible donor, *and* no

donor kidney. Each patient-donor pair participating in exchange involves two people undergoing transplantation (both the patient and donor); so, a 2-cycle involves four surgeries, a 3-cycles involves six, and so on. For this reason many transplant centers can only accommodate 2- and 3-cycles though much longer cycles have been reported.

Intuitively, longer cycles create opportunities to match patients with donors; by one estimate, using unlimited-length cycles can increase the fraction of transplants in a random exchange pool from 54% to 91% [267]. Fortunately, only 2- and 3-cycles are required to achieve most of this benefit (if only cycles are allowed). Using theoretical models of kidney exchange, Roth et al. [270] and later Ashlagi and Roth [23] show that cycles longer than 3 pairs does not significantly increase the number of transplants in an exchange pool.

Modern exchanges also include Non-Directed Donors (NDDs), sometimes referred to as "altruists," who contribute a donor organ to the exchange without receiving one in return. If a NDD can begin a **chain** of transplants by donating to a patient-donor pair; the paired donor is then free to donate to another patient-donor pair, and so on. By donating to an exchange pool rather than a deceased donor on the waiting list, a NDD may produce more than one kidney transplant—multiplying the impact of their donation. Using simulated exchanges, Dickerson et al. [106] show that if NDDs initiate chains rather than donate to the waiting list, $4-5\%$ more patients can receive transplants; even greater benefit is possible in smaller and sparser exchanges. Chains first appeared in the form of domino paired donation (DPD), where a NDD donates to a patient-donor pair, who then donates to another pair, and so on; the final pair in the chain donates to a patient on the waiting list. Most exchanges require DPD chain transplants to be carried out simultaneously, which (as with cycles) limits their length to 2 or 3.

Non-simultaneous chains can be far longer than DPD chains, and can significantly increase the size and equity of kidney exchange. The first reported non-simultaneous chain occurred in 2009, consisting of 10 transplants, and over a period of 8 months [259]. These are referred to as non-simultaneous extended altruistic donor (NEAD) chains, and are currently used by many exchanges in the US and abroad [130]. As with cycles, non-simultaneous chains run the risk of "breaking" if a donor reneges or the transplant is canceled. This risk is exacerbated when chains occur over long time periods and long distances. Indeed many exchanges—such as the UNOS Pilot Program, and the national exchanges in the Netherlands and the UK—limit chains to a length of 4; in many cases this limit is due to the requirement of simultaneous transplants [210].

While simultaneous transplants are still the norm, non-simultaneous chains and cycles are becoming more common, and initial findings show that the risk of a chain "breaking" is low. One study of the National Kidney Registry (NKR, a US-based exchange), only 5.6% of all NEAD chains were broken for any reason, and an estimate 1.5% of all donors reneged on a transplant; in aggregate, broken chains did not impact the overall number of transplants [92].

### 2.2.2 Transplantation in the United States

In many countries organ transplantation is coordinated by a government organization, and kidney exchange is one small part of this operation. In the United States, organ transplantation is handled by the Organ Procurement and Transplantation Network (OPTN)—an organization structure established by the US congress in the National Organ Transplant Act (NOTA) of 1984.[6]. OPTN is managed by a private

[6] http://uscode.house.gov/view.xhtml?hl=false&edition=prelim&req=granuleid%3AUSC-2014-title42-section274

contractor, the United Network for Organ Sharing (UNOS), who oversees day-to-day operations and policy management. UNOS has many responsibilities: managing the organ waiting list, maintaining a database of all transplant data in the US, monitoring all organ patient-donor matching to ensure compliance with policies, educating health care experts and the public about organ transplantation, and managing transplantation policy.[7]

In the US, health care providers (e.g., hospitals) play an especially large role in transplantation. Individual hospitals conduct their own donor and patient recruitment, screening, and matching of patient-donor pairs; many hospitals also conduct their own internal exchanges, involving only their patients and donors.

### 2.2.3 An Operational Perspective of Kidney Exchange

From the perspective of an exchange coordinator, there are roughly three steps involved in facilitating a kidney exchange, outlined in Figure 2.2.

1. First, the exchange **identifies the exchange pool**, including all potential patients, donors, and compatible transplants. During this step, potential transplants are deemed *feasible* if they are both medically and logistically compatible, according to the exchange policy. Some exchanges use additional factors to prioritize certain transplants over others, related to patient health, location, waiting time, prior donation history, and so on; each of these factors is recorded during this step. During this step, some exchanges ask individual patients (and their transplant teams) whether they would accept a particular donor organ. This *pre-screening* information can substantially improve the exchange outcome, and this is the focus of Chapter 8.

[7] https://unos.org/about/

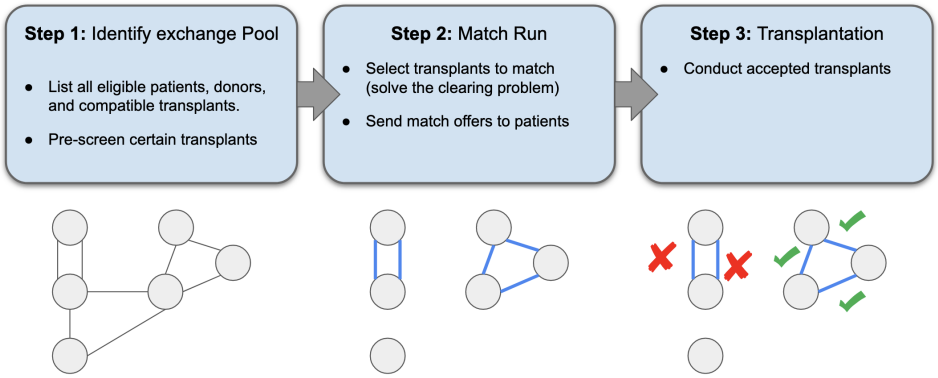



FIGURE 2.2: A high-level operational perspective of kidney exchange.

2. Next, the exchange **matches patients and donors** according to a "matching policy." In other words, the matching policy selects which transplants are used to facilitate organ exchange. In countries with centralized exchanges—and in some US-based exchanges—this matching policy is a matter of public policy.[8] This is also the most computationally intensive step of kidney exchange: modern matching policies often require solving an NP-hard problem; much of this thesis focuses on solving variants of this problem. For each transplant selected by the matching policy, a *match offer* is sent to the patient and their care team—with information about the donor organ, and instructions on how to accept or reject the offer.

3. If a match offer is rejected, this can cause some transplants to become *infeasible*. For example in Figure 2.1 if "husband" rejects a match offer from "mother", then the transplant from "wife" to "daughter" cannot proceed because "wife" will donate her kidney without "husband" receiving one in return—so both transplants in this exchange become infeasible. Finally, all feasible patient-donor pairs are **transplanted**.

[8]For the US organ clearinghouse, OPTN, policy changes are subject to a six-phase review process which can take years (https://unos.org/policy/policy-changes/).

A substantial body of research from computer science, operations research, and economics focuses on step two of this process: developing matching policies for kidney exchange. Indeed, this is the focus of Chapters 3, 4, 5, and 8.

## 2.3 The Kidney Exchange Clearing Problem (KEP)

Kidney exchange matching policies use information about the exchange pool to recommend match offers (i.e., potential transplants) to the exchange coordinator. To design real kidney exchange policies we first analyze the *Kidney Exchange clearing Problem* (KEP)—which uses a (slight) abstraction of the real kidney exchange matching process.

The KEP uses a directed graph representation of kidney exchange $G = (V, E)$, where each vertex $v \in V$ represents a participant in the exchange (including patient-donor pairs and NDDs). Vertices are partitioned into patient-donor pairs $P \subseteq V$ and NDDs $N \subseteq V$, with $V \equiv P \cup N$. [5, 267, 268]. Each potential transplant from a donor at vertex $u$ to a patient at vertex $v$ is represented by a directed edge $e = (u, v) \in E$, which has an associated weight $w_e \in \mathbf{w}$; weights are set by the exchange coordinator, and usually reflect both the medical utility of the transplant, as well as ethical considerations (e.g., prioritizing patients by waiting time, age, and so on). Directed cycles in $G$ correspond to cyclic trades between multiple patient-donor pairs in $P$, and chains correspond to donations that begin with an NDD in $N$ and continue through multiple patient-donor pairs in $P$. An example exchange is shown in Figure 2.3, with seven patient-donor pairs (nodes $p_i$, $d_i$) and a single NDD ($n$). There are several cycles and chains in this graph: the NDD can initiate a chain with either pair 1 or pair 7, leading to 12 *different* chains, up to length 5; note that only one of these chains can be matched, since the NDD can participate in a matching only once.

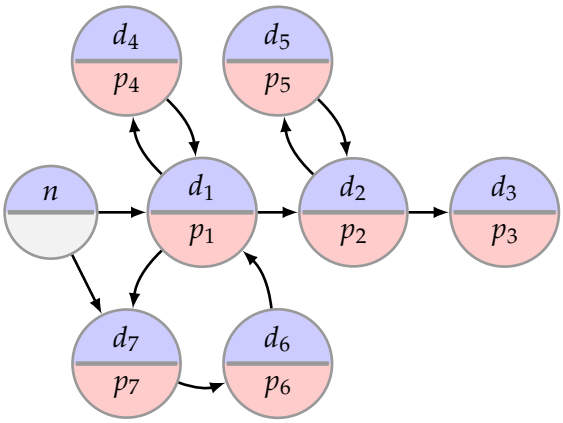



FIGURE 2.3: Example exchange graph with seven patient-donor pairs $(p_i, d_i)$ and a single NDD $n$. There are 12 distinct chains up to length 5, and three cycles.

There are three cycles, between pairs (1, 4), (2, 5), and (1, 7, 6).

Since real exchanges typically limit the length of cycles and chains, in the KEP we allow cycles up to length $K$ and chains up to length $L$; these are referred to the cycle and chain *cap*. Furthermore, each vertex (participant) can only participate in one cycle or chain—since each person can only donate or receive one kidney. Thus, the set of match offers in an exchange corresponds to a set of vertex-disjoint cycles and chains in $G$, consisting of cycles of up to length $K$ and chains of up to length $L$. We refer to any such set of cycles and chains as a *matching*, where $\mathcal{M}$ denotes the set of all matchings in $G$. Combining these considerations yields the KEP.

**Definition 1** (Kidney Exchange clearing Problem (KEP))**.** Find a matching with maximal total edge weight in $G$.

## 2.4 Complexity

The complexity of the KEP depends on the cycle cap $K$ and chain cap $L$. When $K = 2$ and $L = 1$ the KEP can be reduced to max-weight matching, and thus can be solved in polynomial time. The reduction is quite simple: create an undirected graph $G' = (V, E')$ which has the same vertices as the KEP graph $G = (V, E)$. For each 2-cycle in

$G$ between vertices $v_1$ and $v_2$, add a single undirected edge to $E'$ between $v_1$ and $v_2$, with weight equal to the sum of edges in the 2-cycle. For each edge $e = (n, v) \in E$ from a NDD $n$ to a patient-donor pair $v$, add a single undirected edge to $E'$ between $n$ and $v$ with edge weight equal to $w_e$. There is a 1-1 correspondence between each undirected matching in $G'$ and each cycle-and-chain matching in $G$, and thus a max-weight matching in $G$ can be found by finding a max-weight matching on $G'$.

Interestingly, the KEP can also be solved in polynomial time when $K = L = \infty$ by reduction to bipartite matching.

For all other cycle and chain caps $2 < K < \infty$ and $1 < L < \infty$, the KEP is both NP- and APX-hard, even with uniform edge weights [5, 49]. Until the mid 2010s, most KEP solution methods involved enumerating all cycles and chains in $G$, which can quickly become intractable. This is evident even in the small exchange in Figure 2.3: even with eight vertices there are 12 distinct chains and three cycles to choose from.

**KEP as Cycle Packing** The KEP is a variant of *cycle packing*, a fundamental problem in algorithmic graph theory with classical results dating to at least 1959 [123, 205]. There are many variants of the cycle packing problem—on directed and undirected graphs, with and without edge weights, and with and without minimum or maximum cycle lengths—and many of these are NP- and APX-hard [254]. Cycle packing problems naturally arise in many real-world settings, including scheduling, transportation, and health care. This thesis covers several real-world challenges related to the KEP, which naturally apply to other variants of cycle packing: such as data-driven uncertainty, fairness, and tractability.

## 2.5 Solving the KEP

Modern KEP solution methods use Mixed Integer Programming (MIP) and constraint or column generation techniques, and can solve realistic problem instances with hundreds of vertices and thousands of edges in fractions of a second. There are several MIP formulations in the literature which can be solved using software packages such as Gurobi and CPLEX. The subsequent chapters of this thesis use these MIP formulations as a subroutine, or as a benchmark, to address other practical challenges of kidney exchange. In these chapters I discuss three different MIP formulations for the KEP (PICEF [109], PC-TSP [16], and PI-TSP [216]). All of these formulations are a variation of the following approach: let $\boldsymbol{x} \in \mathbb{Z}^n$ be a vector of $n$ *decision variables*, each representing a different structure in the kidney exchange. For example if we treat edge as a separate structure, then $n = |E|$, and $\boldsymbol{x}_e = 1$ if edge $e$ is matched, and $\boldsymbol{x}_e = 0$ otherwise. Let $\mathcal{M}$ denote the set of *legal matchings* on $G$, meaning that $\boldsymbol{x} \in \mathcal{M}$ if $\boldsymbol{x}$ is a set of vertex-disjoint cycles of length $C$ or less, and chains of length $L$ or less. Finally, let $\boldsymbol{w} \in \mathbb{R}^n$ be a vector of *weights*, where $\boldsymbol{w}_i$ is the weight of edge or cycle or chain corresponding to $\boldsymbol{x}_i$ The KEP can now be written as $\max_{\boldsymbol{x} \in \mathcal{M}} \boldsymbol{w}^\top \boldsymbol{x}$.

The difficulty here lies in defining a reasonable set of decision variables $\boldsymbol{x}$ and legal matchings $\mathcal{M}$. Typically we write this formulation in terms of the set of *cycles* (denoted by $C$), and cycles and chains (denoted by $\mathcal{C}$). To illustrate this difficulty, here I introduce two simple integer programming formulations that serve as the foundation of all modern KEP solution approaches: the *edge formulation* and the *cycle formulation*.

**Edge Formulation** The basic edge formulation assigns one binary decision variable to each *edge* in the exchange graph; the number of edges in real exchange graphs is around $10,000$, so the total number of decision variables is relatively small. However the number of constraints in the edge formulation can be quite large. First we require *capacity constraints*: each vertex can be matched at most once in an exchange; this requires one constraint for each vertex: the number of incoming edges that are matched can be at most 1 for each $v \in V$. Next we require *flow constraints*: each patient-donor pair can only donate a kidney if they receive one in return. Again we need one constraint for each vertex: the number of matched outgoing edges can be at most equal to the number of matched incoming edges, for each $v \in V$. So there are $O(|V|)$ flow and capacity constraints, which is reasonable—real exchanges have at most 100s of vertices. If there is no limit on cycle and chain length, we are done, and the formulation has $O(|E|)$ decision variables and $O(|V|)$ constraints (recall that the KEP can be solved in polynomial time with no cycle and chain cap). However there is a finite cycle or chain cap, we need additional constraints to limit cycle and chain length. There are a variety of ways to restrict cycle and chain length in edge formulations, and most of them require an exponential number of constraints. For example the approach of Anderson et al. [16] uses constraint generation to selectively restrict illegal chains, and Abraham et al. [6] uses constraint generation for restricting illegal cycles *and* chains. The current state-of-the-art KEP solvers use *compact* KEP formulations (i.e., with a (polynomial number of constraints and variables) based on the edge formulation; this includes the position-indexing (PICEF) approach of Dickerson et al. [109] and the extended-edge-formulation of Constantino et al. [90].

**Cycle Formulation** Another fundamental IP formulation for the KEP assigns one binary decision variable to each legal *cycle* in the exchange graph. In the cycle formulation, only capacity-like constraints are required: for each vertex $v \in V$, the number of matched cycles and chains containing $v$ can be at most 1; so the number of constraints is $O(|V|)$. Furthermore, the cycle formulation can also include chains: consider a graph $G' = (V, E')$, where $E'$ includes all edges in $E$ as well as one zero-weight directed edge (a "dummy edge") from each patient-donor pair to each NDD. Each cycle in $G'$ that includes an NDD corresponds to a chain in $G$; in this way the cycle formulation can be modified include chains. However the number of potential cycles grows exponentially with cycle length, so the cycle formulation is intractable for even moderate cycle and chain caps. Indeed, Abraham et al. [6] was unable to store cycle formulations in memory for a $1,000$-vertex graph and a cycle cap of three. One way to avoid enumerating all cycles is through *branch-and-price*: a column- generation method, which adds cycles selectively to the formulation as needed [6]. Fortunately, current exchanges are sufficiently small and sparse that all legal cycles (but not chains) can be enumerated very quickly. State-of-the-art KEP solution methods use a mix of cycle and edge formulation to reduce runtime; these solution methods are used throughout this thesis.

## 2.6 The KEP as an Abstraction of Kidney Exchange

It is important to emphasize that the KEP is an abstraction of the messy, real-world process of kidney exchange. While the KEP is useful for reasoning about how organs can and should be allocated, there are several practical considerations that researchers should keep in mind when studying the KEP. As a conclusion to this chapter I will illustrate some of the most important practical considerations for computer

scientists interested in kidney exchange research. These should be seen as *research opportunities*—studying them helps us uncover new technical challenges, and makes our research more useful for stakeholders.

**Uncertainty** Two factors are especially uncertain in kidney exchange: (1) the *feasibility* of a potential transplant, and (2) the *quality* of a potential transplant. (1): There are a variety of reasons that a potential transplant is infeasible—most commonly, a patient and donor may in fact be medically incompatible, often due to a mismatch in Human Leukocyte Antigen (HLA) typing. Transplants may also be infeasible due to logistical or transportation issues, or due to a patient or donor refusing the transplant. (2): The *value* of a transplant is a matter of moral judgment. Even among experts there is disagreement about which transplants are most *valuable* to an exchange, or to society. This disagreement can be seen as a type of uncertainty: suppose we can only match one of two cycles: either (a) a 2-cycle that will extend the lives of two children by 30 years, or (b) a 3-cycle that will extend the lives of three elderly organ donors by 10 years each. There are good reasons to match both of these cycles, and finding the "correct" answer is not a question of computer science. Even if a clear measure of transplant value is established—which is the case of OPTN in the US [252]—these value measurements are very uncertain. These measurements usually depend on medical screenings and logistical assumptions, both of which can be inaccurate. Most of the following chapters in Part I of this thesis deal with this type of uncertainty.

**Dynamics** Researchers often treat kidney exchange as a *static* problem: the patient-donor pool is considered to be constant (i.e., patient-donor pairs do not enter or exit the pool while a matching is being constructed), and matching policies tend to consider only the *current* exchange pool, and not future pools. However kidney

exchange is an inherently dynamic process: patient-donor pairs and NDDs are constantly entering and exiting the exchange, and today's matching inevitably impacts tomorrow's exchange pool. Research has demonstrated that matching policies designed for a dynamic environment can increase overall welfare [30, 104, 105, 307]. Unlike static policies, dynamic policies can be updated based on observations of previous exchange pools, the current state of the exchange pool, and the chance that patients will enter or exit the pool in the next stage. That is, dynamic policies can *change* based on the exchange pool, and can change over time. However dynamic kidney exchange is both computationally and theoretically more difficult to analyze than static exchange, and conclusive results are scarce (see Chapter 16).

**Incentives and Preferences** Each participant in a kidney exchange has different incentives. First, patients typically want to find a high-quality compatible organ as quickly as possible. What constitutes a "good match" depends on the particular patient and transplant team. Donor-patient compatibility is in fact a spectrum: different levels of HLA mismatch are acceptable (and can be accommodated with immunosuppression treatment), and blood-type incompatible transplants are now relatively common. Match quality involves a trade-off between a variety of factors, and deciding whether or not to accept a particular organ is a matter of preference [28]. Different patients have different preferences over donor kidneys [264], and surgeons' preferences sometimes differ from those of their patients. Solomon et al. [291] finds that as patients spend more time on the deceased donor waiting list, they assign less importance to donor kidney quality. The foundational matching models of kidney exchange explicitly account for patient preferences over donors [267, 269], and modern exchanges do account for patient preferences to some extent. Many exchanges allow patients to specify general parameters of organs they are willing

to accept—for example by specifying acceptable donor blood types, donor age, or donor organ health. It should be noted that *donor* preferences certainly play a role in living donor transplantation. However many exchanges (and the OPTN in the US) attempt to minimize donor influence over their organ recipient to avoid arbitrary biases—in particular racism [304].

*Hospital incentives* are also an important factor in kidney exchange. This is especially a problem in the US, where the majority of kidney exchange occurs within private hospitals rather than a centralized program. Many exchanges involve several hospitals; each hospital manages their own network of patients and donors, and each decides which patients and donors to include in the exchange. This is the model used by many US-based exchanges, including the OPTN/UNOS exchange, the Alliance for Paired Donation[9], and the National Kidney Registry.[10] For hospitals participating in these exchanges, there are some benefits to keeping certain transplants in-house, rather than including them in an exchange. First, hospitals benefit by conducting transplants in-house rather than sending their patients and donors to another hospital; this keeps their staff trained, brings in additional revenue, and keeps their patients within the hospital network. Estimated overall revenue per transplant is over $\$100,000$ [164], so hospitals are strongly incentivized to match their own patients and donors rather than exchange them with other transplant centers. For example, a competitive transplant hospital may wish to keep high-quality donors within their local exchange rather than "sharing" this donor through a national or regional exchange; this results in fewer, and lower-quality transplants for patients. While this problem is not a central focus of this thesis, it is an active area of market design research [7, 27, 32, 156, 292].

[9] https://paireddonation.org/
[10] https://www.kidneyregistry.org

# Chapter 3: Robust Kidney Exchange: Protecting against the Worst-Case

## 3.1 Introduction

Real-world optimization problems face various types of uncertainty that impact both the quality and feasibility of candidate solutions. Uncertainty in combinatorial optimization is especially troublesome: if the *existence* of certain constraints or variables is uncertain, identifying a good—or even feasible—solution can be extremely difficult. Stochastic optimization approaches endeavor to maximize the *expected* objective value, under uncertainty. While sometimes successful, stochastic optimization relies heavily on a correct characterization of uncertainty; furthermore, stochastic approaches are often intractable—especially in combinatorial domains [44]. A complementary approach is *robust optimization*, which protects against worst-case outcomes. Robust approaches can be less sensitive to the exact characterization of uncertainty, and are often far more tractable than stochastic approaches [41].

**Uncertainty in kidney exchange.** Presently-fielded kidney exchange algorithms largely do not address uncertainty. In this chapter we consider two types of uncertainty in kidney exchange: over the *quality* of the transplant (weight uncertainty) and over the *existence* of potential transplants (existence uncertainty). Policymakers assign weights to potential transplants, which are (imperfect) estimates of transplant

quality; weight uncertainty stems from both measurement uncertainty (e.g., medical compatibility and kidney quality) and uncertainty in the prioritization of some patients over others. Transplant existence is always uncertain: matched transplants "fail" before executing for a variety of reasons, severely impacting a planned kidney exchange. To address both cases, we propose *uncertainty sets* containing different realizations of the uncertain parameters. We then develop a scalable robust optimization approach, and demonstrate its success on data from a large fielded kidney exchange.

Robust optimization is a popular approach to optimization under uncertainty, with applications in reinforcement learning [243], regression [322], classification [80], and network optimization [222]. Motivated by real-world constraints, we apply robust optimization to kidney exchange—a graph-based market clearing or resource allocation problem.

**Contributions.**

- To our knowledge, weight uncertainty has not been previously addressed in the kidney exchange literature. Our approach is similar to that of Bertsimas and Sim [43] and Poss [249], and uses some of their results. Several approaches have been proposed for existence uncertainty, primarily based on stochastic optimization [16, 109, 110] or hierarchical optimization [210]. The primary disadvantage of these approaches—in addition to tractability—is their reliance on, and sensitivity to, the explicit estimation of the probability of each particular potential transplant. This probability is extremely difficult to determine [110, 147], and prevents the translation of those methods into practice. Our approach uses a simpler notion of edge existence uncertainty—an upper-bound on the number of non-existent edges—which is easier to interpret and

estimate. Glorie [146] proposed a related robust formulation that is exponentially larger than ours, and is intractable for realistically-sized exchanges.

- We introduce a new scalable formulation for kidney exchange that combines concepts from two state-of-the-art formulations [16, 109], handles long or uncapped NDD-initiated chains without requiring expensive constraint generation, and ties into a developed literature on fairness in kidney exchange—thus addressing use cases that are becoming more common in fielded exchanges [16].

## 3.2 Preliminaries

*Robust optimization* is a common approach to optimization under uncertainty, which is often more tractable and requires less accurate uncertainty information than other approaches [44]. This approach begins by defining an *uncertainty set* $\mathcal{U}$ for the uncertain optimization parameter, where $\mathcal{U}$ contains different *realizations* of this parameter. Consider the example of edge weight uncertainty: we might design an edge weight uncertainty set $\mathcal{U}_w$ that contains the *realized* (i.e., "true") edge weights $\hat{\boldsymbol{w}}$ with high probability, $P(\hat{\boldsymbol{w}} \in \mathcal{U}_w) \geq 1 - \epsilon$, for $0 < \epsilon \ll 1$. The parameter $\epsilon$ is referred to as the *protection level*, and is often used to control the number of realizations in $\mathcal{U}$.

After designing $\mathcal{U}$, the robust approach finds the best solution, assuming the *worst-case* realization within $\mathcal{U}$. For kidney exchange (a maximization problem), this corresponds to a *minimization* over $\mathcal{U}$; for example, Problem 3.1 is the robust formulation for a KEP with uncertain edge weights.

$$\max_{\boldsymbol{x} \in \mathcal{M}} \min_{\hat{\boldsymbol{w}} \in \mathcal{U}} \quad \hat{\boldsymbol{w}}^\top \boldsymbol{x} \tag{3.1}$$

The robustness of this approach depends on the proportion of possible realizations contained in $\mathcal{U}$. If $\mathcal{U}$ contains all possible realizations, the approach may be too conservative; if $\mathcal{U}$ only contains one possible realization of $\hat{w}$, the solution may be too myopic. The number of realizations in $\mathcal{U}$ is often controlled by a parameter: either an *uncertainty budget* $\Gamma$, or the protection level $\epsilon$. Next we introduce the first type of uncertainty considered in this chapter: edge weight uncertainty.

## 3.3 Optimization in the Presence of Edge Weight Uncertainty

Edge weights in kidney exchange represent the medical and social utility gained by a *single* kidney transplant. Weights are determined by policymakers, and are subject to several types of uncertainty.[1] Part of this uncertainty is due to insufficient knowledge of the future: a patient or donor's health may change, raising or lowering the "true" weight of their transplant edges. Another type of uncertainty stems from disagreement between policymakers regarding the social utility of a transplant. For example, some policymakers might prioritize young patients over older patients; other policymakers might prioritize the sickest patients above all healthier patients. Policymakers aggregate these value judgments to assign a single weight to each transplant edge, but this aggregation is a contentious and imperfect process (although recent work from the AI community has begun to address this using techniques from computational social choice and machine learning [132, 229]). Still, there is no way to measure the "true" social utility of a transplant, and therefore this uncertainty is not easily measured.

**Interval weight uncertainty.** It is beyond the scope of this work to characterize all sources of edge weight uncertainty. Rather, we assume that the *nominal* edge weights

[1] The process used to set weights by the UNOS US-wide kidney exchange is published publicly [306].

$w$, provided by policymakers, are an uncertain estimate of the *realized* edge weights $\hat{w}$, i.e., the "true" value of each transplant. Next, we formalize edge weight uncertainty and our robust approach. This section focuses on edge weights, so we write our formulations with decision variables $x \in \{0,1\}^{|E|}$ corresponding to individual edges: $x_e$ is 1 if $e$ is matched, and 0 otherwise.

We assume that realized edge weights $\hat{w}$ are random variables with a partially known symmetric distribution, centered about the nominal weights $w$. This assumption implies that $E[\hat{w}] = w$, thus a non-robust approach that maximizes $w$ is equivalent to a stochastic optimization approach that maximizes *expected* edge weight. We refer to this edge uncertainty model as *interval uncertainty*.

**Definition 2** (Interval Edge Weight Uncertainty)**.** Let $\hat{w}_e$ be the realized weight of edge $e$, with nominal weight $w_e$, and maximum discount $0 \leq d_e \leq w_e$. Let $\hat{w}_e \equiv w_e + d_e\alpha_e$, where $\alpha_e$ is the *fractional deviation* of edge $e$. Both $\alpha_e$ and $\hat{w}_e$ are continuous random variables, symmetrically distributed on $[-1,1]$ and $[w_e - d_e, w_e + d_e]$ respectively. Edge discount factors $d \in \mathbb{R}^{|E|}$ are exogenous, while $\alpha \in [0,1]^{|E|}$ are treated as variables.

Edge discount factor $d_e$ should reflect the level of uncertainty in edge $e$'s nominal weight, $w_e$. If $w_e$ is known exactly, then $d_e = 0$; if $w_e$ is very uncertain, then we might set $d_e = w_e$, or higher.

To vary the degree of uncertainty, we use an *uncertainty budget* $\Gamma$, which limits the total deviation from nominal edge weights. With our uncertainty model, it is natural to let $\Gamma$ limit the total fractional deviation of each edge weight—i.e., sum of all $\alpha_e$. This uncertainty set $\mathcal{U}_\Gamma^I$ is defined as:

$$\mathcal{U}_\Gamma^I = \left\{ \hat{w} \mid \hat{w}_e = w_e + d_e\alpha_e, |\alpha_e| \leq 1, \sum_{e \in E} |\alpha_e| \leq \Gamma \ \forall e \in E \right\}$$

For example if $\Gamma = 3$, there may be three edges with $|\boldsymbol{\alpha}_e| = 1$, or one edge with $|\boldsymbol{\alpha}_e| = 1$ and four edges with $|\boldsymbol{\alpha}_e| = 1/2$, and so on.

Choosing an appropriate $\Gamma$ is not straightforward. Matchings often use only a small fraction of the decision variables (e.g., transplant edges), and it is difficult to predict the size of the optimal matching. Intuitively, $\Gamma$ should reflect the size of the final matching: for example if we assume that half of any matching's edges will be discounted, then we should set $\Gamma \simeq |\boldsymbol{x}|/2$. Generalizing this concept, we define a *variable*-budget uncertainty set $\mathcal{U}^I_\gamma$, with budget function $\gamma(|\boldsymbol{x}|)$.

$$\mathcal{U}^I_\gamma = \left\{ \hat{\boldsymbol{w}} \mid \hat{\boldsymbol{w}}_e = \boldsymbol{w}_e + \boldsymbol{d}_e \boldsymbol{\alpha}_e, |\boldsymbol{\alpha}_e| \leq 1, \sum_{e \in E} |\boldsymbol{\alpha}_e| \leq \gamma(|\boldsymbol{x}|) \ \forall e \in E \right\}$$

Next, to define $\gamma$, we relate it to a much more intuitive parameter: the protection level $\epsilon$.

### 3.3.1 Uncertainty Budget and Protection Level

The protection level $\epsilon$ mediates between a completely conservative approach, and the non-robust approach: as $\epsilon \to 0$ the approach becomes more conservative, and $\epsilon = 1$ corresponds to a non-robust approach. In this section we relate $\gamma$ to $\epsilon$, beginning with the following Theorem 3.1.

**Theorem 3.1** (Adapted from Theorem 3 of [43]). *For a matching $\boldsymbol{x} \in \mathcal{M}$ with $|\boldsymbol{x}|$ edges, and uncertainty set $\mathcal{U}^I_\Gamma$, the probability that $\mathcal{U}^I_\Gamma$ contains the realized edge weights for matching $\boldsymbol{x}$ is bounded below by*

$$P(\hat{\boldsymbol{w}} \in \mathcal{U}^I_\Gamma) \geq 1 - B(|\boldsymbol{x}|, \Gamma),$$

*with*

$$B(|\boldsymbol{x}|,\Gamma) = \frac{1}{2^{|\boldsymbol{x}|}}\left((1-\mu)\binom{|\boldsymbol{x}|}{\lfloor\eta\rfloor} + \sum_{l=\lfloor\eta\rfloor+1}^{|\boldsymbol{x}|}\binom{|\boldsymbol{x}|}{l}\right),$$

*with* $\eta = (\Gamma + |\boldsymbol{x}|)/2$ *and* $\mu = \eta - \lfloor\eta\rfloor$.

That is, for some $\epsilon$, if $\Gamma$ is chosen such that $\epsilon = B(|\boldsymbol{x}|,\Gamma)$, then the inequality $P(\hat{\boldsymbol{w}} \in \mathcal{U}_\Gamma^I) \geq 1-\epsilon$ holds by Theorem 3.1. Next, we use this result to define a variable uncertainty budget function $\gamma$, using the intuitive definition introduced by Poss [249]: for matching $\boldsymbol{x} \in \mathcal{M}$ and protection level $\epsilon$, we find the minimum $\Gamma$ such that $B(|\boldsymbol{x}|,\Gamma) \leq \epsilon$. If this is not possible (i.e., the matching is too small, or $\epsilon$ is too small), then $\gamma = |\boldsymbol{x}|$. This budget function is defined as:

$$\beta(|\boldsymbol{x}|) = \begin{cases} |\boldsymbol{x}| & \text{if } \min\limits_{\Gamma>0}\{\Gamma \mid B(|\boldsymbol{x}|,\Gamma) \leq \epsilon\} \text{ is infeasible,} \\ \min\limits_{\Gamma>0}\{\Gamma \mid B(|\boldsymbol{x}|,\Gamma) \leq \epsilon\} & \text{otherwise.} \end{cases}$$

It may not be clear how to solve the edge weight robust problem with this variable uncertainty budget. We use the approach of Poss [249], which solves the variable-budget robust problem by solving several instances of the *constant*-budget robust problem; details of this approach can be found in Appendix A.1.4. Thus, to solve the variable-budget robust problem we first solve the constant-budget robust problem.

### 3.3.2 Constant-Budget Edge Weight Robust Approach

We now describe our approach to the constant-budget edge weight robust problem; a full discussion and derivation can be found in Appendix A.1. We need to solve Problem 3.1 with edge weight uncertainty set $\mathcal{U}_\Gamma^I$. This requires a minimization of the objective, over $\hat{\boldsymbol{w}} \in \mathcal{U}_\Gamma^I$, followed by a maximization over matchings in $\mathcal{M}$.

First we *directly minimize* the objective of Problem 3.1 over $\mathcal{U}_\Gamma^I$. That is, for any

matching $\boldsymbol{x} \in \mathcal{M}$, we find the minimum objective value for any realized edge weights in $\mathcal{U}_\Gamma^I$, denoted by $Z(\boldsymbol{x})$:

$$Z(\boldsymbol{x}) = \min_{\hat{\boldsymbol{w}} \in \mathcal{U}_\Gamma^I} \hat{\boldsymbol{w}}^\top \boldsymbol{x} \tag{3.2}$$

Thus, solving the robust problem corresponds to maximizing $Z(\boldsymbol{x})$ over all feasible matchings. Our approach to doing so is as follows. First, we linearize $Z(\boldsymbol{x})$ using several new variables and constraints; we then add these to an existing kidney exchange formulation [109]. The complete linear formulations of $Z(\boldsymbol{x})$ and Problem 3.1 are given in Appendix A.1.2. Our robust formulation is scalable—it has a polynomial count of variables and constraints, regardless of finite chain cap; on realistic exchanges it takes only a few seconds to solve. We demonstrate our method's impact on match composition in Section 3.5, and show how it effectively controls for the impact of robustness using protection level $\epsilon$.

## 3.4 Optimization in the Presence of Edge Existence Uncertainty

In this section we consider *edge existence uncertainty*, where an algorithmic match must be chosen before the full realization of edges is revealed. Algorithmically-matched transplants in a kidney exchange can fail before transplantation for a variety of reasons: a patient may become too ill to undergo transplantation, or pre-transplantation testing may reveal that a patient is incompatible with her planned donor kidney. Furthermore, some edges are more likely to fail than others (e.g.,

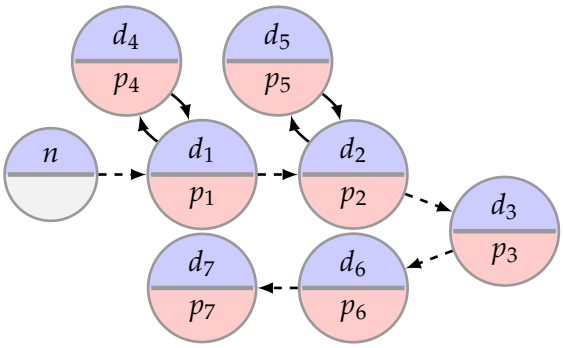



FIGURE 3.1: Sample exchange graph with a 5-chain and two 2-cycles. The NDD is denoted by $n$, and each patient (and her associated donor) is denoted by $p_i$ ($d_i$). A maximum-cardinality matching algorithm would select the 5-chain, denoted with dashed edges; however, the smaller matching consisting of two disjoint 2-cycles, shown with solid edges, may be more robust to edge failure.

edges into particularly sick patients). Edge failure significantly impacts fielded exchanges, with failure rates above 50% in many cases [16, 26, 110].

For illustration, consider the simple exchange in Figure 3.1 with two potential matchings: single 5-chain initiated by the NDD, or two 2-cycles (with pairs $\{1,4\}$ and $\{2,5\}$). The 5-chain matches the most patient, but is less robust to edge failures. Consider the *worst-case* outcome for each matching, when 1 edge is *guaranteed* to fail: with the 5-chain, in the worst-case the *first* edge fails, causing the entire chain to fail; with the 2-cycles, a single edge failure only causes a *single* cycle to fail, leaving the other cycle complete. With this notion of edge existence uncertainty (which we define later), the 2-cycles are more robust than the 5-chain.

Managing edge failure in kidney exchange has been addressed in the AI and optimization literature in application-specific [81, 210] or stochastic-optimization-based [16, 109, 110, 184] ways. These *failure-aware* approaches associate with each edge a pre-determined failure probability $\boldsymbol{p}_e$; these probabilities are used to then maximize *expected* matching score, possibly subject to some recourse actions. This stochastic approach is tractable when $\boldsymbol{p}_e$ is identical for each edge. Our work addresses two major drawbacks of the failure-aware approach. First, when each edge

has a unique $p_e$, those models require enumerating every cycle and chain, which is intractable for large graphs or long chains. Second, the failure-aware approach is very sensitive to $p_e$ (as discussed in, e.g., §4.4 of Dickerson et al. [110]). In practice, precise values of $p_e$ are not known, thus the failure-aware approach can easily produce unreliable results. We use a simpler notion of edge existence uncertainty, which assumes that in any matching, the number of edges is *bounded* by a constant ($\Gamma$). This parameter is intuitive and simple to estimate from past exchanges.

To our knowledge, ours is the first *scalable* robust optimization approach to edge existence uncertainty in kidney exchange. Glorie [146] develops several elegant robust methods for edge existence uncertainty, but requires that all cycles and chains are found during preprocessing and stored in memory. The number of chains grows exponentially in both the number of edges and the maximum chain length; thus, these approaches are intractable for exchanges involving more than a few dozen patient-donor pairs and NDDs.

**Edge existence uncertainty.** Here we briefly describe our robust approach to edge existence uncertainty; a full discussion and derivation can be found in Appendix A.2. For ease of exposition, in this section, decision variables $x \in \{0,1\}^{\mathcal{C}}$ correspond to cycles and chains rather than edges: $x_c$ is 1 if cycle or chain $c$ is matched, and 0 otherwise; in this section we use $\mathcal{C}$ to denote the set of all cycles and chains in $G$. We use the following model of edge existence uncertainty in the remainder of this chapter.

**Definition 3** ($\Gamma$-Failures Edge Existence Uncertainty)**.** Up to $\Gamma$ edges may fail in any matching. After failures occur, the realized exchange graph is $\hat{G} = (V, \hat{E})$, such that edges $\hat{E} \subseteq E$ proceed to transplantation, while all other edges do not.

With this notion of uncertainty, without regard to computational or memory constraints, a stochastic-optimization-based approach could identify the best matching

over all possible realizations $\hat{G}$ [16]. This is clearly intractable, as the number of realized graphs is exponential in $|E|$. Instead, we take a robust optimization approach by maximizing the worst-case (minimum) matching score over a set of realizable graphs $\hat{G}$ in an uncertainty set $\mathcal{U}$. Like the stochastic approach, the robust approach considers a huge number of realizations $\hat{G}$; however the robust approach is far more tractable, as it need only find the worst-case realization and need not represent all realizable graphs explicitly.

**Uncertainty set.** Let $F \subseteq E$ be the subset of failed edges for a realized graph $\hat{G}$; thus, $\hat{E} = E \setminus F$ is the set of realized edges. Equation 3.3 defines uncertainty set $\mathcal{U}_\Gamma^{ex}$ in this way: up to $\Gamma$ edges may fail (i.e., $|F| \leq \Gamma$).

$$\mathcal{U}_\Gamma^{ex} = \left\{\hat{G} = (V, \hat{E}) \mid \hat{E} = E \setminus F, |F| \leq \Gamma\right\} \tag{3.3}$$

In kidney exchange, one edge failure can cause other edge failures: if one cycle edge fails, all edges in the cycle also fail; edge failure in a chain causes all *subsequent* chain edges to also fail. This leads to a notion of weight uncertainty for cycles and chains, where the realized weight of a cycle or chain $\hat{\boldsymbol{w}}_c$ may be smaller than nominal weight $\boldsymbol{w}_c$. Let $\boldsymbol{\alpha}_c$ be a discount parameter for cycle or chain $c$, such that $\hat{\boldsymbol{w}}_c = \boldsymbol{w}_c(1 - \boldsymbol{\alpha}_c)$. For example, if any edge fails in cycle $c$, then the entire cycle fails and $\boldsymbol{\alpha}_c = 0$. We define the cycle/chain weight uncertainty set $\mathcal{U}_\Gamma^{w}$ in this way:

$$\mathcal{U}_\Gamma^{w} = \left\{\hat{\boldsymbol{w}} \mid \hat{\boldsymbol{w}}_c = \boldsymbol{w}_c(1 - \boldsymbol{\alpha}_c), \boldsymbol{\alpha}_c \in [0,1], \sum_{c \in X} \boldsymbol{\alpha}_i \leq \Gamma \forall c \in \mathcal{C}\right\}$$

This uncertainty set is less intuitive than $\mathcal{U}_\Gamma^{ex}$, but more suited to the robust approach. In Appendix A.2 we show that $\mathcal{U}_\Gamma^{w}$ and $\mathcal{U}_\Gamma^{ex}$ are equivalent for integer $\Gamma$, and thus can be used for our robust approach.

### 3.4.1 Robust Optimization Approach

In this section we briefly describe our robust approach; for a full discussion and derivation, please see Appendix A.2. Our robust formulation for uncertainty set $\mathcal{U}_\Gamma^w$ follows a similar approach to Section 3.3. First, we directly minimize the kidney exchange objective over $\mathcal{U}_\Gamma^w$, for some feasible solution $\boldsymbol{x} \in \mathcal{M}$. We express this minimization as a function $Z(\boldsymbol{x})$: in effect, $Z(\boldsymbol{x})$ discounts the $\Gamma$ largest-weight cycles and chains. We then linearize $Z(\boldsymbol{x})$ using several variables and constraints—this requires a formulation with variables tracking individual total chain weights—which is not possible in any existing compact kidney exchange formulations. For this purpose, we introduce a new kidney exchange formulation.

**The PI-TSP formulation.** We propose the position-indexed TSP formulation (PI-TSP); for details, please see Appendix A.2. Our formulation combines innovations from the two leading kidney exchange clearing approaches: PICEF [109] and PC-TSP [16]. PICEF introduced an indexing schema that enables a more compact formulation in the context of long chains; our formulation builds on this to allow tracking of individual chain weights, a necessity that PICEF could not do. PC-TSP builds on techniques from the prize-collecting traveling salesperson problem [34] to provide a tight linear programming relaxation; in general, the PC-TSP formulation has exponentially many constraints and thus requires constraint generation to solve. Our formulation uses an efficient version of position indexing that also requires only $O(|E|) + O(|V| \cdot |N|)$ constraints. Unlike PICEF, our formulation does not grow with the chain cap $L$: PICEF uses $O(|V|^3)$ variables (when $L \to |V|$); for large graphs, the PICEF model becomes too large to fit into memory [109]. Our formulation uses a fixed number of variables—$O(|V|^2)$—for any chain cap, alleviating this memory

problem. This is especially relevant to existing exchanges, as long chains can significantly increase efficiency in kidney exchange [25]. PI-TSP uses the following parameters:

- $G$: kidney exchange graph,
- $C$: a set of cycles on exchange graph $G$,
- $L$: chain cap (maximum number of edges used in a chain),
- $\boldsymbol{w}_e$: edge weights for each edge $e \in E$,
- $\boldsymbol{w}_c^C$: cycle weights for each cycle $c \in C$,

and the following decision variables:

- $\boldsymbol{p}_e \geq 1$: the position of edge $e$ in any chain,
- $\boldsymbol{p}_v \geq 1$: the position of patient-donor vertex $v$ in any chain,
- $\hat{\boldsymbol{p}}_e \geq 0$: equal to $\boldsymbol{p}_e$ if $e$ is used in a chain, and 0 otherwise,
- $\boldsymbol{z}_c \in \{0,1\}$: 1 if cycle $c$ is used in the matching, and 0 otherwise,
- $\boldsymbol{y}_e \in \{0,1\}$: 1 if edge $e$ is used in a chain, and 0 otherwise,
- $\boldsymbol{y}_e^n \in \{0,1\}$: 1 if edge $e$ is used in a chain starting with NDD $n$, and 0 otherwise,
- $\boldsymbol{w}_n^N$: total weight of the chain starting with NDD $n$,
- $\boldsymbol{f}_v^i$ and $\boldsymbol{f}_v^o$: chain flow into and out of vertex $v \in P$,
- $\boldsymbol{f}_v^{i,n}$ and $\boldsymbol{f}_v^{i,n}$: chain flow into and out of vertex $v \in P$, from the chain starting with NDD $n \in N$.

The PI-TSP formulation with chain cap $L$ is given in Problem 3.4. We use the notation $\delta^-(v)$ for the set of edges into vertex $v$ and $\delta^+(v)$ for the set of edges out of $v$.

$$\max \quad \sum_{n\in N} w_n^N + \sum_{c\in C} w_c^C z_c \tag{3.4a}$$

$$\text{s.t.} \quad \sum_{e\in E} w_e y_e^n = w_n^N \qquad n \in N \tag{3.4b}$$

$$\sum_{n\in N} y_e^n = y_e \qquad e \in E \tag{3.4c}$$

$$\sum_{e\in\delta^-(v)} y_e = f_v^i \qquad v \in V \tag{3.4d}$$

$$\sum_{e\in\delta^+(v)} y_e = f_v^o \qquad v \in V \tag{3.4e}$$

$$\sum_{e\in\delta^-(v)} y_e^n = f_v^{i,n} \qquad v \in V, n \in N \tag{3.4f}$$

$$\sum_{e\in\delta^+(v)} y_e^n = f_v^{o,n} \qquad v \in V, n \in N \tag{3.4g}$$

$$f_v^o + \sum_{c\in C: v\in c} z_c \leq \quad f_v^i + \sum_{c\in C: v\in c} z_c \leq 1 \qquad v \in P \tag{3.4h}$$

$$f_v^o \leq 1 \qquad v \in N \tag{3.4i}$$

$$p_e = 1 \qquad e \in \delta^+(N) \tag{3.4j}$$

$$\hat{p}_e = p_e y_e \qquad e \in E \tag{3.4k}$$

$$p_v = \sum_{e\in\delta^-(v)} \hat{p}_e \qquad v \in P \tag{3.4l}$$

$$p_e = p_v + 1 \qquad v \in P, e \in \delta^+(v) \tag{3.4m}$$

$$\sum_{e\in E} y_e^n \leq L \qquad n \in N \tag{3.4n}$$

$$f_v^{o,n} \leq f^{i,v} \leq 1 \qquad v \in V, n \in N \tag{3.4o}$$

$$y \in \{0,1\}^{|E|}, z \in \{0,1\}^{|C|} \tag{3.4p}$$

$$y^n \in \{0,1\}^{|E|} \qquad n \in N \tag{3.4q}$$

The ability to express individual chain weights as decision variables has applications beyond robustness. For particularly valuable NDDs (such as those with so-called "universal donor" blood-type O), exchanges may enforce a *minimum* chain length or chain weight, to ensure that these rare NDDs are not "used up" on short chains; such a policy was formerly used by the United Network for Organ Sharing [106], using a much less scalable form of optimization—that also does not consider uncertainty—than our approach [5]. Such a policy can be implemented efficiently with PI-TSP, inefficiently with PC-TSP, and not with PICEF, where decision variables do not indicate from which NDD a chain originated. In Appendix A.2 we show using real kidney exchange data that PI-TSP can enforce a minimum chain length, and that this restriction has *almost no* impact on overall matching score.

## 3.5 Experimental Results

In this section, we compare each robust formulation against the leading non-robust formulation, PICEF [109], with varying levels of uncertainty. These experiments use real exchange graphs collected from the United Network for Organ Sharing (UNOS)—a large US-wide kidney exchange with over 160 participating transplant centers—between 2010 and 2016, as well simulated exchanges generated from known patient statistics using the standard method [110].[2]

For each exchange, we calculate the optimal non-robust matching $M_{\text{OPT}}$ (with total score $|M_{\text{OPT}}|$), and the robust matching $M_{\text{R}}$, for varying uncertainty budgets. We then draw many *realizations* of the exchange graph, based on the uncertainty model, and calculate the realized scores of the robust matching $|M_{\text{R}}|$ and non-robust matching $|M_{\text{NR}}|$. We are primarily interested in the fractional difference from $|M_{\text{OPT}}|$,

[2] All experiments were implemented in Python and used Gurobi [155], a state-of-the-art industrial combinatorial optimization toolkit, as a sub-solver. Our code is available on GitHub: `https://github.com/duncanmcelfresh/RobustKidneyExchange`.

calculated as $\Delta OPT\left(M_{\{\mathrm{R,NR}\}}\right) = \left(|M_{\mathrm{OPT}}| - |M_{\{\mathrm{R,NR}\}}|\right)/|M_{\mathrm{OPT}}|$.

We calculate $\Delta OPT\left(M_{\mathrm{R}}\right)$ and $\Delta OPT\left(M_{\mathrm{NR}}\right)$ for $N = 400$ realizations, and compare the robust and non-robust approaches. In rare cases the optimal matching is empty (i.e., there is no solution, or the uncertainty budget exceeds the matching size), we exclude these exchanges from the results.

**Edge Weight Uncertainty** We begin by exploring the impact on match utility of robust approaches to managing edge weight uncertainty. Here, each edge is randomly labeled as *probabilistic* (P) or *deterministic* (D). P edges receive weight 0 or 1 with probability 0.5, while D edges always receive weight 0.5; thus, expected edge weight is always 0.5. The non-robust approach maximizes *expected* edge weight, making this a kind of stochastic approach. The robust approach considers the discount value (0 or 0.5) of each edge, and avoids edges with a positive discount value. To vary the level of uncertainty, we vary the fraction of P edges ($\alpha$). Each realization is drawn by assigning the P edges to have weight either 0 or 1.

We compute $M_{\mathrm{R}}$ for protection levels $\epsilon \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 0.5\}$, and then calculate both $\Delta OPT\left(M_{\mathrm{R}}\right)$ and $\Delta OPT\left(M_{\mathrm{NR}}\right)$. Figure 3.2 shows $\Delta OPT$ on realistic 64-vertex simulated graphs (left) and larger (typically 150–300-vertex) real UNOS graphs (right); these figures show results for each protection level $\epsilon$ and for various $\alpha$. Note that $M_{\mathrm{NR}}$ does not depend on $\epsilon$, and thus the non-robust results are shown as (constant) dashed lines.

The robust approach guarantees a better worst-case (minimum) $\Delta OPT$, but results in a lower median $\Delta OPT$. The protection level $\epsilon$ controls the robustness of our approach; smaller $\epsilon$ protects against more uncertain outcomes, but at greater cost to nominal behavior. As $\epsilon \to 1$, the robust approach protects against fewer bad outcomes, and approaches the behavior of non-robust.

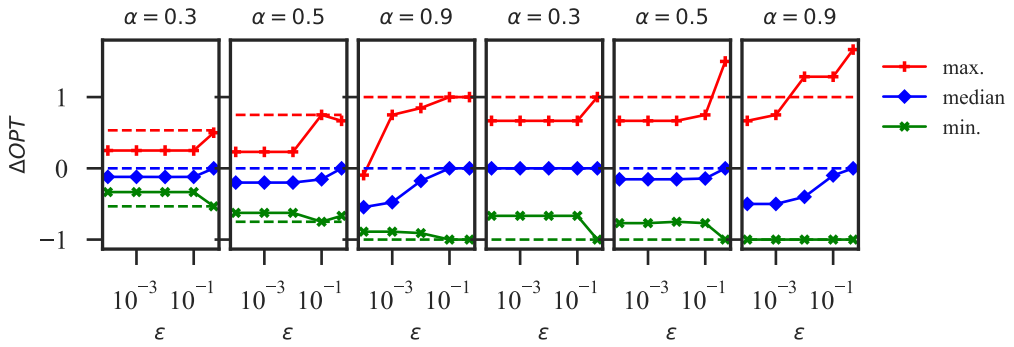



FIGURE 3.2: $\Delta OPT$ for non-robust (dashed lines) and edge weight robust (solid lines) matchings, for 64-vertex simulated exchange graphs (3 left plots) and real UNOS exchanges (3 right plots).

**Edge Existence Uncertainty** We now address edge existence uncertainty, and compare the robust and non-robust approaches with $\Gamma$ edge failures, for $\Gamma \in \{1, 2, 3, 4, 5\}$. Each $\Gamma$ corresponds to a different notion of uncertainty, such that exactly $\Gamma$ edges fail.[3] For each $\Gamma$, we calculate $M_R$, and draw $N = 400$ realizations by failing $\Gamma$ edges in the matching.

We calculate $\Delta OPT$ for each realization, and compare these results for the robust and non-robust matchings. With edge existence uncertainty, the worst-case outcome is almost always an empty matching ($\Delta OPT = -1$). Thus, rather than compare the worst-case $\Delta OPT$, we compare the *distribution* of $\Delta OPT$ for each approach: we treat $\Delta OPT$ as a random variable, and use three simple statistical tests to demonstrate that—as expected—the robust approach produces more conservative and predictable results.

First, we use the Wilcoxon signed-rank test to determine that the robust and non-robust approaches produce a different distribution of $\Delta OPT$. For each $\Gamma$, this test produces $p$-values well below $10^{-3}$, indicating that the distributions of $\Delta OPT$ *are different* for the robust and non-robust approach. Second, for all exchanges and all $\Gamma$, the mean $\Delta OPT$ is typically 1% *higher*, and the standard deviation 1–2% *lower*

[3] This is slightly more conservative than the notion of uncertainty introduced previously; in Section 3.4, *up to* $\Gamma$ edges may fail, while in the experiments *exactly* $\Gamma$ edges fail.

for the robust approach. That is, the robust approach more consistently produces higher-weight solutions.

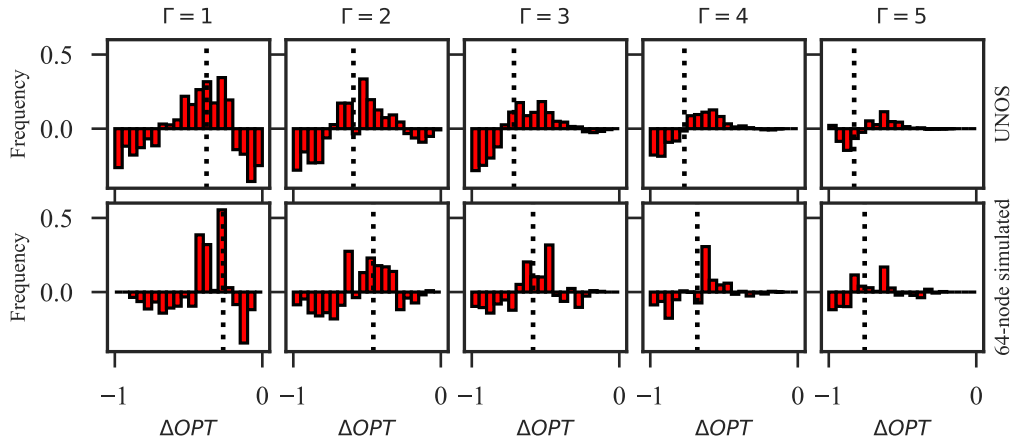



FIGURE 3.3: Difference between the robust and non-robust histograms of Δ*OPT* (robust minus non-robust) for real UNOS (top) and simulated exchanges (bottom), for various Γ. Dotted line: mean Δ*OPT* for non-robust.

Third, we visualize the difference between these distributions using their histograms. Figure 3.3 shows the bin-wise difference between the histograms of Δ*OPT* (robust minus non-robust), with mean Δ*OPT* for non-robust shown as a dotted line. In these plots, the height of the bars indicate the change in probability density due to robustness. On all plots, the bars are *negative* for high and low values of Δ*OPT*, meaning that the robust matching is *less likely* to have an abnormally high or low Δ*OPT*. The bars are *positive* when Δ*OPT* is near its mean non-robust value—meaning that the robust matching is *more likely* to have a Δ*OPT* near the mean non-robust value. This is exactly the desired behavior: robustness produces more predictable and less varied results. In this application robustness exceeds expectations: the robust approach achieves a lower variance, *and* slightly improves nominal performance.

## 3.6 Robustness as Fairness

Balancing efficiency and fairness is a classic economic problem; recently, a body of literature covering fairness in kidney exchange has developed in the AI/Economics [26, 108, 114, 215] and medical ethics [140] communities; Appendix A.3 presents a more thorough discussion. We now draw connections between robustness and fairness in kidney exchange. We show that budgeted edge weight uncertainty generalizes *weighted fairness* in kidney exchange, a generalization of "priority point" systems used in practice (see, e.g., [306]). Though seemingly unrelated, fairness and robustness share a key characteristic: the balance between two competing properties. Fairness rules in kidney exchange often mediate between a fair and efficient outcome, using a parameter to set the balance. Similarly, robustness mediates between a good nominal outcome with the worst-case outcome, using an uncertainty budget or protection level to set that balance.

In kidney exchange, fairness most often refers to the prioritization of both pediatric and *highly-sensitized* patients, who are unlikely to find a match due to medical characteristics that make them incompatible with nearly all potential donors. In the weighted fairness approach, edges that represent transplants to prioritized patients receive additional edge weight, making them more likely to be matched by standard algorithms; versions of this prioritization scheme are used by most exchanges, including UNOS. To generalize weighted fairness, let each edge have a *priority weight* $\hat{w}_e \in [0, \infty)$, equal to the nominal weight $w_e$ multiplied by a factor $(1 + \alpha_e)$, with $\alpha_e \in [-1, \infty)$. For example, we might set $\alpha_e > 0$ for all edges *into* prioritized patients; this will help prioritized patients, but will likely lower overall efficiency (a trade-off often described as the *price of fairness* [45, 72, 108, 215]).

To balance fairness with efficiency, policymakers limit the degree of prioritization. Let $\mathcal{P}_\Gamma$ be a *budgeted prioritization set*, which bounds the sum of absolute differences between each $w_e$ and $\hat{w}_e$; this prioritization set is given as:

$$\mathcal{P}_\Gamma = \left\{ \hat{w} \mid \hat{w}_e = w_e(1+\alpha_e), \alpha_e \geq -1, \sum_{e \in E} \alpha_e w_e \leq \Gamma \ \forall e \in E \right\}$$

As with edge weight uncertainty, the budget $\Gamma$ balances between fairness and efficiency. If $\Gamma$ is large, the algorithm might sacrifice matching size in order to match prioritized patients—but the maximum amount of efficiency sacrificed will be predictable, given $\Gamma$, which is attractive to policymakers. In Appendix A.3 we further develop this concept, propose fairness rules that use $\mathcal{P}_\Gamma$, and present some theoretical results regarding the balance between fairness and efficiency.

## 3.7 Discussion

In this chapter, we presented the first *scalable* robust formulations of kidney exchange. Our methods address both uncertainty over the *quality* and the *existence* of a potential transplant. On real and simulated data from a large, fielded kidney exchange, we showed that our methods (i) clear the market within seconds and (ii) result in more predictable and better quality matchings than the status quo.

Adapting automated ethical decision making frameworks that aggregate noisy human value judgments [58, 132, 229] into our robust formulation is a natural way to handle uncertainty in the weights determined by a committee of stakeholders.

## 3.8 Authors and Publication

This chapter was written by Duncan McElfresh, Hoda Bidkhori, and John P Dickerson. It appeared at the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19) [216].

# Chapter 4: CVaR for Edge Weight Uncertainty

## 4.1 Introduction

In Chapter 3 we addressed uncertainty in kidney exchange through *robust optimization*, which protects against the worst-case outcome. There are several reasons to use a robust approach: it is computationally tractable, it doesn't rely on statistical characterizations of uncertainty, and it is very risk-averse. However, the robust approach can often be *too* conservative. In this chapter we take a different approach to edge weight uncertainty in kidney exchange, which moderates risk aversion using a Conditional Value-at-Risk (CVaR) objective.

One major difference between this approach and the robust approach is assumptions of uncertainty: in the previous chapter we assumed *no knowledge* of the edge weight distribution. In this chapter we assume that the edge weight distribution is estimated from a small number of *measurements* of each edge weight.

**Contributions**

- We develop a risk-averse formulation for the KEP with respect to uncertainty in over the *quality* of the individual match (*edge weight uncertainty*) in the KEP. This model balances the *expected* edge weight value of the solution—the focus of classical models [82] as well as state-of-the-art kidney-exchange-specific models [109]—and the expected lowest $\alpha$% total weight. Solving this model

exactly is intractable, so we use a Sample-Average Approximation (SAA) approach. We demonstrate in computational simulations that our SAA methods effectively balances risk in the resulting KEP solution, over state-of-the-art baseline methods.

## 4.2 Characterizing Edge Weight Uncertainty in Kidney Exchange.

In the experiments for this chapter, we use a common notion of edge weights that reflect the *survival time* of the transplanted organ [211, 306]. This is the approach taken by Li et al. [202], who use the Living Donor Kidney Profile Index (LDKPI)—a score based on the estimated survival time of a transplanted kidney in the donor. These scores are estimated from past transplants (e.g., using Cox survival analysis), and are thus inherently uncertain.[1] In line with the assumptions of the LDKPI, for our experiments we assume that each edge weight is drawn from an exponential distribution that depends on patient and donor characteristics; we use this distributional assumption to define an edge weight distribution for our experiments in Section 4.4.

### 4.2.1 Risk Measures & Assessment

In this chapter we incorporate Conditional Value-at-Risk (CVaR) into the KEP.

CVaR is one of the most common risk measures with applications in support vector machines (SVM), reinforcement learning, financial management, and many other areas. However, the CVaR approach is often mathematically tractable [265]; and is also the case in the CVaR approach to the KEP. In this chapter we propose

[1] For additional information and a real LKDPI calculator, please see http://www.transplantmodels.com/LDKPI/.

one complementary approach to solving our CVaR-based KEP model, based on the sample average approximation.

**Stochastic programming via SAA** Our first approach uses sample average approximation (SAA), which was first studied by Shapiro and Homem-de Mello [283] and Kleywegt et al. [183]. SAA approximates the unknown distribution through empirical distributions. If the data samples are drawn from the *true* unknown distribution, the optimal solution of SAA converges to the true optimal with probability 1; however, if only a small number of samples are available, the performance of the obtained solution is often poor.

## 4.3 A CVaR Model for the KEP

Our kidney exchange model balances the expected outcome with a worst-case outcome, using a conditional value-at-risk (CVaR) approach, which we describe in detail later in this section. In contrast, most kidney exchange models focus on the *expected* matching weight (e.g., [16, 110, 184]) or the *worst-case* outcome (e.g., [75, 216]). To our knowledge, the only prior work that considers a CVaR model of kidney exchange are due to Zheng et al. [332], which does not limit the cycle or chain length (and for this reason, they can use a min-cost flow approach), and Bidkhori et al. [46], which does not consider uncertainty in edge weights.

To account for edge weight uncertainty, we propose the following objective function for kidney exchange:

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

where $\mu$ represents the *expected* edge weight of the matching, and $\mu_\alpha$ represents $\alpha$% *worst-case* mean matching weight (i.e., the mean of all possible edge weights from

the 0th through the $\alpha$th percentile). Compared to the standard model, the second term is new and measures the variance over different match weights. The positive parameter $\gamma$ controls the trade-off between the average performance and the risk of the solution: $\gamma = 0$ will optimize only the mean matching weight, while $\gamma = 1$ will optimize only the $\alpha$% worst-case mean. Both $\alpha$ and $\gamma$ are exogenous parameters (chosen by the user) to reflect their desired level of risk aversion.

**The PICEF Formulation** Our CVaR model for the KEP is based on the PICEF mixed-integer formulation for kidney exchange, of Dickerson et al. [109], and we define the PICEF constraints here for convenience. This formulation uses two sets of decision variables, $\boldsymbol{x} \in \{0,1\}^{|C|}$ for cycles and $\boldsymbol{y} \in \{0,1\}^{|N|\times L}$ for chains, where $C$ is the set of all cycles in $G$, $N$ is the set of all non-directed donors, and $L$ is the chain cap. Consistent with most modern KEP formulations, we assume that all cycles are identified ahead of time, and no additional constraints are required for cycle: $\boldsymbol{x}_c$ is 1 if cycle $c$ is matched, and 0 otherwise. On the other hand, chain variables are defined as follows: $\boldsymbol{y}_{ek}$ is 1 if edge $e$ is matched at the $k$th position of any chain, and 0 otherwise. To define these variables, we use three additional quantities:

- $\delta^+(i) \subseteq P$ for each $i \in V$ is the set of *incoming* edges to vertex $i$,
- $\delta^-(i) \subseteq P$ for each $i \in V$ is the set of *outgoing* edges to vertex $i$, and
- $\mathcal{K}(e)$ is the set of positions that $e$ may take in a chain. If edge $e$ originates with a NDD, then $\mathcal{K}(e) = \{1\}$; otherwise, $\mathcal{K}(e) = \{2, \ldots, L\}$.

Using these decision variables and functions, the set of feasible matchings in the PICEF formulation is defined

$$\mathcal{M} \equiv \begin{cases} \sum_{e \in \delta^-(i)} \sum_{k \in \mathcal{K}(e)} y_{ek} + \sum_{c \in C : i \in c} z_c \leq 1 & \forall i \in P \\ \sum_{e \in \delta^-(i) \wedge k \in \mathcal{K}(e)} y_{ek} \geq \sum_{e \in \delta^+(i)} y_{e,k+1} & \forall i \in P, k \in \{1, \ldots, L-1\} \\ \sum_{e \in \delta^+(i)} y_{e1} \leq 1 & \forall i \in N\,', \\ y_{ek} \in \{0,1\}, & \forall e \in E, k \in \mathcal{K}(e) \\ z_c \in \{0,1\} & \forall c \in C, \end{cases} \tag{4.1}$$

The first constraint ensures that each patient-donor pair vertex is matched at most once; the second constraint requires each patient-donor pair can only be matched with an outgoing chain edge if they are also matched with an incoming chain edge; the third constraint requires that each NDD is matched at most once.

**CVaR Formulation of the KEP** While it is common in the literature to represent kidney exchange as a maximization problem (with a non-negative objective), we instead formulate it as minimization problem (with non-positive objective) to be consistent with the convention of risk-minimization. Using this convention, the average of the $\alpha$% worst (highest) losses is known as the Conditional Value-at-Risk (CVaR) [265] at level $\alpha$. To represent the objective we use auxiliary variables to represent the *negative* matching indicators for each edge, defined as

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

where $\mathbf{1}(e \in c)$ is 1 if edge $e$ is in cycle $c$, and 0 otherwise. Finally, we formulate the CVaR model of KEP as

$$\begin{aligned} \min \quad & \mathbb{E}\left[\boldsymbol{w}^\top \boldsymbol{m}\right] + \gamma \text{CVaR}_\alpha\left[\boldsymbol{w}^\top \boldsymbol{m}\right] \\ s.t. \quad & \boldsymbol{m}_e = -\sum_{k \in \mathcal{K}(e)} \boldsymbol{y}_{ek} - \sum_{c \in C} \mathbf{1}(e \in c)\boldsymbol{z}_c \quad \forall e \in E \\ & \boldsymbol{m} \in \mathbb{R}^{|E|} \\ & \{\boldsymbol{y}, \boldsymbol{z}\} \in \mathcal{M}, \end{aligned} \tag{4.2}$$

where the expectation is over is the (known) distribution of edge weights.

The objective of Problem 4.2 minimizes a weighted sum of the mean and $\text{CVaR}_\alpha$ of the loss (negative weight) of a matching, where $\alpha \in (0, 100]$ is the confidence level of $\text{CVaR}_\alpha$. The parameter $\gamma \geq 0$ is set by the user, and defines the trade-off between the mean and $\text{CVaR}_\alpha$: $\gamma = 0$ means that this model only optimizes the expected matching weight, and $\gamma > 0$ includes the CVaR objective. We reformulate $\text{CVaR}_\alpha$ by introducing an auxiliary variable $d$, as in [265]. Using this reformulation,Problem 4.2 is equivalent to

$$\begin{aligned} \min \quad & \mathbb{E}\left[\boldsymbol{w}^\top \boldsymbol{m}\right] + \gamma\left(d + \frac{1}{\alpha}\mathbb{E}\left[(\boldsymbol{w}^\top \boldsymbol{m} - d)^+\right]\right) \\ s.t. \quad & \boldsymbol{m}_e = -\sum_{k \in \mathcal{K}(e)} \boldsymbol{y}_{ek} - \sum_{c \in C} \mathbf{1}(e \in c)\boldsymbol{z}_c \quad \forall e \in E \\ & \{\boldsymbol{y}, \boldsymbol{z}\} \in \mathcal{M} \\ & \boldsymbol{m} \in \mathbb{R}^{|E|} \\ & d \in \mathbb{R}, \end{aligned} \tag{4.3}$$

where $(\cdot)^+$ denotes the positive part, and as before all expectations are taken over a

known edge weight distribution. Problem (4.3) is not analytically solvable, for a general edge weight distribution. Therefore, we propose a sample average approximation-based (SAA) approach to solve problem (4.3).

### 4.3.1 Solving Problem 4.3 with SAA

In this section, we assume the available edge weight measurements are representative of the *true* edge weight distribution. Our approach uses these measurements to approximately[2] solve Problem 4.3. Instead we use a *sample average approximation* (SAA), which replaces the exact expectations of Problem 4.3 with sample-averages. For this we assume there are $J$ *measurements* of each edge's weight, independently drawn from the edge weight distribution; we denote the $j$th measurement as $\hat{\boldsymbol{w}}^j$, for $j \in \{1, \ldots, J\}$. The SAA of Problem 4.3 is expressible as a mixed-integer linear program, given in Proposition 4.1.

**Proposition 4.1.** *The SAA of (4.3) is equivalent to the following mixed-integer program.*

$$
\begin{aligned}
\min \quad & \frac{1}{J}\sum_{j=1}^{J} \boldsymbol{r}_j + \gamma d \\
s.t. \quad & \boldsymbol{m}_e = -\sum_{k \in \mathcal{K}(e)} \boldsymbol{y}_{ek} - \sum_{c \in C} \mathbf{1}(e \in c)\boldsymbol{z}_c && \forall e \in E \\
& \boldsymbol{r}_j \geq (\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} && \forall j \in \{1, \ldots, J\} \\
& \boldsymbol{r}_j \geq \left(1 + \frac{\gamma}{\alpha}\right)(\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} - \frac{d\gamma}{\alpha} && \forall j \in \{1, \ldots, J\} \\
& \{\boldsymbol{y}, \boldsymbol{z}\} \in \mathcal{M} \\
& \boldsymbol{r} \in \mathbb{R}^J \\
& \boldsymbol{m} \in \mathbb{R}^{|E|} \\
& d \in \mathbb{R},
\end{aligned}
\tag{4.4}
$$

[2] If there is sufficient data to accurately characterize the edge weight distribution, we might instead try to solve Problem 4.3 exactly. However, even in this case, Problem 4.3 is not analytically solvable.

*Proof.* We begin with the SAA of Problem 4.3, written explicitly as follows

$$\begin{aligned}
\min \quad & \frac{1}{J}\sum_{j=1}^{J}(\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} + \gamma\left(d + \frac{1}{\alpha J}\sum_{j=1}^{J}\left((\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} - d)^+\right)\right) \\
s.t. \quad & \boldsymbol{m}_e = -\sum_{k\in\mathcal{K}(e)} \boldsymbol{y}_{ek} - \sum_{c\in C}\mathbf{1}(e\in c)\boldsymbol{z}_c \qquad \forall e \in E \\
& \{\boldsymbol{y}, \boldsymbol{z}\} \in \mathcal{M} \\
& \boldsymbol{m} \in \mathbb{R}^{|E|} \\
& d \in \mathbb{R}.
\end{aligned}$$

Next, we rewrite the objective of this problem using the following steps

$$\frac{1}{J}\sum_{j=1}^{J}(\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} + \gamma\left(d + \frac{1}{\alpha J}\sum_{j=1}^{J}\left((\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} - d)^+\right)\right) = \ldots$$

$$\begin{aligned}
&= \frac{1}{J}\sum_{j=1}^{J}\left((\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} + \frac{\gamma}{\alpha}\left((\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} - d\right)^+\right) + \gamma d \\
&= \frac{1}{J}\sum_{j=1}^{J}\left((\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} + \frac{\gamma}{\alpha}\max\left((\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} - d, 0\right)\right) + \gamma d \\
&= \frac{1}{J}\sum_{j=1}^{J}\max\left(\left(1+\frac{\gamma}{\alpha}\right)(\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} - \frac{d\gamma}{\alpha}, (\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m}\right) + \gamma d
\end{aligned}$$

Next we introduce auxiliary variables $\boldsymbol{r} \in \mathbb{R}^J$, defined as follows:

$$\boldsymbol{r}_j \equiv \max\left(\left(1+\frac{\gamma}{\alpha}\right)(\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} - \frac{d\gamma}{\alpha}, (\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m}\right), \quad \forall j \in \{1, \ldots, J\}.$$

To define these variables in a MIP we use the following constraints for each $j \in \{1, \ldots, J\}$.

$$\begin{aligned} \boldsymbol{r}_j &\geq (\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} \\ \boldsymbol{r}_j &\geq \left(1 + \frac{\gamma}{\alpha}\right) (\hat{\boldsymbol{w}}^j)^\top \boldsymbol{m} - \frac{d\gamma}{\alpha}. \end{aligned}$$

Adding auxiliary variables $\boldsymbol{r}$ to the SAA formulation yields the MIP in Proposition 4.1. □

## 4.4 Experiments

We use 32 randomly-generated exchange graphs resembling the structure of real exchanges, using anonymized data from the United Network for Organ Sharing (UNOS), a US-based kidney exchange [253]. To reflect edge weight uncertainty we define an edge weight distribution inspired by real sources of uncertainty in kidney exchange, based on transplant survival rate.

**Edge Weight Distribution: Survival Rate.** We define a simple edge weight distribution based on the Living Donor Kidney Profile Index (LDKPI), a recent metric due to Massie et al. [211] that is now used as a decision support tool for kidney transplantation. Using the method of Li et al. [202], we assume that the survival rate of a donor organ is an exponential random variable with mean proportional to the LDKPI. We define a simple distribution by assuming that edge weights depend *only* on the donor LDKPI.[3] Each donor node is randomly assigned to have *low LDKPI* (14.93) or *high LDKPI* (59.37), each with probability 1/2; this corresponds to the mean LDKPI plus/minus one standard deviation of the distribution estimated by Saidman et al.

[3] In reality, LDKPI also weakly depends on the recipient, including recipient age, if the donor and recipient are related, and a few additional donor-recipient factors [211].

[273]. After assigning an LDKPI for each donor, we draw edge weight (i.e., survival time) from the exponential distribution defined in [202]: $w_e \sim 14.78e^{-0.01239 \times \text{LDKPI}}$.

**Stochastic and Deterministic Edges.** In practice the variance of edge weight distributions can be very different: some edge weights may be very uncertain, while others may be nearly constant [211]. To simulate this, we randomly assign edges to be either *stochastic* (edge weights are drawn from its underlying distribution) or *deterministic* (the edge weight is equal to the *first* draw from its distribution). Note that we always define an edge distribution, but deterministic edges only use a single draw from this distribution. In our experiments, half of all edges are deterministic while the other half are probabilistic.

### 4.4.1 Results

In this setting, $J$ edge weight measurements are made of each edge, and we assume that these measurements come from the *correct* edge weight distribution. We compare three methods that use these $J$ measurements: our SAA-CVaR approach, a state-of-the-art Robust Optimization approach [216], and a non-robust matching algorithm which maximizes the sample mean edge weight. We show that SAA-CVaR has better $\alpha$% worst-case performance than both other methods; The main results are summarized in Figure 4.1.

**Implementation details** We denote the SAA-CVaR approach of Section 4.3.1 as `SAA`. For each of the 32 exchange graphs, we simulate the LDKPI edge weight distribution; we simulate 4 different edge weight distributions (randomly) for each graph (in effect, creating 4 different graph instances, each sharing the same underlying structure). For each graph-distribution pair we simulate $J = 200$ draws (i.e., *measurements*) from the edge weight distribution, and use these measurements to find

optimal matchings using each of the following methods: NR: maximize the sample mean edge weight (where the sample mean is calculated over all $J$ simulated weight measurements), RO the robust optimization approach of [216], and SAA, our proposed approach. For RO we use parameter $\Gamma = 5$, meaning that up to 5 edges may have realized weight of 0. For SAA we set $\alpha$ to 50, meaning that the this approach aims to maximize realized matching weight over the 50% worst-case outcomes.To vary the balance between the mean matching weight and the worst-case objective we vary parameter $\gamma$, with $\gamma \in \{0.1, 1.0, 10.0, 100.0\}$. With $\gamma = 0$, SAA only optimizes the expected matching weight; as $\gamma$ increases, the objective is weighted more toward the $\alpha$% worst-case outcomes.

**Metrics.** To test the quality of these methods, we simulate 1000 *realizations* of each edge weight. Our approach is designed to protect against the $\alpha$% worst-case outcomes; so, we compare the means of the lowest $\alpha$% realized matching weights for each method. We use the following metric to compare each method

$$\text{FRAC-NR} \equiv (\mu_\alpha^{\texttt{X}} - \mu_\alpha^{\texttt{NR}})/\mu_\alpha^{\texttt{NR}},$$

where $\mu_\alpha^{\texttt{X}}$ is the mean of the $\alpha$% lowest-weight realizations for method X. We use the non-robust method NR as a baseline to compare the previous robust-optimization-based method RO with our proposed method SAA. In many cases the robust methods identified the same optimal matching as non-robust; in order to highlight the differences between robust and non-robust, we remove these cases from our results (i.e., the cases where FRAC-NR $= 0$).

**Results.** Figure 4.1 shows the simulation results. Recall that parameter $\gamma$ moderates between the *expected* edge weight and the CVaR objective (to maximize the $\alpha$%

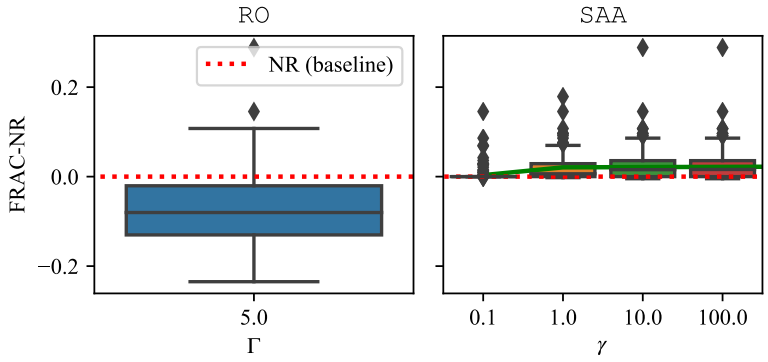



FIGURE 4.1: FRAC-NR for both the RO method due to McElfresh et al. [216] and our SAA method, for $\gamma \in \{0.1, 1.0, 10.0, 100.0\}$.

worst-case edge weight); as demonstrated in Fig. 4.1, when $\gamma$ is small the behavior of SAA is similar to non-robust (i.e., FRAC-NR is almost always zero). With larger $\gamma$, SAA prioritizes the CVaR objective, and achieves a higher weight for $\alpha$% worst-case weight (i.e., FRAC-NR is positive). Note that the robust optimization approach of McElfresh et al. [216] (RO) is far too conservative, and in fact achieves lower CVaR objective than non-robust (i.e., FRAC-NR is negative).

### 4.4.2 Comparisons of the Structures in Matchings

Here we provide two tables detailing the differences in the matchings produced by each method in the experiments of the last two subsections. In short, these tables demonstrate robust methods tend to avoid *uncertain* edges, even if this results in a lower average matching weight.

**Matched Edges.** First, we present the number of each *type* of edge matched by each method. Our experiments use four different edge types, which are randomly assigned based on each edge's donor, which we summarize as; high-weight deterministic (D-High), high-weight stochastic (S-High), low-weight deterministic (D-Low), and low-weight stochastic (S-Low). Both D-High and S-High have mean weight 12.3, while D-Low and S-Low have mean weight 7.1. However, both D-High and

D-Low are exponentially distributed, meaning that the worst-case weight can be quite low (approaching zero) in any particular realization. Our CVaR approach is designed to balance between the mean matching weight, and the worst-case weight in $\alpha\%$ of all possible outcomes. Thus, we expect that robust optimization approaches should favor deterministic edges over probabilistic edges. Table 4.1 shows the number of each edge matched by each method, compared with non-robust.

| **Method** | **D-High** | **S-High** | **D-Low** | **S-Low** |
|---|---|---|---|---|
| Non-Robust (baseline) | 699 (0) | 700 (0) | 555 (0) | 517 (0) |
| SAA ($\gamma = 0.1$) | 706 (+7) | 693 (-7) | 558 (+3) | 514 (-3) |
| SAA ($\gamma = 1.0$) | 735 (+36) | 661 (-39) | 562 (+7) | 514 (-3) |
| SAA ($\gamma = 10.0$) | 747 (+48) | 637 (-63) | 571 (+16) | 514 (-3) |
| RO ($\gamma = 5$) | 640 (-59) | 683 (-17) | 568 (+13) | 580 (+63) |

TABLE 4.1: Total number of edges of each type matched by each method. The difference between each method and non-robust is indicated in parentheses.

Intuitively, most robust methods match more deterministic edges (D-High and D-Low), and fewer stochastic edges (S-High and S-Low) than non-robust. Notably, SAA tends to match *more* deterministic edges as $\gamma$ increases. It is also notable that RO does not strictly follow this trend. This behavior helps explain Figure 4.1 in contrast to SAA, method RO is selecting edges with uncertain edge weights.

**Cycle and Chain Structure.** Next, we analyze the overall matching structure. In kidney exchange, the solution (matching) is composed of cycles and chains of a fixed length. The composition of each matching can be a useful indicator of the quality or robustness of a solution. Table 4.2 shows the total number of cycles and chains of each *length* matched by each method, over all exchange graphs.

There is little difference between the *types* of structures produced by each method—i.e., the resulting cycle and chain lengths are very similar. As in the case of edge *existence* uncertainty, where risk-averse methods will select smaller structures due to

| Method | 2-cycles | 3-cycles | 1-chains | 2-chains | 3-chains | 4-chains |
|---|---|---|---|---|---|---|
| Non-Robust (baseline) | 198 (0) | 433 (0) | 48 (0) | 50 (0) | 52 (0) | 118 (0) |
| SAA ($\gamma = 0.1$) | 208 (+10) | 428 (-5) | 49 (+1) | 44 (+1) | 58 (+6) | 115 (-3) |
| SAA ($\gamma = 1.0$) | 203 (+5) | 429 (-4) | 44 (-4) | 51 (-4) | 59 (+7) | 114 (-4) |
| SAA ($\gamma = 10.0$) | 206 (+8) | 425 (-8) | 44 (-4) | 47 (-4) | 56 (+4) | 119 (+1) |
| RO ($\gamma = 5$) | 215 (+17) | 418 (-15) | 44 (-4) | 51 (-4) | 43 (-9) | 128 (+10) |

TABLE 4.2: Total number of cycles and chains of each length matched by each method; difference between each method and non-robust is given in parentheses.

their decreased fragility to edge failure, we see that nearly all robust methods result in more 2-cycles and fewer 3-cycles than non-robust, due to the decreased variance in weight for smaller structures.

## 4.5 Discussion

We proposed data-driven methods to solve the mean-risk KEP (4.2). On realistic data drawn from a large, fielded US-based kidney exchange, we validated that our methods strike a balance between protecting against worst-case realizations and maintaining strong average-case performance. In many matching applications—including kidney allocation-it is likely that uncertainty (over not just match quality, as we consider in this chapter, but also match existence) is correlated, due to medical and/or non-medical reasons, with sensitive attributes such as race; past work has considered this in the deterministic setting [126, 185, 215], and valuable future work could extend those approaches to our setting.

## 4.6 Authors and Publication

This chapter was written by Duncan McElfresh, Ke Ren, John P Dickerson, and Hoda Bidkhori. A full version of this chapter will appear at the 2021 Winter Simulation Conference (WSC-21). [219]

# Chapter 5: Dealing with Edge Existence Uncertainty in the KEP

In this chapter we build on the ideas presented in the two previous chapters to comprehensively address edge existence uncertainty in the KEP. First 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 in the KEP, with *non-identical* edge failure probabilities. This is an important step forward, as failure probabilities are known to be inhomogeneous–some edges are inherently riskier than others [110]. We provide a mixed-integer linear program for our approach, which is compact (for a fixed cycle cap $K$) 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 (UNOS).

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 § 5.2.1).

## 5.1 Prior Work on Edge Existence Uncertainty

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. [109] provides a scalable formulation in this case, and Dickerson et al. [110] 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 [13, 90]. Zheng et al. [332] 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 [16], forms of "fallback" options [38, 210, 314], and pre-match edge queries [54, 55, 214]. 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.

## 5.2 Edge Existence Uncertaianty Model

We assume that edge failure probabilities $\boldsymbol{p} \in [0,1]^{|E|}$ are known in advance and are not necessarily homogeneous. That is, if edge $e = (v_i, v_j)$ is matched, then with probability $\boldsymbol{p}_e$ the patient of $v_j$ would still fail to receive a kidney from $v_i$'s donor. 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, and we demonstrate this with a small example.

### 5.2.1 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 5.1, 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 [16, 305]) 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. [110]), and select 2-cycle $(1, 3)$. Indeed this matching achieves total expected weight 6.23, nearly twice the expected weight of cycle $(1, 2)$. Of course, cycle $(1, 3)$ has a significant (roughly 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*

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

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

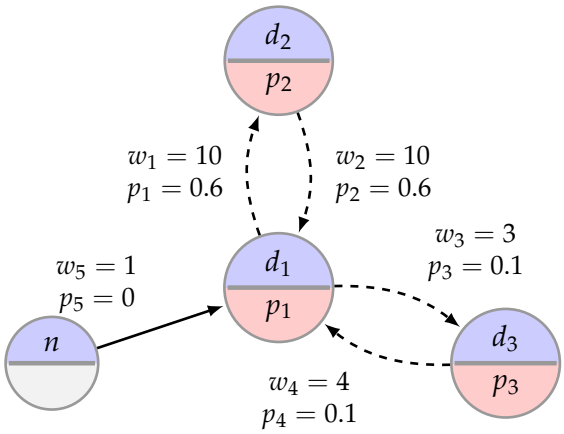



FIGURE 5.1: Example exchange graph with a single NDD $n$, and three patient-donor pairs; weights $w$ and failure probabilities $p$ are shown for each edge. The max-weight matching is the cycle between pairs 1 and 2; the max-expected-weight matching is the cycle between pairs 1 and 3, and the risk-averse/robust optimal matching is any the chain beginning with the edge from $n$ to pair 1.

outcome (e.g., as in McElfresh et al. [216]). 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.

## 5.3 Maximizing Expected Matching Weight with Inhomogeneous Edge Existence Uncertainty

We are primarily interested in maximizing the *expected* weight of a matching; indeed this is the focus of most prior work (see § 5.1). 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.

**Expected weight of a cycle** The expected weight of a $k$-cycle $c$ reflects the fact that the *whole cycle* will fail if any single transplant fails. The expected weight of a cycle

$u(c)$ is expressible in terms of the failure probabilities of its edges:

$$u(c) = \left(\sum_{e \in c} \boldsymbol{w}_e\right) \left[\prod_{e \in c}(1 - \boldsymbol{p}_e)\right].$$

**Expected weight of a chain** We denote a $k$-chain in terms of its constituent edges, $\kappa \equiv (e_1, ..., e_k)$, where $e_1$ originates with a non-directed donor (NDD). The expected weight of chain $\kappa$ is expressed as:

$$u(\kappa) = \sum_{i=2}^{k} \boldsymbol{p}_i \left(\sum_{j=1}^{i-1} \boldsymbol{w}_j\right) \prod_{j=1}^{i-1}(1 - \boldsymbol{p}_j) + \left(\sum_{i=1}^{k} \boldsymbol{w}_i\right) \prod_{i=1}^{k}(1 - \boldsymbol{p}_i). \tag{5.1}$$

In the above expression $\boldsymbol{p}_i$ and $\boldsymbol{w}_i$ denote the failure probability and weight of the $i$th edge in the chain. The first term above is the sum of expected weights for the chain executing exactly $i - 1 = \{1, ..., k - 1\}$ edges and then failing on the $i$th edge. 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 $(\boldsymbol{C}, \boldsymbol{K}) \in \mathcal{M}$ denote a feasible matching consisting of cycles $\boldsymbol{C}$ and chains $\boldsymbol{K}$. Problem 5.2 is an equivalent formulation of the stochastic KEP.

$$\max_{(\boldsymbol{C}', \boldsymbol{K}') \in \mathcal{M}} \quad \sum_{c' \in \boldsymbol{C}} u(c') + \sum_{\kappa' \in \boldsymbol{K}} u(\kappa') \tag{5.2}$$

Next we describe our solution approach for Problem 5.2, and an equivalent compact mixed-integer linear program formulation.

### 5.3.1 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. Here, *compact* means that the counts of variables and constraints are polynomial in the size of the input, for a fixed cycle

cap $K$. Prior to this work there was no scalable solution for Problem 5.2. For example, in [110] the authors propose a solution approach 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.

Here we propose an exact, compact representation for Problem 5.2 using an equivalent expression for expected chain weight $u(\kappa)$ given in Lemma 5.1.

**Lemma 5.1.** *The expected weight $u(\kappa)$ of the k-chain $\kappa = (e_1, ..., e_k)$ is*

$$u(\kappa) = \sum_{i=1}^{k} \boldsymbol{w}_i \prod_{j=1}^{i} \left(1 - \boldsymbol{p}_j\right),$$

*where $\boldsymbol{w}_i$ and $\boldsymbol{p}_i$ are the edge weight and failure probability of the $i^{th}$ edge in the chain.*

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

$$\begin{aligned} u(k) = &\sum_{i=2}^{k} p_i (\sum_{j=1}^{i-1} w_j) \prod_{j=1}^{i-1} (1 - p_j) \\ &+ (\sum_{i=1}^{k} w_i) \prod_{i=1}^{k} (1 - p_i). \end{aligned}$$

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,

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

□

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

The objective of Problem 5.3 uses Lemma 5.1 to compactly 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, $\boldsymbol{o}_{ek}$ (the discount factor of edge $e$ at position $k$ in a chain) and $\boldsymbol{v}_c$ (the success probability of cycle $c$). Problem 5.3 uses the following parameters:

- $G = (E, V)$: kidney exchange graph
- $C$: all legal cycles in exchange graph $G$
- $L$: chain cap (max. number of edges in a chain)
- $\boldsymbol{w}_e$: edge weights for each edge $e \in E$
- $\boldsymbol{w}_c$: cycle weights for each cycle $c \in C$, defined as $w_c = \sum_{e \in c} \boldsymbol{w}_e$
- $\delta^-(i)$: the set of edges into vertex $i$
- $\delta^+(i)$: the set of edges out of vertex $i$
- $\mathcal{K}(e)$: the set of legal positions in a chain that edge $e$ can occupy (see Section 4.3)
- $\boldsymbol{p}_e$: failure probability for edge $e \in E$

The following decision variables are used. Both $\boldsymbol{z}$ and $\boldsymbol{y}$ are borrowed from the PICEF formulation of Dickerson et al. [109].

- $\boldsymbol{z}_c \in \{0, 1\}$: 1 if cycle $c$ is used in the matching, and 0 otherwise
- $\boldsymbol{y}_{ek} \in \{0, 1\}$: 1 if edge $e$ is used at position $k$ in a chain, and 0 otherwise
- $\boldsymbol{o}_{ek} \in [0, 1]$: discount factor of edge $e$ at position $k$ in a chain

The complete formulation for Problem 5.2 is given in Problem 5.3.

$$\max \quad \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} \boldsymbol{w}_e \boldsymbol{y}_{ek} \boldsymbol{o}_{ek} + \sum_{c \in C} \boldsymbol{w}_c \boldsymbol{z}_c \boldsymbol{v}_c \tag{5.3a}$$

$$s.t. \quad \sum_{e\in\delta^-(i)\wedge k\in\mathcal{K}(e)} o_{ek}y_{ek} \geq \sum_{e\in\delta^+(i)} \frac{o_{e,k+1}y_{e,k+1}}{1-p_e} \quad \forall i \in P, k \in \{1,\ldots,L-1\} \quad (5.3b)$$

$$0 \leq o_{ek} \leq 1-p_e \quad \forall e \in E, k \in \mathcal{K}(e) \quad (5.3c)$$

$$v_c = \prod_{e\in c}(1-p_e) \quad \forall c \in C \quad (5.3d)$$

$$o_{ek} \in \mathbb{R} \quad \forall e \in E, k \in \mathcal{K}(e) \quad (5.3e)$$

$$\{y, z\} \in \mathcal{M} \quad (5.3f)$$

where $\mathcal{M}$ denotes the set of feasible decision variables for the PICEF formulation of kidney exchange; for a description of this set please see Equation 4.1.

Constraints (5.3b), (5.3c), and (5.3d) define the discounted (expected) 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 expected weight of all cycles is expressed as $\sum_{c\in C} w_c z_c v_c$. Since all cycles are known ahead of time, both $w_c$ and $v_c$ can be treated as constant parameters rather than variables.

- For a *chain*, the expected weight is expressed using Lemma 5.1. Consider the following example: suppose a $k$-chain consists of edges $e_1,\ldots,e_k$. 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 (5.3b) we have $o_{e_1,1} \geq \frac{1}{1-p_{e_2}} o_{e_2,2}$ for vertex $i$. The sums in constraint (5.3b) contain no other terms, because $\mathcal{M}$ 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 $j = 2,\ldots,k-1$. Since Problem 5.3 is a maximization, 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 expected weight of all chains according to Lemma 5.1.

### 5.3.2 MIP Reformulation of Problem 5.3

Although Problem 5.3 exactly maximizes expected edge weight under non-identical edge failure probabilities, it is a nonconvex optimization problem since it involves several products of variables. For this reason, Problem 5.3 cannot be directly solved using integer linear programming. In this section, we reformulate this problem as a mixed-integer linear program which can be solved using general-purpose solvers. Proposition 5.1 gives our primary result; the main idea is to define a set of new variables $\theta_{ek}$ to replace $y_{ek} o_{ek}$ in Problem 5.3.

**Proposition 5.1.** *Problem 5.3 is equivalent to*

$$\begin{aligned} \max \quad & \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e \theta_{ek} + \sum_{c \in C} w_c z_c \left( \prod_{e \in c} (1 - p_e) \right) \\ s.t. \quad & \{y, z\} \in \mathcal{M}, \\ & \{y, \theta, o\} \in \mathcal{X}, \end{aligned} \tag{5.4}$$

*where $\mathcal{X}$ is defined as*

$$\mathcal{X} = \left\{ \begin{array}{l} \sum_{e \in \delta^-(i) \wedge k \in \mathcal{K}(e)} \theta_{ek} \geq \sum_{e \in \delta^+(i)} \frac{\theta_{e,k+1}}{1 - p_e} \quad \forall i \in P, k \in \{1, \ldots, L-1\} \\ \left. \begin{array}{l} \theta_{ek} \leq y_{ek} \\ \theta_{ek} \leq o_{ek} \\ 0 \leq o_{ek} \leq 1 - p_e \\ o_{ek}, \theta_{ek} \in \mathbb{R} \end{array} \right\} \quad \forall e \in E, k \in \mathcal{K}(e) \end{array} \right\} \tag{5.5}$$

TABLE 5.1: Comparison of stochastic and robust approaches to kidney exchange, which use a setting comparable to ours. Column"Opt." indicates the type of optimization approach used: Robust, Stochastic, or None. Column "Homog." indicates whether the approach assumes homogeneous edge failure probabilities (only for stochastic optimization approaches). The rightmost columns indicate the number of variables and constraints in each formulation.

| Formulation | Ope. | Homog. | # Vars. | # Constr. |
|---|---|---|---|---|
| PC-TSP [16] | None | N/A | $O(\|E\| \cdot \|V\| + \|V\|^2 + \|C\|)$ | $O(\|V\| \cdot (\|E\| + 2^{\|V\|} + \|C\|)$ |
| PICEF [109] | Stoch. | Yes | $O(L \cdot \|E\| + \|C\|)$ | $O(L \cdot \|V\| + L \cdot \|E\| + \|C\|)$ |
| ROBUST [216] | Robust | N/A | $O(\|E\| \cdot \|V\| + \|V\|^2 + \|C\|)$ | $O(\|E\| \cdot \|V\| + \|V\|^2 + \|C\|)$ |
| DPS-18 [110] | Stoch. | No | $O(\|V\|^L + \|C\|)$ | $O(\|V\|)$ |
| Our model (5.4) | Stoch. | No | $O(L \cdot \|E\| + \|C\|)$ | $O(L \cdot \|V\| + L \cdot \|E\| + \|C\|)$ |

**Scalability**

We compare our model size with state-of-the-art approaches in literature. We summarize all approaches in Table 5.1. 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 $|P| = O(|V|)$.

The size of our formulation in Proposition 5.1 is comparable with PICEF, while accounting for non-identical failure probabilities. DPS-18 [110] 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.

## 5.4 Edge Existence Uncertainty and CVaR

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 § 5.2.1). This is especially true in kidney exchange, where a single edge failure can impact an entire cycle or chain.

### 5.4.1 Conditional Value-at-Risk Model for Edge Existence Uncertainty

Here we follow a similar approach to that in Section 4.3, which proposed a CVaR model for the KEP with edge weight uncertainty; since the resulting CVaR formulation is intractable to solve exactly, we used a sample-average-approximation (SAA) to solve it. Here we follow a nearly-identical procedure to instead address edge *existence* uncertainty in the KEP.

At a high level, the CVaR objective for kidney exchange is expressed as $\mu + \gamma \times \mu_\alpha$, where $\mu$ is the expected matching weight and $\mu_\alpha$ 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 $\gamma$ is set by the user, and controls the trade-off between average performance and the *risk* of the solution. While the KEP is typically expressed as a maximization problem, here we formulate it as a minimization in keeping with the conventions of risk-minimization. For this purpose, we define the *negative* matching indicator of edge $e$ using auxiliary variables $\boldsymbol{m} \in \mathbb{R}^{|E|}$:

$$\boldsymbol{m}_e = -\sum_{k \in \mathcal{K}(e)} \boldsymbol{y}_{ek} - \sum_{c \in C} \mathbf{1}(e \in c)\boldsymbol{z}_c, \quad \forall e \in E.$$

That is, $\boldsymbol{m}_e = -1$ if edge $e$ is matched, and $\boldsymbol{m}_e = 0$ otherwise. We use $\boldsymbol{w} \in \mathbb{R}^{|E|}$ to represent the random discounted edge weights under known edge failure probabilities. Using these variables, $\boldsymbol{w}^\top \boldsymbol{m}$ represents the "loss" (negative weight) of a

matching. The $\alpha \times 100\%$ worst-case (highest) mean loss is equivalent to the CVaR objective [265] at level $\alpha$. The corresponding optimization problem is expressed in Problem 5.6, by introducing an auxiliary variable $d$.

$$\begin{aligned} \min \quad & \mathbb{E}\left[\boldsymbol{w}^\top \boldsymbol{m}\right] + \gamma\left(d + \frac{1}{\alpha}\mathbb{E}\left[\left(\boldsymbol{w}^\top \boldsymbol{m} - d\right)^+\right]\right) \\ s.t. \quad & \boldsymbol{m}_e = -\sum_{k \in \mathcal{K}(e)} \boldsymbol{y}_{ek} - \sum_{c \in C} \mathbf{1}(e \in c)\boldsymbol{z}_c \qquad \forall e \in E \\ & \boldsymbol{m} \in \mathbb{R}^{|E|} \\ & \{\boldsymbol{y}, \boldsymbol{z}\} \in \mathcal{M} \\ & d \in \mathbb{R}. \end{aligned} \tag{5.6}$$

As before, $(\cdot)^+$ denotes the positive part, and the expectation in (5.6) is taken over the known edge failure distribution. As before, $\mathcal{M}$ denotes the set of feasible matchings using the PICEF formulation.

### 5.4.2 An SAA-based Approach for Problem 5.6

The main difficulty in solving Problem 5.6 is that term $\mathbb{E}\left[\left(\boldsymbol{w}^\top \boldsymbol{m} - d\right)^+\right]$ does not have a simple closed-form reformulation. Instead, to approximate the objective of this problem we propose an approach based on Sample Average Approximation (SAA) [16] to solve (5.6); this is the same approach used in Section 4.3 for edge weight uncertainty. The main idea is to first sample $J$ "measurements" of edge existence according to the known edge failure probabilities; for each measurement we formulate a mixed-integer linear program representing the matching weight under this realization. Finally, we combine all $J$ models to obtain an optimization problem that approximates Problem 5.6 based on these $J$ measurements.

We express each measurement using variables $\hat{f}_e^j$, which is 1 if edge $e$ succeeds in measurement $j$ and 0 otherwise We assume that these measurements are drawn

independently from the *true* edge failure distribution.

These measurement variables are used as input to Problem 5.7, which uses decision variables $\hat{\boldsymbol{m}}^j$ to represent the edge discount factor for measurement $j$–that is, $\hat{\boldsymbol{m}}_e^j$ is 1 if edge $e$ is matched and succeeds in realization $j$ and 0 otherwise.

Using these decision variables, the objective of Problem 5.7 includes two terms: the mean matching weight, and the CVaR objective–both approximated using all $J$ samples (i.e., the sample-average approximation).

**Proposition 5.2.** *Problem 5.7 is equivalent to the SAA of Problem 5.6 under $J$ edge existence measurements represented by $\hat{\boldsymbol{f}}$, with*

$$
\begin{aligned}
\min \quad & \frac{1}{J}\sum_{j=1}^{J} \boldsymbol{w}^\top \hat{\boldsymbol{m}}^j + \gamma\left(d + \frac{1}{\alpha J}\sum_{j=1}^{J} \boldsymbol{r}_j\right) \\
s.t. \quad & \hat{\boldsymbol{m}}_e^j = -\sum_{k\in\mathcal{K}(e)} \boldsymbol{\theta}_{ekj} - \sum_{c\in C} \mathbf{1}(e\in c)\boldsymbol{z}_c \boldsymbol{v}_{cj} && \forall e\in E, j\in\{1,\dots,J\} \\
& \boldsymbol{r}_j \geq 0 && \forall j\in\{1,\dots,J\} \\
& \boldsymbol{r}_j \geq \boldsymbol{w}^\top \hat{\boldsymbol{m}}^j - d && \forall j\in\{1,\dots,J\} \\
& \boldsymbol{v}_{cj} = \min_{e\in c}\{\hat{\boldsymbol{f}}_e^j\} && \forall c\in C, j\in\{1,\dots,J\} \\
& \hat{\boldsymbol{m}}^j \in \mathbb{R}^{|E|} && \forall j\in\{1,\dots,J\} \\
& \boldsymbol{r}\in\mathbb{R}^J \\
& \{\boldsymbol{y},\boldsymbol{z}\}\in\mathcal{M} \\
& \{\boldsymbol{o},\boldsymbol{\theta}\}\in\mathcal{Q}
\end{aligned}
\tag{5.7}
$$

*where $\mathcal{M}$ is the set of PICEF constraints and $\mathcal{X}$ is defined as*

$$\mathcal{Q} = \left\{ \begin{array}{l} \sum_{e \in \delta^-(i) \wedge k \in \mathcal{K}(e)} \boldsymbol{\theta}_{ekj} \geq \sum_{e \in \delta^+(i)} \boldsymbol{\theta}_{e,k+1,j} \quad \forall i \in P, k \in \{1, \ldots, L-1\}, j \in \{1, \ldots, J\} \\ \left. \begin{array}{l} \boldsymbol{\theta}_{ekj} \leq \boldsymbol{y}_{ek} \\ \boldsymbol{\theta}_{ekj} \leq \boldsymbol{o}_{ekj} \\ \boldsymbol{o}_{ekj} \leq \hat{\boldsymbol{f}}_e^j \\ \boldsymbol{o}_{ekj}, \boldsymbol{\theta}_{ekj} \in [0,1] \end{array} \right\} \quad \forall e \in E, k \in \mathcal{K}(e), j \in \{1, \ldots, j\} \end{array} \right\}$$

*Proof.* We begin by verifying that auxiliary variables $\hat{\boldsymbol{m}}$ are defined correctly: where $\hat{\boldsymbol{m}}_e^j$ is $-1$ if $e$ is matched *and* succeeds in measurement $j$, and 0 otherwise. Note that a matched cycle edge succeeds only if none of the cycle edges fail; a matched chain edge succeeds only if it does not fail, *and* none of the previous matched edges in the chain fail. We also use the fact that an edge can be matched *only* in a single chain or a single cycle, due to PICEF constraints $\mathcal{M}$. First we show that constraints in Problem 5.7 define these auxiliary variables correctly, and we prove this by checking five (exhaustive) cases:

1. $e$ is not matched ($\hat{\boldsymbol{m}}_e^j = 0$ for all $j$): then $\boldsymbol{y}_{ek} = 0$ and $\boldsymbol{z}_c = 0$ for all $k \in \mathcal{K}(e)$, and for all $c \in C : e \in c$, as required by the PICEF constraints $\mathcal{M}$. In this case, due to constraints in $\mathcal{Q}$, $\boldsymbol{\theta}_{ej} = 0$ for all $j$, and thus $\hat{\boldsymbol{m}}_e^j = 0$ for all $j$.

2. $e$ is matched in a cycle $c$, and none of the cycle edges fail in measurement $j$ ($\hat{\boldsymbol{m}}_e^j = -1$): then $\boldsymbol{v}_{cj} = 1$, and $\boldsymbol{\theta}_{ekj} = 0$ since all $\boldsymbol{y}_{ek} = 0$ due to PICEF constraints. Thus, $\hat{\boldsymbol{m}}_e^j = -1$.

3. $e$ is matched in a cycle $c$, and the cycle fails in measurement $j$ ($\hat{\boldsymbol{m}}_e^j = 0$): in this case $\boldsymbol{v}_{cj} = 0$ and all $\boldsymbol{\theta}_{ekj} = 0$ since all $\boldsymbol{y}_{ek} = 0$ due to PICEF constraints. Thus, $\hat{\boldsymbol{m}}_e^j = 0$.

4. $e$ is matched at position $k$ in a chain, with prior edges $\{e_1, e_2, \ldots, e_{k-1}\}$, and succeeds in measurement $j$ ($\hat{\boldsymbol{m}}_e^j = -1$): in this case all $y_{e'k'} = 1$ and $\hat{\boldsymbol{f}}_{e'}^j = 1$ for all prior edges in the chain, up to edge $e$; this also means that the only constraints on $\boldsymbol{\theta}_{e'k'j}$ for all edges in this chain can be written as:

$$\sum_{e \in \delta^-(i) \wedge k \in \mathcal{K}(e)} \boldsymbol{\theta}_{ekj} \geq \sum_{e \in \delta^+(i)} \boldsymbol{\theta}_{e,k+1,j} \quad \forall i \in P, k \in \{1, \ldots, L-1\},$$

and since this is a minimization problem, and $\boldsymbol{w} > \boldsymbol{0}$, all $\boldsymbol{\theta}_{e'k'j}$ will be set to 1 in the optimal solution to Problem 5.7. Thus, $\hat{\boldsymbol{m}}_e^j = -1$.

5. $e$ is matched at position $k$ in a chain, with prior edges $\{e_1, e_2, \ldots, e_{k-1}\}$, and does not succeed in measurement $j$ ($\hat{\boldsymbol{m}}_e^j = 0$): let $e'$ be the failed edge, at position $k'$, in this chain (with the possibility that $e' = e$). Due to constraints in $\mathcal{Q}$, $\boldsymbol{\theta}_{e'k'j} = 0$; furthermore, due to the first constraints in $\mathcal{Q}$, for all subsequent edges in the chain $e''$ at position $k''$, $\boldsymbol{\theta}_{e''k''j} = 0$. Since all $\boldsymbol{\theta}$ variables for this chain are 0, and $e$ is matched in a chain $\hat{\boldsymbol{m}}_e^j = 0$.

Using these auxiliary variables, we can explicitly write the SAA of Problem 5.6 as follows:

$$\begin{aligned}
\min \quad & \frac{1}{J} \sum_{j=1}^{J} \boldsymbol{w}^\top \hat{\boldsymbol{m}}^j + \gamma \left( d + \frac{1}{\alpha J} \sum_{j=1}^{J} \left( \boldsymbol{w}^\top \hat{\boldsymbol{m}}^j - d \right)^+ \right) \\
s.t. \quad & \hat{\boldsymbol{m}}_e^j = - \sum_{k \in \mathcal{K}(e)} \boldsymbol{\theta}_{ekj} - \sum_{c \in C} \mathbf{1}(e \in c) \boldsymbol{z}_c \boldsymbol{v}_{cj} && \forall e \in E, j \in \{1, \ldots, J\} \\
& \hat{\boldsymbol{m}}^j \in \mathbb{R}^{|E|} && \forall j \in \{1, \ldots, J\} \\
& \{\boldsymbol{y}, \boldsymbol{z}\} \in \mathcal{M} \\
& \{\boldsymbol{o}, \boldsymbol{\theta}\} \in \mathcal{Q} \\
& d \in \mathbb{R}.
\end{aligned} \tag{5.8}$$

Next we replace the positive part using the auxiliary variables $\boldsymbol{r} \in \mathbb{R}^J$, with:

$$\boldsymbol{r}_j \equiv \left(\boldsymbol{w}^\top \hat{\boldsymbol{m}}^j - d\right)^+ = \min\{\boldsymbol{w}^\top \hat{\boldsymbol{m}}^j - d, 0\} \quad \forall j \in \{1, \ldots, J\}.$$

We define these variables using the constraints $\boldsymbol{r}_j \geq 0$ and $\boldsymbol{r}_j \geq \boldsymbol{w}^\top \hat{\boldsymbol{m}}^j - d$; since variables $\boldsymbol{r}_j$ appear with positive coefficients in the objective, one of these two constraints will be tight. Adding all constraints to the MIP formulation above, we recover Problem 5.7. □

There are both positive and negative aspects to the SAA formulation in Problem 5.7. First, this formulation can explicitly represent $J$ potential outcomes (edge failures/successes), and as we increase $J$ by taking more measurements, this formulation approaches the exact CVaR formulation. However the number of variables and constraints in this formulation grows polynomially in $J$. This is mainly a concern because Problem 5.7 is a MIP which, like the KEP, is NP-hard. Even though small instances of the KEP can be solved quickly using PICEF, performance can suffer as we modify the formulation. In the next section we demonstrate the benefits of this CVaR formulation, as well as the computational costs.

## 5.5 Experiments

First, we benchmark our tractable model for non-identical edge failure probabilities (5.4) ("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 [109] ("KEP"), and PICEF with identical edge failure probability [110] ("KEP-IP"). Second, we compare our CVaR model (5.7) ("CVAR") against KEP, KEP-IP, and KEP-NP; to our knowledge,

there are no other tractable approaches using the CVaR objective in our setting. We then briefly present the running time of all implemented approaches.

### 5.5.1 Stochastic Objective

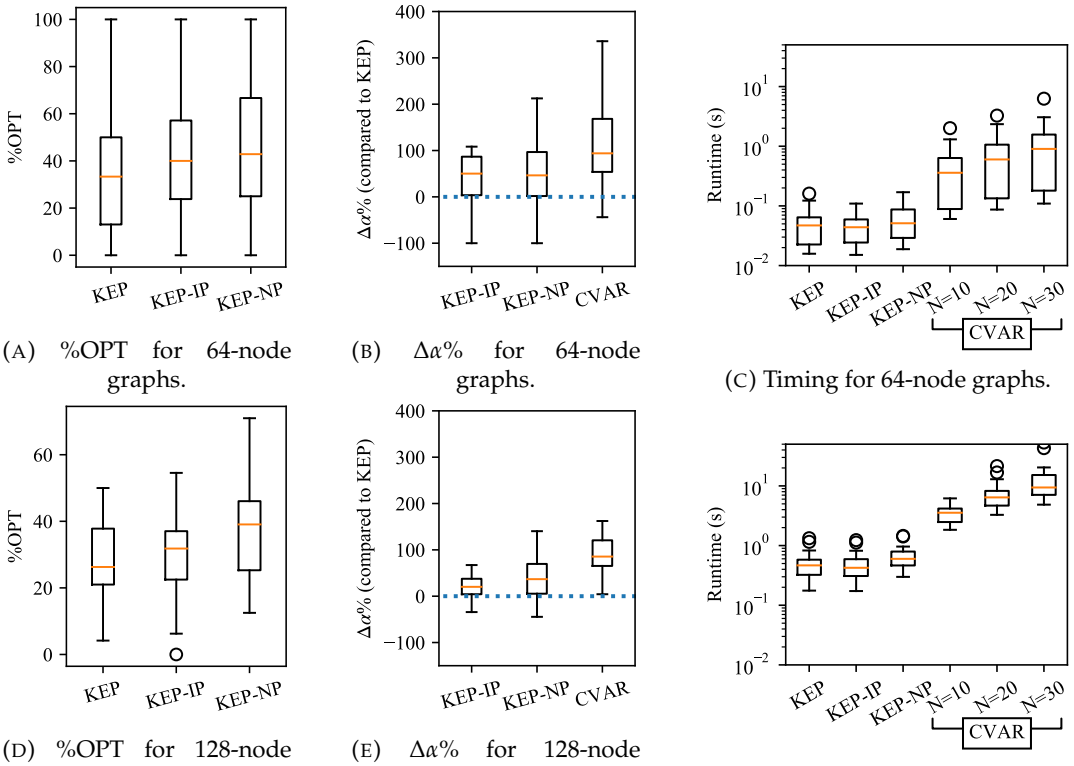
We use two sets of 32 randomly-generated graphs, one with 64 nodes each and one with 128 nodes each. These graphs resemble the structure of real exchanges, and are generated using anonymized data from the United Network for Organ Sharing (UNOS), a US-based kidney exchange. We simulate edge existence 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 $\equiv 100 \times W_M / W_{OPT}$. Figure 5.2 (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.

### 5.5.2 CVaR Objective

We implement CVAR (§ 5.5.1) using $N = 10$ simulated edge realizations, with $\gamma = 10$, and $\alpha = 0.5$.




(A) %OPT for 64-node graphs.

(B) $\Delta\alpha$% for 64-node graphs.

(C) Timing for 64-node graphs.

(D) %OPT for 128-node graphs.

(E) $\Delta\alpha$% for 128-node graphs.

(F) Timing for 128-node graphs.

FIGURE 5.2: 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.

**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 $\mu_P^\alpha$ be the $\alpha$% lowest-realized matching weights for KEP, and let $\mu_M^\alpha$ 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 5.2 (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 5.2 (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 ($N$).

## 5.6 Discussion

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 [198]); 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 protects 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 [110]—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 [134, 229].

Finally, each of the approaches discussed in this chapter 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 [215, 269, 324].

## 5.7 Authors and Publication

This chapter was written by Duncan McElfresh, Ke Ren, Hoda Bidkhori, and John P Dickerson. It appeared at the 2020 Conference on Uncertainty in Artificial Intelligence (UAI-21). [46]

# Chapter 6: Fairness in Kidney Exchange

## 6.1 Introduction

In this chapter we turn to the issue of *fairness* for hard-to-match patients in kidney exchanges. Certain patients are particularly disadvantaged because, due to health characteristics and/or logistical factors, they are very unlikely to find a compatible donor. Intuitively, any enforcement of a fairness constraint or consideration may impact the overall economic efficiency (as measured by the total number or quality of goods exchanged). A quantification of this trade-off is known as the *price of fairness* [45]. Recent work by Dickerson et al. [108] adapted this concept to the kidney exchange case, and presented two fair allocation methods that strike a balance between fairness and efficiency. Yet, as we show in this chapter, those methods can "fail" unpredictably, yielding an arbitrarily high price of fairness.

With this as motivation, we adapt to the kidney exchange case a recent technique for trading off a form of fairness and utilitarianism in a principled manner. This technique is parameterized by a bound on the price of fairness, as opposed to a set of parameters that may result in hard-to-predict final matching behavior, as in past work. We implement our fair algorithm in a realistic mathematical programming framework and–on real data from a large, multi-center, fielded kidney exchange–show that our algorithm effectively balances fairness and efficiency without unwanted outlier behavior.

### 6.1.1 Related Work

We briefly overview related work in balancing efficiency and fairness in resource allocation problems. Bertsimas et al. [45] define the *price of fairness*; that is, the relative loss in system efficiency under a fair allocation algorithm. Hooker and Williams [168] give a formal method for combining utilitarianism and equity. We direct the reader to those two papers for a greater overview of research in fairness in general resource allocation problems.

Fairness in the context of kidney exchange was first studied by Roth et al. [269]; they explore concepts like Lorenz dominance in a stylized model, and show that preferring fair allocations can come at great cost. Li et al. [200] extend this model and present an algorithm to solve for a Lorenz dominant matching. Stability in kidney exchange, a concept intimately related to fairness, was explored by Liu et al. [204]. The use of randomized allocation mechanisms to promote fairness in stylized models is theoretically promising [33, 125, 212]. Recent work discusses fairness in stylized random graph models of dynamic kidney exchange [15, 26]. None of these papers provide practical models that could be implemented in a fully-realistic and fielded kidney exchange.

Practically speaking, Yılmaz [324] explores in simulation equity issues from combining living and deceased donor allocation; that paper is limited to only short length-two kidney swaps, while real exchanges all use longer cycles and chains. Dickerson et al. [108] introduced two fair allocation algorithms explicitly in the context of kidney exchange, and proved bounds on the price of fairness under those methods in a random graph model; we build on that work in this chapter, and describe it in greater detail later. That work has been incorporated into a framework for learning to balance efficiency, fairness, and dynamism in matching markets [104];

we note that the fair allocation algorithm we present in this chapter could be used in that framework as well.

**Contributions**

- Dickerson et al. [108] finds that the theoretical price of fairness in kidney exchange is small when *only* patient-donor pairs participate in the exchange. They did not include non-directed donors (NDDs). However, in modern kidney exchanges, non-directed donors (NDDs) provide many more matches than patient-donor pairs; furthermore, NDDs create more opportunities to expand the fair matching, potentially increasing the price of fairness. Here, we prove that adding NDDs to the theoretical model actually *decreases* the price of fairness, and that—with enough NDDs—the price of fairness is zero.

- Real kidney exchanges are less dense and more uncertain than the (standard) theoretical model in which we prove our results. Previous approaches to incorporating fairness into kidney exchange have neglected this fact: they have been either ad-hoc—e.g., "priority points" decided on by committee [180]—or brittle [108, 269], resulting in an unacceptably high price of fairness. We provide the first approach to incorporating fairness into kidney exchange in a way that both prioritizes disadvantaged participants, but also comes with acceptable worst-case guarantees on the price of fairness. Our method is easily applied as an objective in the mathematical-programming-based clearing methods used in today's fielded exchanges; indeed, using real data we show that this method guarantees a limit on efficiency loss.

### 6.1.2 The Price of Fairness

As an example for this chapter, we focus on *highly-sensitized* patients, who have a very low probability of passing compatibility test with a random donor organ. Sensitization is determined using the Calculated Panel Reactive Antibody (CPRA) level of each patient, a number between 0 and 100, which reflects the likelihood that a patient will find a matching donor. The CPRA level indicates roughly how many donors are *incompatible* with the patient (higher CPRA means fewer compatible donors), and patients with CPRA above 80 are generally considered highly-sensitized. It can be very difficult for highly-sensitized patients to find a compatible donor, and their waiting times for a transplant can be much longer than for less-sensitized patients.[1] Utilitarian objectives will, in general, marginalize these patients. Sensitization is determined using the Calculated Panel Reactive Antibody (CPRA) level of each patient, which reflects the likelihood that a patient will find a matching donor.

Formally, we denote the sensitization of each patient-donor vertex $v$ as $v_s \in [0, 100]$, the CPRA level of $v$'s patient; NDD vertices are not associated with patients, so they do not have sensitization levels. Each patient-donor vertex $v \in P$ is considered highly sensitized if $v_s$ exceeds threshold $\tau \in [0, 100]$, and lowly-sensitized otherwise. These vertex sets $V_H$ and $V_L$ are defined as:

- Lowly sensitized: $V_L = \{v \mid v \in P : v_s < \tau\}$
- Highly sensitized: $V_H = \{v \mid v \in P : v_s \geq \tau\}$.

By definition, highly-sensitized patients are harder to match than lowly-sensitized patients. Naturally, efficient matching algorithms prioritize easy-to-match vertices in $V_L$, marginalizing $V_H$. Let $u_f : \mathcal{M} \to \mathbb{R}$ be a *fair* utility function. Formally, a utility function is fair when its corresponding optimal match $M_f^*$ is viewed as fair, where

[1] https://optn.transplant.hrsa.gov/data/

$M_f^*$ is defined as:

$$M_f^* = \arg\max_{M \in \mathcal{M}} u_f(M)$$

Bertsimas et al. [45] defined the *price of fairness* to be the "relative system efficiency loss under a fair allocation assuming that a fully efficient allocation is one that maximizes the sum of [participant] utilities." Caragiannis et al. [72] defined an essentially identical concept in parallel. Formally, given a fair utility function $u_f$ and the utilitarian utility function $u$, the price of fairness is:

$$\mathsf{POF}(\mathcal{M}, u_f) = \frac{u\left(M^*\right) - u\left(M_f^*\right)}{u\left(M^*\right)} \tag{6.1}$$

The price of fairness $\mathsf{POF}(\mathcal{M}, u_f)$ is the relative loss in (utilitarian) efficiency caused by choosing a fair outcome $M_f^*$ (selected by a fair utility function $u_f$), rather than the most efficient outcome. In the next section we show that the theoretical price of fairness in kidney exchange is small, even when both cycles *and chains* are used—thus generalizing an earlier result due to Dickerson et al. [108] to modern kidney exchanges.

## 6.2 The Theoretical Price of Fairness with Chains is Low (or Zero)

In this section we use the random graph model for kidney exchange introduced by Ashlagi and Roth [24] to show that the theoretical price of fairness is always small, especially when NDDs are included. A complete description of this model can be found in Appendix B.1. Dickerson et al. [108] finds that without NDDs, the

maximum price of fairness is 2/33. Adding NDDs to this model creates more opportunities to match highly sensitized patients, which could potentially lead to a higher price of fairness. However we find that including chains in this model only *decreases* the price of fairness; furthermore, when the ratio of NDDs to patient-donor pairs is high enough, the price of fairness is zero.

### 6.2.1 Price of Fairness

Ashlagi and Roth [24] characterize efficient matchings in a random graph model without chains, and Dickerson et al. [108] build on this to show that the price of fairness without chains is bounded above by 2/33. Dickerson et al. [106] extend the efficient matching of Ashlagi and Roth [24] to include chains, but do not calculate the price of fairness. We close the gap in theory regarding the price of fairness with chains.

Given $|P|$ patient-donor pairs, we parameterize the number of NDDs $|N|$ with $\beta \geq 0$ such that $|N| = \beta|P|$. Theorems 6.1 and 6.2 state our two main results: adding chains to the random graph model does not increase the price of fairness, and when the fraction of NDDs is high enough ($\beta > 1/8$), the price of fairness is zero. The proofs of the following theorems are given in Appendix B.1.

**Theorem 6.1.** *Adding NDDs to the random graph model ($\beta > 0$) does not increase the upper bound on the price of fairness found by Dickerson et al. [108].*

**Proof Sketch:** We explore every possible efficient matching on the random graph model with chains; only four of these matchings have nonzero price of fairness. For each case, we compare the price of fairness to that of the efficient matching without chains found in Dickerson et al. [108], and find that the upper bound does not increase.

**Theorem 6.2.** *The price of fairness is zero when* $\beta > 1/8$.

**Proof sketch:** For each matching with nonzero price of fairness, $\beta \leq 1/8$. When $\beta > 1/8$, a different matching occurs, and the price of fairness is zero.

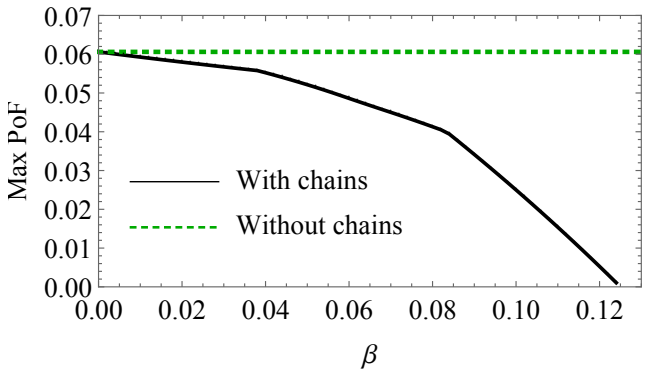



FIGURE 6.1: Price of fairness with chains. (The horizontal dotted line at 2/33 is the price of fairness without chains.)

To illustrate these results, we compute the price of fairness when $\beta \in [0, 1/8]$. These calculations confirm our theoretical results, as shown in Figure 6.1: the price of fairness decreases as $\beta$ increases, and is zero when $\beta > 1/8$.

The worst-case price of fairness is small in the random graph model, with or without NDDs. However, real exchange graphs are typically much sparser and less uniform—in reality the price of fairness can be high. In the next section, we discuss two notions of fairness in kidney exchange and determine their worst-case price of fairness.

## 6.3 The Price of Fairness in State-of-the-Art Fair Algorithm can be Arbitrarily Bad

The price of fairness depends on how fairness is defined. This is especially true in real exchanges where the price of fairness can be unacceptably high. In this section, we discuss two kidney-exchange-specific fair algorithms introduced by Dickerson et al. [108]: lexicographic fairness and weighted fairness. These methods favor the

disadvantaged class without considering overall loss in efficiency; we show that in the worst case these methods allow the the price of fairness to approach 1 (i.e., total efficiency loss).

### 6.3.1 Lexicographic Fairness

As proposed by Dickerson et al. [108], $\alpha$-lexicographic fairness assigns nonzero utility only to matchings that award at least a fraction $\alpha$ of the maximum possible fair utility. Letting $u_H(M)$ and $u_L(M)$ be the utility assigned to only vertices in $V_H$ and $V_L$, respectively, the utility function for $\alpha$-lexicographic fairness is given in Equation 6.2.

$$u_\alpha(M) = \begin{cases} u_L(M) + u_H(M) & \text{if } u_H(M) \geq \alpha \max\limits_{M' \in \mathcal{M}} u_H(M') \\ 0 & \text{otherwise.} \end{cases} \tag{6.2}$$

Theorems 6.3 and 6.4 state that strict lexicographic fairness ($\alpha = 1$) allows the price of fairness to approach 1.

**Theorem 6.3.** *For any cycle cap $K$ there exists a graph $G$ such that the price of fairness of $G$ under $\alpha$-lexicographic fairness with $0 < \alpha \leq 1$ is bounded by $\mathsf{POF}(\mathcal{M}, u_\alpha) \geq \frac{K-2}{K}$.*

*Proof.* Consider a kidney exchange graph with one highly-sensitized patient $H$ and $k$ non-highly-sensitized patients $V_1, \ldots, V_k$, where all $V_i$ vertices form a directed cycle of length $k$. A 2-cycle connects $H$ with one $V_i$; see Figure 6.2 (right) for an example with $k = 4$. With a cycle cap of $k$, the optimal utilitarian matching has utility $k$, while the optimal lexicographic matching has utility $u_\alpha = 2$, for any $0 < \alpha \leq 1$. The price of fairness in this graph is $\mathsf{POF}(\mathcal{M}, u_\alpha) = (k-2)/k$. □

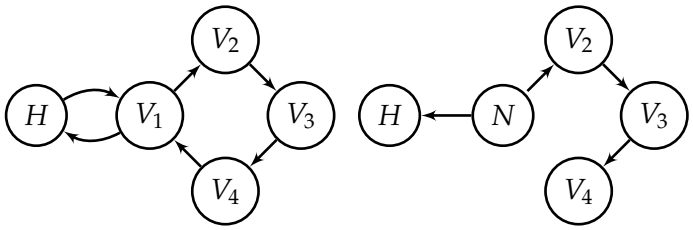



FIGURE 6.2: Supporting graphs for Theorems 6.3 (left) and 6.4 (right), with cycle cap 4 and chain cap 3, respectively.

**Theorem 6.4.** *For any chain cap $k$ there exists a graph $G$ such that the price of fairness of $G$ under the $\alpha$-lexicographic fair algorithm with $0 < \alpha \leq 1$ is bounded by $\mathsf{POF}(\mathcal{M}, u_\alpha) \geq \frac{k-1}{k}$.*

*Proof.* Consider an example graph with one highly sensitized vertex $H$, one NDD, and $k$ non-sensitized vertices $V_1, \ldots, V_k$. The NDD can initiate one of two chains: a 1-chain to $H$, and a $k$-chain including all $k$ non-sensitized vertices; see Figure 6.2 (left) for an example with $k = 4$. With a chain cap of $k$, the optimal utilitarian matching has utility $k$, while the optimal $\alpha$-lexicographic matching has utility $u_\alpha = 1$ for any $0 < \alpha \leq 1$. The price of fairness in this graph is $\mathsf{POF}(\mathcal{M}, u_\alpha) = (k-1)/k$. □

Thus, $\alpha$-lexicographic fairness allows for a price of fairness that approaches 1 as the cycle and chain cap increase.

### 6.3.2 Weighted Fairness

The weighted fairness algorithm [108] defines a utility function by first modifying the original edge weights $w_e$ by a multiplicative factor $\gamma \in \mathbb{R}$ such that

$$w'_e = \begin{cases} (1+\gamma)w_e & \text{if } e \text{ ends in } V_H \\ w_e & \text{otherwise.} \end{cases}$$

Then the weighted fairness algorithm $u_{WF}$ is

$$u_{WF}(M) = \sum_{c \in M} u'(c),$$

where $u'(c)$ is the utility of a chain or cycle $c$ with modified edge weights. The modified edge weights prompt the matching algorithm to include more highly-sensitized patients; as in the lexicographic case, we now show that the price of fairness approaches 1 under weighted fairness.

**Theorem 6.5.** *For any cycle cap $k$ and $\gamma > k-1$, there exists a graph $G$ such that the price of fairness of $G$ under the weighted fairness algorithm is bounded by $\mathit{POF}(\mathcal{M}, u_{WF}) \geq \frac{k-2}{k}$.*

*Proof.* Consider the graph used in the proof of Theorem 6.3, with all edge weights equal to 1. Weighted fairness increases the weight of the edge ending in $H$ to $(1+\gamma)$. The weighted utility of the 2-cycle is $2+\gamma$, while the weighted utility of the $L$-cycle is $L$. If $\gamma$ is chosen such that $\gamma > L-2$, then the 2-cycle will be chosen over the $L$-cycle, resulting in the price of fairness $\mathsf{POF}(\mathcal{M}, u_{WF}) = (k-2)/k$. □

**Theorem 6.6.** *For any chain cap $k$ and $\gamma > k-1$, there exists a graph $G$ such that the price of fairness of $G$ under the weighted fairness algorithm is bounded by $\mathit{POF}(\mathcal{M}, u_{WF}) \geq \frac{k-1}{k}$.*

*Proof.* Consider the graph used in the proof of Theorem 6.4, with all weights equal to 1. The weighted utility of the 1-chain is $1+\gamma$, while the weight of the $k$-chain is $R$. If $\gamma$ is chosen such that $\gamma > R-1$, then the 1-chain will be chosen over the $k$-chain, resulting in the price of fairness $\mathsf{POF}(\mathcal{M}, u_{WF}) = (k-1)/k$. □

In the worst case, weighted fairness allows a price of fairness that approaches 1 as the cycle and chain caps increase. The price of fairness also approaches 1 as $\gamma$ increases.

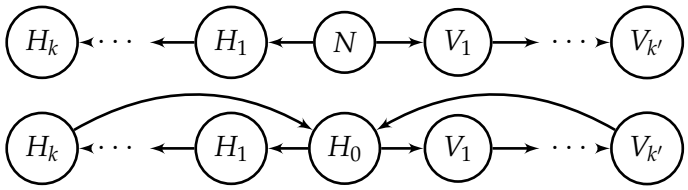



FIGURE 6.3: Graphs for Theorems 6.7 (top) and 6.8 (bottom).

**Theorem 6.7.** *With no chain cap, there exists a graph $G$ such that the price of fairness of $G$ under the weighted fairness algorithm is bounded by $\mathsf{POF}(\mathcal{M}, u_{WF}) \geq \frac{\gamma}{\gamma+1}$.*

*Proof.* Consider a graph with a single NDD connected to a chain with highly-sensitized patients $H_i$ of length $k$, and a chain with non-highly sensitized patients $V_i$ of length $k' \equiv \lfloor(\gamma+1)k\rfloor - 1$; this graph is shown in Figure 6.3 (top). Under weighted fairness, the chain with $k'$ non-sensitized vertices receives utility $u_L = \lfloor(\gamma+1)k\rfloor - 1$ while the chain with $k$ highly-sensitized vertices receives utility $u_H = (\gamma+1)k$, so $u_H > u_L$. The price of fairness for this graph is

$$\mathsf{POF}(\mathcal{M}, u_{WF}) = \frac{\lfloor(\gamma+1)k\rfloor - 1 - k}{\lfloor(\gamma+1)k\rfloor - 1} = \frac{\lfloor\gamma k\rfloor - 1}{\lfloor(\gamma+1)k\rfloor - 1} \geq \frac{\gamma k - 2}{(\gamma+1)k - 1}.$$

Taking the limit as $k \to \infty$, we have

$$\lim_{k\to\infty} \frac{\gamma k - 2}{(\gamma+1)k - 1} = \frac{\gamma}{\gamma+1},$$

which implies $\mathsf{POF}(\mathcal{M}, u_{WF}) \geq \frac{\gamma}{\gamma+1}$. □

A similar result exists with cycles rather than chains.

**Theorem 6.8.** *With no cycle cap there exists a graph $G$ such that the price of fairness of $G$ under the weighted fairness algorithm is bounded by $\mathsf{POF}(\mathcal{M}, u_{WF}) \geq \frac{\gamma}{\gamma+1}$.*

*Proof.* Consider the graph used in the proof of Theorem 6.7, where the NDD $N$ is instead a highly-sensitized pair $H_0$, and the end vertices of both chains both have

edges ending in $H_0$; this graph is shown in Figure 6.3 (bottom). Under weighted fairness, the cycle with non-sensitized vertices receives utility $u_L = \lfloor(\gamma+1)k\rfloor$, while the cycle with sensitized vertices receives utility $u_H = (\gamma+1)k+1$, so $u_H > u_L$. The price of fairness for this graph is

$$\mathsf{POF}(\mathcal{M}, u_{WF}) = \frac{\lfloor(\gamma+1)k\rfloor - k - 1}{\lfloor(\gamma+1)k\rfloor} \geq \frac{\gamma k - 2}{(\gamma+1)k}.$$

Taking the limit as $k \to \infty$, we have

$$\lim_{k\to\infty} \frac{\gamma k - 2}{(\gamma+1)k} = \frac{\gamma}{\gamma+1},$$

□

These bounds show that weighted fairness allows for a price of fairness that approaches 1, i.e., arbitrarily bad, as the cycle cap, chain cap, or $\gamma$ increase.

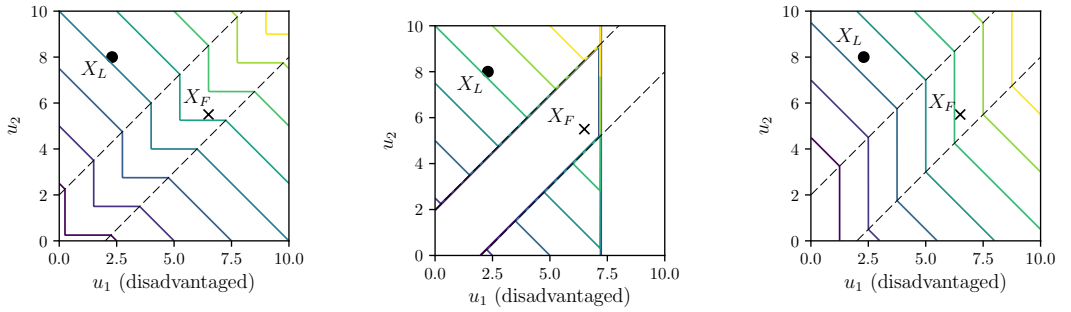
We have shown that the worst-case prices of fairness approach 1 under both the lexicographic and weighted fairness algorithms of Dickerson et al. [108]. Next, we propose an algorithm that favors disadvantaged groups, but also strictly *limits* the price of fairness using a parameter set by policymakers.

## 6.4 Hybrid Fairness Algorithm

In this section, we present a hybrid fair utility function that balances lexicographic fairness and a utilitarian objective. We generalize the hybrid utility function proposed by Hooker and Williams [168], which chooses between a Rawlsian (or maximin) objective and a utilitarian objective for multiple classes of agents.

### 6.4.1 Utilitarian and Rawlsian Fairness

Consider two classes of agents that receive utilities $u_1(X)$ and $u_2(X)$, respectively, for outcome $X$. The fair algorithms introduced by Hooker and Williams [168] maximizes the utility of the worst-off class, unless this requires taking too many resources from other classes. When the inequality exceeds a threshold $\Delta$ (i.e., $|u_1(X) - u_2(X)| > \Delta$) they switch to a utilitarian objective that maximizes $u_1(X) + u_2(X)$. The utility function for this method is




(A) Hybrid Rawlsian-Utilitarian

(B) $\Delta$-Lexicographic ($u_{\Delta 1}$)

(C) Relaxed $\Delta$-Lexicographic ($u_\Delta$)

FIGURE 6.4: Level sets for hybrid fair utility functions with $\Delta = 2$, with example outcomes $X_L$ and $X_F$.

$$u_\Delta(X) = \begin{cases} 2\min(u_1(X), u_2(X)) + \Delta & \text{if } |u_1(X) - u_2(X)| \leq \Delta \\ u_1(X) + u_2(X) & \text{otherwise.} \end{cases}$$

The parameter $\Delta$ is problem-specific, and should be chosen by policymakers. Figure 6.4a shows the level sets of this utility function, with $\Delta = 2$. This utility function can be generalized by switching to a different method in the *fair region* (i.e., when $|u_1(X) - u_2(X)| \leq \Delta$). The next section generalizes this method using lexicographic fairness.

### 6.4.2 Hybrid-Lexicographic Algorithm

When it is desirable to favor one class of agents $g_1$ over class $g_2$, lexicographic fairness favors $g_1$. We propose an algorithm that implements lexicographic fairness only when inequality between groups does not exceed $\Delta$. This algorithm uses two steps: (1) determine whether inequality is small enough to use lexicographic fairness (2) choose the optimal outcome. These steps are outlined below, and formalized in Algorithm 1.

**Step 1:** Find all outcomes that maximize a hybrid utility function, and determine whether lexicographic fairness is appropriate.

We use a utility function to identify outcomes that satisfy either a lexicographic or utilitarian objective. Equation 6.3 shows one option for such a utility function, which assigns strict lexicographic utility ($\alpha = 1$) according to Equation 6.2 in the fair region, and utilitarian utility otherwise.

$$u_{\Delta 1}(X) = \begin{cases} u_1(X) + u_2(X) & \text{if } |u_1(X) - u_2(X)| \leq \Delta \text{ and } u_1(X) = \max\limits_{X' \in \mathcal{X}}(u_1(X')) \\ u_1(X) + u_2(X) & \text{if } |u_1(X) - u_2(X)| > \Delta \\ 0 & \text{otherwise.} \end{cases} \tag{6.3}$$

where $\mathcal{X}$ is the set of all possible outcomes. Figure 6.4b shows the contours $u_{\Delta 1}$. This utility function is clearly too harsh—it assigns zero utility to outcomes in the fair region that do not maximize $u_1$, and its optimal outcomes are not always Pareto efficient. Consider outcomes $X_F$ and $X_L$ in Figure 6.4b. $X_F$ is in the fair region but does not maximize $u_1$, so $u_{\Delta 1}(X_F) = 0$; $X_L$ is in the utilitarian region but is less efficient, so $u_{\Delta 1}(X_L) = u(X_L)$. Under utility function $u_{\Delta 1}$, the less-efficient outcome $X_L$ is chosen over $X_F$.

To address this problem we introduce $u_\Delta$ in Equation 6.4, which relaxes $u_{\Delta 1}$. For outcomes in the fair region (that is, with $|u_1 - u_2| \leq \Delta$), utility is assigned proportional to $u_1$. As shown in Figure 6.4c, the contours of $u_\Delta$ are continuous.

$$u_\Delta(X) = \begin{cases} u_1(X) + u_2(X) - \Delta & \text{if } u_2(X) - u_1(X) > \Delta \\ 2u_1(X) & \text{if } |u_1(X) - u_2(X)| \leq \Delta \\ u_1(X) + u_2(X) + \Delta & \text{if } u_1(X) - u_2(X) > \Delta \end{cases} \tag{6.4}$$

Let $X_{OPT}$ be the set of "optimal" outcomes, which maximize $u_\Delta$. If $X_{OPT}$ contains multiple outcomes, it is reasonable to randomly sample an optimal outcome, unless there are additional ways to evaluate outcomes. If the randomly-sampled outcome is in the utilitarian region, then we are done. However, if any outcomes in $X_{OPT}$ are in the fair region, then Step 2 must be used. This process is described below, and formalized in Algorithm 1.

**Step 2:** If any solution in $X_{OPT}$ is in the fair region, select the lexicographic-optimal solution in the fair region. That is, select a solution in the fair region that maximizes $u_2$. Since the utility function $u_\Delta$ assigns the same utility to all solutions in the fair region with the same $u_1(X)$, no matter the value of $u_2(X)$. However, if there exist two outcomes $X_A$ and $X_B$ such that $u_1(X_A) = u_1(X_B)$ and $u_2(X_A) > u_2(X_B)$, then $X_A$ is lexicographically preferred to $X_B$.

**Algorithm 1** LexFair($\Delta, \mathcal{M}$)

**Require:** Threshold $\Delta$, outcomes $\mathcal{M}$

$\mathcal{M}_{OPT} \leftarrow \arg\max_{M \in \mathcal{M}} u_\Delta(M)$

**if** $|\mathcal{M}_{OPT}| > 1$ **then**

Select an outcome $M$ uniformly at random from $\mathcal{M}_{OPT}$

**if** $M$ is in the utilitarian region **then**

$M^* \leftarrow M$

**else**

$\mathcal{M}_1 \leftarrow \{M' \in \mathcal{M}_{OPT} \mid u_1(M') = u_1(M)\}$

$M^* \leftarrow \arg\max_{M' \in \mathcal{M}_1} u_2(M')$

**else**

$M^* \leftarrow \mathcal{M}_{OPT}$

**return** $\mathcal{M}_{OPT}$

### 6.4.3 Hybrid Algorithm for Several Classes

We now generalize the hybrid-lexicographic fair algorithm to more than two classes. Consider a set $\mathcal{P}$ of classes $g_i$, $i = 1, \ldots, |\mathcal{P}|$. Let there be an ordering $\succ$ over $g_i$, where $g_a \succ g_b$ indicates that $g_a$ should receive higher priority over $g_b$. Without loss of generality we assume the preference ordering over groups to be $1 \succ 2 \succ \cdots \succ |\mathcal{P}|$. Let $u_i(X)$ be the utility received by group $i$ under outcome $X$. As in the previous section, we (1) use a utility function to identify candidate outcomes, and (2) select a candidate outcome that is lexicographically optimal, if necessary.

**Step 1:** To define a utility function, we observe that in Equation 6.4, in the utilitarian region a positive offset $\Delta$ is added if $u_1(X) - u_2(X) > \Delta$, and a negative offset is added if $u_2(X) - u_1(X) > \Delta$. With $|\mathcal{P}|$ classes, each outcome receives a utility offset of $+\Delta$ if $u_1(X) - u_i(X) > \Delta$, and a negative offset of $-\Delta$ if $u_i(X) - u_1(X) > \Delta$,

for each class $i = 2, 3, \ldots, |\mathcal{P}|$. As in the previous section, these offsets help ensure the outcome is Pareto efficient, and has a bounded price of fairness; we formalize both of these claims after presenting our hybrid method. The utility function used in Step 1 is

$$u_\Delta(X) = u_1(X) + \sum_{i=2}^{|\mathcal{P}|} u_{\Delta i}(X) \tag{6.5}$$

with

$$\begin{aligned} u_{\Delta i}(X) \equiv \Big( & u_1(X) \cdot \mathbf{1}\{|u_i(X) - u_1(X)| \leq \Delta\} \\ & + u_i(X) \cdot \mathbf{1}\{|u_i(X) - u_1(X)| > \Delta\} \\ & + \Delta \cdot \mathbf{1}\{u_1(X) - u_i(X) > \Delta\} \\ & - \Delta \cdot \mathbf{1}\{u_i(X) - u_1(X) > \Delta\} \Big) \end{aligned}$$

where $\mathbf{1}\{\cdot\}$ is the indicator function. The first term in $u_{\Delta i}(\cdot)$ adds $u_1(X)$ if the difference between $u_1(X)$ and $u_i(X)$ does not exceed $\Delta$, and the second term adds $u_i(X)$ otherwise. The second two terms add offsets of $+\Delta$ or $-\Delta$ if the absolute difference between $u_1(X)$ and $u_i(X)$ exceeds the threshold $\Delta$. Note that Equation 6.5 is equivalent to Equation 6.4 when $|\mathcal{P}| = 2$.

**Step 2:** As in the 2-class case, we use an algorithm to ensure the final outcome is Pareto efficient, using the following procedure: If multiple outcomes maximize $u_\Delta(\cdot)$, then one is selected at random. If there is no $1 < i \leq |\mathcal{P}|$ such that $|u_1(X) - u_i(X)| \leq \Delta$, then we are done. Otherwise, to ensure Pareto efficiency we must select a lexicographic-optimal outcome subject to the preference ordering $1 \succ 2 \succ \cdots \succ |\mathcal{P}|$. This process is described in Algorithm 2.

**Algorithm 2** LexFair($\Delta, \mathcal{M}$) for $|\mathcal{P}| \geq 2$ classes

**Require:** Threshold $\Delta$, outcomes $\mathcal{M}$

$\mathcal{M}_{OPT} \leftarrow \arg\max_{M \in \mathcal{M}} u_\Delta(M)$

**if** $|\mathcal{M}_{OPT}| = 1$ **then return** $\mathcal{M}_{OPT}$

**else**

Select an outcome $M$ uniformly at random from $\mathcal{M}_{OPT}$

$J \leftarrow \{i > 1 \mid |u_i(M) - u_1(M)| > \Delta\} \cup \{1\}$

$\mathcal{M}_1 \leftarrow \{X \in \mathcal{M}_{OPT} \mid u_j(X) \geq u_j(M) \forall j \in J\}$

**for** $i = 2, \ldots, |\mathcal{P}|$ **do**

**if** $i \in J$ **then**

$\mathcal{M}_i \leftarrow \mathcal{M}_{i-1}$ (don't update the set of candidate outcomes)

**else**

$z \leftarrow \max\{u_i(X) \mid X \in \mathcal{M}_{i-1}\}$

$\mathcal{M}_i \leftarrow \{X \in \mathcal{M}_{i-1} \mid u_i(X) = z\}$

**return** any outcome in $\mathcal{M}_{|\mathcal{P}|}$

Algorithm 2 starts by identifying the set of outcomes $\mathcal{M}_{OPT}$ that maximize $u_\Delta(\cdot)$. If $\mathcal{M}_{OPT}$ contains multiple outcomes, then a lexicographic maximization procedure is used: first we select an outcome $M$ at random. Then we identify all groups whose utility in $M$ is at least $\Delta$ greater or less than $u_i(M)$; these groups are added to set $J$, along with the most-prioritized group (1); note that $J$ includes all groups whose utilities are represented in $u_\Delta(\cdot)$ for outcome $M$. We then identify all outcomes that are weakly preferred to $M$ by all groups in $J$; these outcomes are denoted by $\mathcal{M}_1$. Next we find a lexicographically-optimal solution, by maximizing $u_i(\cdot)$ for all groups *not* in $J$.

### 6.4.4 Properties of LexFair($\Delta, \mathcal{M}$)

Here we prove two important properties of LexFair($\Delta, \mathcal{M}$): a bounded price of fairness, and Pareto efficiency.

The following theorem gives a bound on the price of fairness for the hybrid-lexicographic algorithm (LexFair($\Delta, \mathcal{M}$)) defined in Algorithm 2.

**Theorem 6.9.** *The price of fairness of LexFair($\Delta, \mathcal{M}$) is at most $2\Delta(|\mathcal{P}| - 1)/u_E$, where $u_E$ is the sum of group utilities in the utilitarian-optimal outcome in $\mathcal{M}$.*

*Proof.* Suppose there are two outcomes: a "fair" outcome $X_F$, which is returned by LexFair($\Delta, \mathcal{M}$), and an "efficient" outcome $X_E$, which maximizes the sum of group utilities. First we define two partitions of the groups $\mathcal{P}$, as follows:

$$Z^{\Delta}(X) \equiv \{i \in \mathcal{P} \setminus \{1\} \mid |u_i(X) - u_1(X)| \leq \Delta\}$$

$$Z^{-}(X) \equiv \{i \in \mathcal{P} \setminus \{1\} \mid u_i(X) - u_1(X) > \Delta\}$$

$$Z^{+}(X) \equiv \{i \in \mathcal{P} \setminus \{1\} \mid u_1(X) - u_i(X) > \Delta\}.$$

That is, $Z^{\Delta}(X)$ is the set of groups where the utility difference $u_i(X)$ and $u_1(X)$ is less than $\Delta$; groups in $Z^{+}$ have utilities less than $u_1(X) - \Delta$; and groups in $Z^{-}$ have utilities greater than $u_1(X) + \Delta$. We observe that $u_{\Delta}(\cdot)$ (Equation 6.5) can be expressed in terms of the above sets:

$$u_{\Delta}(X) = u_1(X) + u_1(X) \cdot |Z^{\Delta}(X)| + \sum_{i \in Z^{+}(X) \cup Z^{-}(X)} u_i(X) + \Delta \left( |Z^{+}(X)| - |Z^{-}(X)| \right).$$

Recall that the sum of utilities is defined as $u(X) \equiv \sum_{i \in \mathcal{P}} u_i(X)$. Substituting $u(X)$ into the above identity, we find

$$u_\Delta(X) = u(X) - \sum_{i \in Z^\Delta(X)} u_i(X) + u_1(X) \cdot |Z^\Delta(X)| + \Delta \left( |Z^+(X)| - |Z^-(X)| \right).$$

Next we bound the price of fairness by bounding the difference between $u(X_E)$ and $u(X_F)$, using the identify above.

The difference we wish to bound is the numerator of the price of fairness (Equation 6.1), which is

$$\begin{aligned} u(X_E) - u(X_F) =& u_\Delta(X_E) - u_\Delta(X_F) && \text{(a)} \\ &+ \sum_{i \in Z^\Delta(X_E)} u_i(X_E) - \sum_{i \in Z^\Delta(X_F)} u_i(X_F) && \text{(b)} \\ &+ u_1(X_F) \cdot |Z^\Delta(X_F)| - u_1(X_E) \cdot |Z^\Delta(X_E)| && \text{(c)} \\ &+ \Delta \left( (|Z^+(X_F)| - |Z^-(X_F)|) - (|Z^+(X_E)| - |Z^-(X_E)|) \right). && \text{(d)} \end{aligned}$$

To bound the price of fairness we will find the maximum value for this expression; we start by maximizing terms (a) and (b) independently:

**(a)** For $X_F$ to be selected over $X_E$ by Algorithm 2 requires that $u_\Delta(X_F) \geq u_\Delta(X_E)$; thus we maximize the price of fairness by making these terms equivalent.

**(b)** Note that for all $i \in Z^\Delta(X)$, the inequality $|u_1(X) - u_i(X)| \leq \Delta$ must hold. Thus, this term is maximized when the identities hold:

$$u_i(X_E) = u_1(X_E) + \Delta \quad \forall i \in Z^\Delta(X_E)$$

$$u_i(X_F) = u_1(X_F) - \Delta \quad \forall i \in Z^\Delta(X_F).$$

After maximizing (a) and (b) with these substitutions, we have

$$\begin{aligned} u(X_E) - u(X_F) \leq\ & (u_1(X_E) + \Delta) \cdot |Z^{\Delta}(X_E)| - (u_1(X_F) - \Delta) \cdot |Z^{\Delta}(X_F)| \\ & + u_1(X_F) \cdot |Z^{\Delta}(X_F)| - u_1(X_E) \cdot |Z^{\Delta}(X_E)| \\ & + \Delta \left( (|Z^{+}(X_F)| - |Z^{-}(X_F)|) - (|Z^{+}(X_E)| - |Z^{-}(X_E)|) \right). \end{aligned}$$

All $u_1(\cdot)$ terms cancel, leaving the following:

$$\begin{aligned} u(X_E) - u(X_F) \leq\ & \Delta \Big( |Z^{\Delta}(X_E)| - |Z^{+}(X_E)| + |Z^{-}(X_E)| \\ & + |Z^{\Delta}(X_F)| + |Z^{+}(X_F)| - |Z^{-}(X_F)| \Big) \\ =\ & \Delta \Big( (|\mathcal{P}| - 1 - 2|Z^{+}(X_E)|) + (|\mathcal{P}| - 1 - 2|Z^{-}(X_F)|) \Big), \end{aligned}$$

where the simplification is due to the identity $|Z^{\Delta}(X_F)| + |Z^{+}(X_F)| + |Z^{-}(X_F)| = |\mathcal{P}| - 1$. This expression is maximized when $Z^{+}(X_E) = Z^{-}(X_F) = \emptyset$, meaning that for all $i > 1$, $u_1(X_E) - u_i(X_E) \leq \Delta$ (in the utilitarian outcome $X_E$), and $u_i(X_F) - u_1(X_F) \leq \Delta$ (in the fair outcome $X_F$). Finally we have $u(X_E) - u(X_F) \leq 2\Delta(|\mathcal{P}| - 1)$, meaning that the price of fairness is at most $2\Delta(|\mathcal{P}| - 1)/u_E$. □

**Theorem 6.10.** *LexFair($\Delta, \mathcal{M}$) Pareto efficient.*

*Proof.* Let $X_F$ be an outcome returned by LexFair($\Delta, \mathcal{M}$) (Algorithm 2). We prove that there is no feasible outcome in $X_E \in \mathcal{M}$ that Pareto-dominates $X_F$, through contradiction. As in Algorithm 2, let $J$ be defined as

$$J \equiv \{i \geq 1 \mid |u_i(X_F) - u_1(X_F)| > \Delta\} \cup \{1\}.$$

As in the proof of Theorem 6.9, we use the following three sets:

$$Z^{\Delta}(X) \equiv \{i \in \mathcal{P} \setminus \{1\} \mid |u_i(X) - u_1(X)| \leq \Delta\}$$

$$Z^{-}(X) \equiv \{i \in \mathcal{P} \setminus \{1\} \mid u_i(X) - u_1(X) > \Delta\}$$

$$Z^{+}(X) \equiv \{i \in \mathcal{P} \setminus \{1\} \mid u_1(X) - u_i(X) > \Delta\}.$$

Suppose that $X_E$ Pareto-dominates $X_F$, and let $I \subset \mathcal{P}$ be the set of groups whose utility strictly improves in $X_E$. By definition, $u_i(X_E) > u_i(X_F)$ for all $i \in I$, and $u_j(X_E) = u_j(X_F)$ for all $j \in \mathcal{P} \setminus I$. For each $i \in I$, let $u_i(X_E) \equiv u_i(X_F) + \delta_i$, with $\delta_i > 0$. Consider three (exhaustive) cases:

**Case 1:** $u_\Delta(X_E) > u_\Delta(X_F)$ In this case, $X_F$ cannot be returned by Algorithm 2 (contradiction).

**Case 2:** $u_\Delta(X_E) = u_\Delta(X_F)$ By definition in Algorithm 2, $X_E \in \mathcal{M}_i$ for all $i \in \mathcal{P}$, since $X_E$ Pareto-dominates $X_F$. Since Algorithm 2 maximizes $u_i(\cdot)$ sequentially for each $i \notin J$, this implies that $u_i(X_F) = u_i(X_E)$ for all $i \notin J$. Therefore, the Pareto improvement must only include groups in $J$, and thus $I \subseteq J$. Since $u_\Delta(X_E) = u_\Delta(X_F)$, we can quantify the Pareto improvement in terms of $u_\Delta(\cdot)$, and we do this separately for each term $u_{\Delta i}(X)$. Consider the following cases, for any $i \in J$:

- **Case 2a:** $i \in Z^{+}(X_F)$ **and** $i \in Z^{\Delta}(X_E)$**.** The change in $u_{\Delta i}(\cdot)$ depends on both the improvement for group $i$ ($\delta_i$) and group 1 ($\delta_1$):

$$\begin{aligned} u_{\Delta i}(X_E) - u_{\Delta i}(X_F) =& u_1(X_E) - (u_i(X_F) + \Delta) \\ &= u_1(X_F) + \delta_1 - u_i(X_F) - \Delta \\ &> 0 \end{aligned}$$

where in the third step we use the fact that $i \in Z^+(X_F)$ implies $u_1(X_F) - u_i(X_F) > \Delta$.

- **Case 2b:** $i \in Z^+(X_F)$ **and** $i \in Z^-(X_E)$**.** The change in $u_{\Delta i}(\cdot)$ is

$$\begin{aligned} u_{\Delta i}(X_E) - u_{\Delta i}(X_F) =& u_i(X_E) - \Delta - (u_i(X_F) + \Delta) \\ &= u_i(X_F) + \delta_i - u_i(X_F) - 2\Delta \\ &= \delta_i - 2\Delta \\ &> 0 \end{aligned}$$

where in the fourth step we add the inequalities due to $i \in Z^+(X_F)$, and $i \in Z^-(X_E)$: since $u_1(X_F) - u_i(X_F) > \Delta$ and $u_i(X_E) - u_1(X_E) > \Delta$, then $\delta_i - 2\Delta > \delta_1 \geq 0$.

- **Case 2c:** $i \in Z^-(X_F)$ **and** $i \in Z^\Delta(X_E)$**.** The change in $u_{\Delta i}(\cdot)$ is

$$\begin{aligned} u_{\Delta i}(X_E) - u_{\Delta i}(X_F) =& u_1(X_E) - (u_i(X_F) - \Delta) \\ &= u_1(X_E) - u_i(X_F) - \delta_i + \delta_i + \Delta \\ &= u_1(X_E) - u_i(X_E) + \delta_i + \Delta \\ &\geq \delta_i \geq 0 \end{aligned}$$

where in the third step we use the fact that $i \in Z^\Delta(X_E)$ implies $|u_1(X_E) - u_i(X_E)| \leq \Delta$.

- **Case 2d:** $i \in Z^-(X_F)$ **and** $i \in Z^+(X_E)$**.** The change in $u_{\Delta i}(\cdot)$ is

$$\begin{aligned} u_{\Delta i}(X_E) - u_{\Delta i}(X_F) =& u_i(X_E) + \Delta - (u_i(X_F) - \Delta) \\ &= u_i(X_F) + \delta_i - u_i(X_F) + 2\Delta \end{aligned}$$

$$= \delta_i + 2\Delta$$

$$\geq 2\Delta \geq 0.$$

- **Case 2e:** $i$ **is in both** $Z^-(X_F)$ **and** $Z^-(X_E)$ ***or*** $i$ **is in both** $Z^+(X_F)$ **and** $Z^+(X_E)$

  In this case $u_{\Delta i}(X_E) - u_{\Delta i}(X_F) = \delta_i > 0$.

Note that in addition to these cases, if the Pareto improvement involves group 1, then $u_\Delta(\cdot)$ improves by $\delta_1 > 0$. Note that in cases 2a, 2b, and 2e, any Pareto improvement strictly increases the term $u_{\Delta i}(\cdot)$. In cases 2c and 2d, $\delta_1 > 0$ and therefore the Pareto improvement also results in a strict increase to $u_\Delta(\cdot)$. Therefore, any Pareto improvement involving any group $i \in J$ increases $u_\Delta(\cdot)$ (contradiction).

**Case 3:** $u_\Delta(X_E) < u_\Delta(X_F)$ According to Cases 2a, 2b, 2c, 2d, and 2e above, any Pareto improvement involving groups in $J$ *strictly increases* $u_\Delta(\cdot)$. Next we show that any Pareto improvement involving $i \notin J \equiv Z^\Delta(X_F)$ also results in a strict increase to $u_\Delta(\cdot)$.

- **Case 3a:** $i \in Z^\Delta(X_F)$ **and** $i \in Z^+(X_E)$ The change in $u_{\Delta i}(\cdot)$ is

$$\begin{aligned} u_{\Delta i}(X_E) - u_{\Delta i}(X_F) =& u_i(X_E) + \Delta - u_1(X_F) \\ &= u_i(X_F) - u_1(X_F) + \delta_i + \Delta \\ &\geq \delta_i \geq 0 \end{aligned}$$

  where in the third step we use the fact that $i \in Z^\Delta(X_F)$ implies $|u_1(X_F) - u_i(X_F)| \leq \Delta$.

- **Case 3b:** $i \in Z^{\Delta}(X_F)$ **and** $i \in Z^{-}(X_E)$

$$\begin{aligned} u_{\Delta i}(X_E) - u_{\Delta i}(X_F) =& u_i(X_E) - \Delta - u_1(X_F) \\ &= u_i(X_E) - \Delta - u_1(X_F) + \delta_1 - \delta_1 \\ &= u_i(X_E) - u_1(X_E) + \delta_1 - \Delta \\ &> \delta_1 \geq 0 \end{aligned}$$

  where in the fourth step we use the fact that $i \in Z^{-}(X_E)$ implies $u_i(X_E) - u_1(X_E) > \Delta$.

- **Case 3c:** $i$ **is in both** $Z^{\Delta}(X_F)$ **and** $Z^{\Delta}(X_E)$ In this case there is no change to $u_{\Delta}(\cdot)$.

Thus, no Pareto improvement results in a decrease to $u_{\Delta}(\cdot)$ (contradiction).

□

### 6.4.5 Hybrid Fairness in Kidney Exchange

The hybrid-lexicographic algorithm defined in Algorithm 2 is easily applied to kidney exchange, with $u_H$ and $u_L$ the total utility received by highly-sensitized and lowly-sensitized patients, respectively,

$$u_{\Delta}(M) = \begin{cases} u_L(M) + u_H(M) - \Delta & \text{if } u_L(M) - u_H(M) > \Delta \\ 2u_H(M) & \text{if } |u_L(M) - u_H(M)| \leq \Delta \\ u_L(M) + u_H(M) + \Delta & \text{if } u_H(M) - u_L(M) > \Delta \end{cases} \tag{6.7}$$

In the following section, we demonstrate the practical effectiveness of the hybrid-lexicographic algorithm by testing it on real kidney exchange data.

## 6.5 Experiments

In this section, we compare the behavior of $\alpha$-lexicographic, weighted, and hybrid-lexicographic fairness. Code for these experiments is available on GitHub.[2] We use each method to find the optimal fair outcomes for 314 real kidney exchanges from the United Network for Organ Sharing (UNOS), collected between 2010 and 2016. We solve the KEP we use the PICEF formulation of Dickerson et al. [109], with cycle cap 3 and various chain caps. In real exchanges, not all recommended edges in a matching result in successful transplants. To reflect this uncertainty, we use the concept of failure-aware kidney exchange introduced in [107]: all edges in the exchange can fail with probability $(1 - p)$; in this version of the KEP we maximize *expected* matching weight, considering edge success probability $p$.

### 6.5.1 Procedure

For each UNOS exchange graph $G$, we use the following procedure to implement each fair algorithm. We repeat the following procedure for chain caps 0, 3, 10, and 20, and for edge success probabilities $p = 0.1n$, with $n = 1, 2, \ldots, 10$.

1. Find the "efficient" matching $M_E$ by solving the KEP on $G$.

2. Find the "fair" matching $M_F$ by solving the KEP on $G' = (V, E')$, where each edge $e \in E'$ has weight 1 if $e$ ends in $V_H$ and 0 otherwise.

3. **Weighted Fairness:** Find the $\gamma$-fair matching $M_\gamma$ by solving the KEP on $G^\gamma = (V, E^\gamma)$, where each edge $e \in E^\gamma$ has weight $1 + \gamma$ if $e$ ends in $V_H$ and 1 otherwise. After finding $M_\gamma$, the reported utilities are calculated using edge weights of $E$ and not $E'$. We use weight parameters $\gamma = 2n$, with $n = 0, 1, 2, \ldots, 10$.

[2] https://github.com/duncanmcelfresh/FairKidneyExchange

4. **$\alpha$-Lexicographic Fairness:** Find the $\alpha$-fair matching $M_\alpha$ by solving the KEP on $G$, with the additional constraint $u_H(M_\alpha) \geq \alpha u_H(M_E)$. We use parameters $\alpha = 0.1n$, with $n = 0, 1, 2, \ldots, 10$.

5. **Hybrid-Lexicographic Fairness:** Find the $\Delta$-fair matching $M_\Delta$ using the $\alpha$-fair matchings $M_\alpha$, and Algorithm 1. That is, $M_\Delta = \text{LexFair}(\Delta, M_\alpha)$. We use parameters $\Delta = 0.1n \cdot u(M_E)$, with $n = 0, 1, 2, \ldots, 10$.

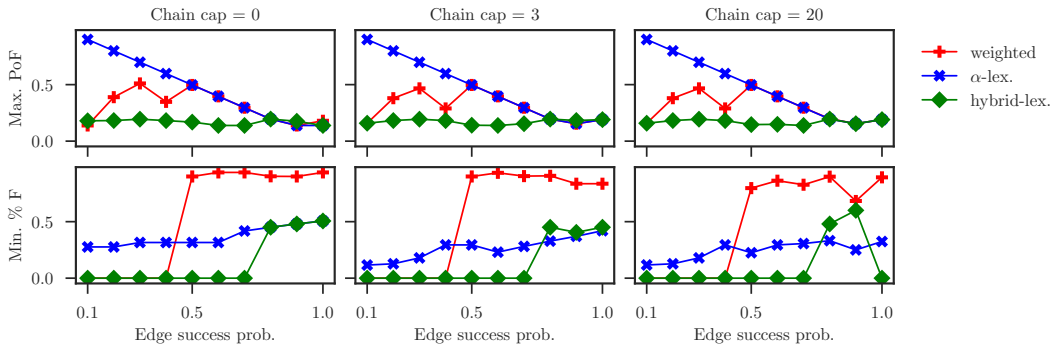



FIGURE 6.5: Worst-case price of fairness and $\%F$ for various edge success probabilities, and fairness parameters $\alpha = 0.1$, $\gamma = 0.1$, $\Delta = 0.1u(M_E)$, across all UNOS graphs.

Throughout this procedure, we calculate the utility of the efficient matching ($u_E$) and the fair matching ($u_F$) for each UNOS graph, and for each fair algorithm—with parameters $\alpha \in [0, 1]$, $\gamma \in [0, 20]$, and $\Delta \in [0, u(M_E)]$.

There are two important outcomes of each fair algorithm: Price of Fairness (PoF), and fraction of the fair score ($\%F$). To calculate PoF we use the definition in Equation 6.1, using $u_E$ and $u_F$. We define $\%F$ as the fraction of the maximum highly sensitized utility, achieved by $M_{\{\alpha,\gamma,\Delta\}}$, defined as

$$\%F(M_{\{\alpha,\gamma,\Delta\}}, M_F) = u_H(M_{\{\alpha,\gamma,\Delta\}})/u_H(M_F).$$

PoF and $\%F$ indicate the efficiency loss and the fairness of each method, respectively.

### 6.5.2 Results and Discussion

Each fair algorithm offers a parameter that balances efficiency and fairness. Two of these methods guarantee a certain outcome: $\alpha$-lexicographic guarantees fairness, but allows high efficiency loss, while hybrid-lexicographic bounds overall efficiency loss. Weighted fairness makes no guarantees.

The price of fairness can be high in real exchanges, especially when edge success probability $p$ is small. In failure-aware kidney exchange, cycles and chains of length $k$ receive utility proportional to $p^k$. Fair matchings often use longer cycles and chains than the efficient matching, in order to reach highly sensitized patients; this leads to a high price of fairness when $p$ is small.

Even when $\alpha$ and $\gamma$ are small, there are cases when both $\alpha$-lexicographic and weighted fairness allow for a high PoF. This becomes worse with lower edge probability. Figure 6.5 shows the worst-case PoF and $\%F$ for each method, for the smallest parameters tested, for a range of edge success probabilities; results for all parameter values are in Appendix B.2.

Hybrid-lexicographic fairness limits PoF within the guaranteed bound of 0.2; this comes at the cost of a low $\%F$—when edge success probability is small, hybrid-lexicographic fairness awards zero fair utility in the worst case. $\alpha$-lexicographic fairness produces the opposite behavior: $\%F$ is always larger than the guaranteed bound of 0.1, but the worst-case price of fairness grows steadily as edge probability decreases.

Theory suggests that the price of fairness is small on denser random graphs (see Section 6.2). We empirically confirm this theoretical finding by calculating the worst-case price of fairness and $\%F$ for random graphs of various sizes generated from real data; these results are given in Appendix B.2 In this case—when the price of fairness

is small—$\alpha$-lexicographic fairness may be appropriate, as overall efficiency loss is not severe.

Both $\alpha$-lexicographic and hybrid-lexicographic fairness are useful, depending on the desired outcome. Policymakers may choose between these methods, and set the parameters $\alpha$ and $\Delta$ to guarantee either a minimum $\%F$ or a maximum price of fairness.

## 6.6 Discussion

We addressed the classical problem of balancing fairness and efficiency, with a specific focus on kidney exchange. Extending work by Ashlagi and Roth [24] and Dickerson et al. [108], we show that the theoretical price of fairness is small on a random graph model of kidney exchange, when both cycles and chains are used. However this model is too optimistic—real kidney exchanges are less certain and more sparse, and in reality the price of fairness can be unacceptably high.

Drawing on work by Hooker and Williams [168], which is not applicable to kidney exchange, we provided the first approach to incorporating fairness into kidney exchange in a way that prioritizes marginalized participants, but also comes with acceptable worst-case guarantees on overall efficiency loss. Furthermore, our method is easily applied as an objective in the mathematical-programming-based clearing methods used in today's fielded exchanges. Using data from a large fielded kidney exchange, we showed that our method bounds efficiency loss while also prioritizing marginalized participants when possible.

## 6.7 Authors and Publication

This chapter was written by Duncan McElfresh and John P Dickerson. Most of this chapter appeared at the 2018 AAAI Conference on Artificial Intelligence (AAAI-18). [215].

# Part II

# Practical Considerations in Matching and Resource Allocation: Examples from Kidney Exchange & Blood Donation

# Chapter 7: Some Differences between Theory and Practice

Part I of this thesis focused on the kidney exchange clearing problem (KEP), which is one aspect of the real and complicated process of kidney exchange. The previous chapters focused mainly on technical aspects of the KEP: how algorithms perform when the input parameters are uncertain, or how different groups of patients are impacted when different objective functions are used. These analyses are useful for several reasons: first, they help us develop better solution methods for the KEP, which in turn can improve the effectiveness of fielded kidney exchange programs. Second, the KEP is a variant of *cycle packing*—a general combinatorial optimization problem which arises in several real-world scenarios. The insights gained by studying the KEP can translate to these other problem domains relatively easily.

On the other hand, algorithmic research on the KEP is often too abstract, or too theoretical to be immediately useful. This problem is not isolated to kidney exchange: a large proportion of computer science and applied mathematics research is too abstract to be deployed without substantial modification.

However, novel computer science research can have greater immediate impact, by focusing more on the underlying real-world problems we are trying to solve. The following two chapters describe two applied research endeavors that focus more explicitly on the underlying application than on a mathematical model; both of these

involve substantial involvement with stakeholders, and both have resulted in improvements to a fielded algorithmic system. This research involves several challenges which are not amenable to mathematical proofs or computational power, but are necessary aspects of applied research. Below I outline three considerations that played a role in this research; I hope these will be of use to computer scientists and mathematicians interested in conducting applied research.

**Stakeholder Buy-In** It goes without saying that applied research should involve people who will be directly impacted by the research and its findings—that is *stakeholders*. Obtaining stakeholder *buy-in* to a research project is often necessary, since stakeholders usually provide the data, the domain expertise, and the access to systems or people required for applied research. Obtaining and maintaining buy-in is an ongoing process, which usually involves stakeholder oversight for the research as well as the resulting publications. This can be seen alternatively as a constraint on the direction of the research, or as an opportunity to engage with a wider community. See [197] for an example of applied computer science research that explicitly considers stakeholder buy-in. In deployed algorithmic systems there can be a huge number of stakeholders, and it may not be necessary or possible to obtain buy-in from everyone. Suppose for example we wish to run an experiment on an online advertising platform. To conduct this research it might be *necessary* to obtain buy-in from the platform's product managers and engineering team. However there are other stakeholders—the advertisers and their customers—from whom we need not, and likely cannot, obtain buy-in. This raises questions of ethics, which we discuss next.

**Ethical Considerations** There is a wide variety of ethical concerns raised by deployed algorithms, many of which are specific to the application domain; these concerns are mainly directed at balancing the harms and benefits to stakeholders. A comprehensive discussion of these concerns would take several more chapters, so I will highlight only a few issues here. Since many modern algorithms are driven by vast amounts of personal data, stakeholder *privacy* is one of the greatest ethical concerns, and an active area of research [142]. Privacy is especially important when stakeholders come from vulnerable groups (such as children, prisoners, and disabled persons), or when the application involves sensitive personal data (as in health care applications or private messaging services). Another important issue is *access*: if a useful technology can only be accessed by certain groups of people, its benefits will be unequally distributed; this can be addressed through *inclusive design* [100]. For example, Johansson et al. [174] finds that people with language- and memory-related disabilities face difficulty in using modern internet applications. There is however another side to access: algorithms present an attractive low-cost alternative to (human) experts, and a low-quality algorithmic alternative can replace high-quality service, particularly for marginalized populations. This is a real risk in health care: AI systems have been proposed as an alternative to costly medical experts, particularly in under-resourced areas [154, 313]. While these AI systems are "better than nothing", they are likely to remain less safe and provide lower-quality care than human experts. Thus due to cost constraints, advanced algorithmic systems might *prevent* access to high-quality health care, by providing an attractive but low-quality alternative. Another important ethical consideration is algorithmic *bias* and *(un)fairness*. Biases can arise from a variety of places: the training dataset, assumptions made in the design process, malicious intent, and—most commonly—negligence. Several real instances of algorithmic bias have been reported by the media and academia;

the most alarming instances involve explicit racism or sexism (see Section 14.2). A high-level overview of this topic can be found in [95].

**Legal Considerations** While ethics issues abound, there are currently few cases where algorithmic oversight is explicitly a legal matter. There are for example several regulations meant to protect stakeholder privacy, including the European Union's General Data Protection Regulation (GDPR) [311], the United States' HIPAA Privacy Rule [1], and California's CCPA [2]. Another example is facial recognition: several local and state governments have recently banned the use of facial recognition systems, mainly targeting law enforcement.[1] There are several reasons that regulators may be hesitant to restrict the development or use of algorithms. Large-scale applications of AI and ML are relatively young, and it is challenging to weigh their harms and benefits. Furthermore, regulators are hesitant to stifle research efforts. However even without regulation, deployed algorithms are still under close scrutiny—particularly in medicine [42, 193, 320] and consumer protection [127, 128].

The following two chapters cover two applied projects; and both deal with the considerations described above. First, Chapter 8 explores an algorithmic process for *pre-screening* potential donors in—once again—kidney exchange. This research is in fact motivated by uncertainty in transplant quality and feasibility, which is also the focus of Chapters 3, 4, and 5. However these earlier chapters addressed transplant uncertainty using mathematical optimization, which requires modifying kidney exchange policy. Here we develop an approach that fits *within* the workflow of a fielded exchange, rather than changing it. Then, in Chapter 9 we describe a new matching system for connecting potential blood donors with donation opportunities. There are several practical challenges here: we aim to engage potential donors

[1] There are currently two US states and 18 cities which have banned some uses of facial recognition, with many other state and local laws currently under consideration (https://www.banfacialrecognition.com/map/).

without overwhelming them with notifications; blood recipients need to be treated *fairly*, otherwise they may opt out of the matching system; maintaining user privacy is especially important, since blood donation involves sensitive health information.

Finally, it is important to note that these practical considerations are best addressed through *direct collaboration* with stakeholders. Going one step further, it is important to understand how stakeholders both interact with and perceive of algorithmic systems; this is the focus of Part III of this thesis.

# Chapter 8: Uncertainty in Kidney Exchange: Pre-Screening

## 8.1 Introduction

This chapter introduces a practical way to deal with uncertainty in kidney exchange, using pre-screening. Like Chapters 3 and 5 we are primarily concerned with uncertainty in the *feasibility* of potential transplants: if a donor is matched with a potential recipient, will the transplant actually occur? Planned transplants may *fail* for a variety of reasons: for example, medical testing may reveal that the donor and recipient are incompatible (a *positive crossmatch*); the recipient or their medical team may reject a donor organ in order to wait for a better match; or the donor may decide to donate elsewhere before the exchange is planned. Failed transplants are especially troublesome in kidney exchange, due to the cycle and chain structures used: for example, suppose that a cyclical swap is planned between three patient/donor pairs; if any one of the planned transplants fails, then none of the other transplants in that cycle can occur. Unfortunately, in fielded exchanges it is quite common for planned transplants to fail. For example, the United Network for Organ Sharing (UNOS) estimates that in FY2019, about 85% of their planned kidney transplants failed [198].

Various matching algorithms have been proposed that aim to mitigate transplant failures (for example, using stochastic optimization [16, 112], robust optimization [216] (Chapter 3), or conditional value at risk [46] (Chapter 5). However, implementing these strategies would require modifying fielded matching algorithms–which in many cases would require changing law or policy. One way to avoid failures without modifying the matching algorithm is to *pre-screen* potential transplants [54, 56, 198], by communicating with the recipients' medical team and possibly using additional medical tests. Pre-screening transplants is costly, as it requires scarce time and resources. Furthermore, there are often many thousand potential transplants in any given exchange; selecting which transplants to screen is not easy.

In this chapter we investigate methods for selecting a limited number of transplants to pre-screen, in order to "guide" the matching algorithm to a better outcome. We formalize this as a multistage stochastic optimization problem, and we consider both an *offline* setting (where screenings are selected all at once), and an *online* setting (where screenings are selected sequentially).

**Related Work.** Prior work has addressed potential transplant failures; our model is inspired by Dickerson et al. [112]. Pre-screening potential transplants has also been addressed in prior work ([56, 226], and § 5.1 of [103]), and our model is similar to stochastic matching and stochastic $k$-set packing [35]. However there are substantial differences between these models and ours: (a) many prior approaches assume that a large number of transplants may be pre-screened [56, 226]–on the order of one for each patient in the exchange; we assume far fewer screenings are possible; (b) prior work often assumes a *query-commit* setting–where successfully pre-screened transplants *must* be matched. Instead we assume that non-screened transplants may also

be matched–which more-accurately represents the way that modern exchanges operate; (c) most prior work assumes that transplants that pass pre-screening are guaranteed to result in a transplant. In reality, transplants often fail after pre-screening, a fact reflected in our model.

One of our approaches is based on *Monte Carlo Tree Search* (MCTS), which allows efficient exploration of intractably large decision trees. While MCTS is primarily associated with Markov decision processes and game-playing [60], it has been used successfully for combinatorial optimization [177]. We use a version of MCTS, Upper Confidence Bounds for Trees (UCT), which balances exploration and exploitation by treating each tree node as a multi-armed bandit problem [29, 186].

### Contributions

1. (§ 8.2) We formalize the *policy-constrained edge query problem*: where a decision maker (such as a kidney exchange program) selects a set of potential edges (potential transplants) to pre-screen, prior to constructing a final packing (a set of transplants) using a fixed algorithm. This model generalizes existing models in the literature, as edge failure probabilities depend on whether or not the edge is pre-screened. Further, we allows for context-specific constraints, such as those imposed by public policy or the particular hospital or exchange.

2. (§ 8.3) We prove that when the decision maker uses a max-weight packing policy (the most common choice among fielded exchanges), the edge query problem is both non-monotonic and non-submodular in the set of queried edges. Despite these worst-case findings we show that this problem is nearly monotonic for real and synthetic data, and simple algorithms perform quite well. On the other hand, when the decision maker uses a *failure-aware* (stochastic)

packing policy, the edge query problem becomes monotonic under mild assumptions.

3. (§ 8.4) We conduct numerical experiments on both simulated and real exchange data from the United Network for Organ Sharing (UNOS). We demonstrate that our methods substantially outperform prior approaches and a randomized baseline.

## 8.2 The Policy-Constrained Edge Query Problem

As in previous chapters, kidney exchanges are represented by a graph $G = (E, V)$ where vertices $V$ represent (incompatible) patient-donor pairs, and non-directed donors (NDDs) who are willing to donate without receiving a kidney in return. Directed edges $e \in E$ between vertices represent potential transplants from the donor of one vertex to the patient of another. Edge weights represent the "utility" of an edge, and are typically set by exchange policy. Solutions to a kidney exchange problem (henceforth, *matchings*) consist of both directed *cycles* on $G$ containing only patient-donor pairs, and directed *chains* beginning with an NDD and passing through one or more pairs. Each vertex may participate in only one edge in a matching–as each vertex can donate and receive at most one kidney.

The dilemma of edge failures is illustrated in the example exchange graph shown in Figure 8.1. This exchange consists of a 3-chain (dashed edges) and two 2-cycles (solid edges). Suppose the decision-maker queries edge $e_A$: if $e_A$ is accepted, then the chain from the NDD ($n$) through pairs $(d_1, p_1)$, $(d_2, p_2)$, and $(d_3, p_3)$, i.e., the dashed edges, can be included in the matching. However if $e_A$ is queried and rejected, then the NDD cannot initiate the chain, and only the cycles may be matched. In our model, if $e_A$ is not queried then it may still be matched.

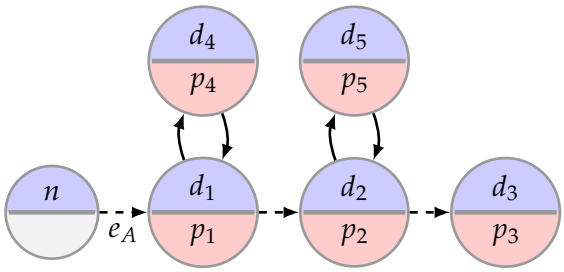



FIGURE 8.1: Sample exchange graph with a 3-chain (dashed edges) and two 2-cycles (solid edges). The NDD is denoted by $n$, and each patient (and associated donor) is denoted by $p_i$ ($d_i$). If edge $e_1$ is not queried, or queried and *accepted*, then the chain may be included in the final matching. However if edge $e_A$ is queried and *rejected*, then only the 2-cycles may be included in the final matching.

In this chapter we use a *cycle-chain* representation for matchings:[1] let $C$ represent cycles and chains in $G$, where each cycle and chain corresponds to a list of edges; as is standard in modern exchanges, we assume that cycles and chains are limited in length. Matchings are expressed as a binary vector $\boldsymbol{x} \in \{0,1\}^{|C|}$, where $\boldsymbol{x}_c = 1$ if cycle/chain $c$ is in the matching, and 0 otherwise. Let $\boldsymbol{w}_c$ be the weight of cycle/chain $c$ (the sum of $c$'s edge weights). Let $\mathcal{M}$ denote the set of *legal matchings*–that is, the set of vertex-disjoint cycles and chains on $G$, with chains up to length $L$ and cycles up to length $K$. Cycle length cap $K$ and chain length cap $L$ are set by the each exchange, typically $K = 3$ and $L = 4$. These length limits serve two purposes: (a) longer cycles and chains are risky, in that they are likely to be impacted by edge failure, and (b) policy often requires that all transplants in a cycle or chain are completed *simultaneously*, and most transplant centers can only accommodate a handful of simultaneous transplants. The total weight of a matching is simply the summed weights of all its constituent cycles and chains: $\sum_{c \in C} \boldsymbol{x}_c \boldsymbol{w}_c$. We denote *sets* of edges using binary vectors, where $\boldsymbol{q} \in \{0,1\}^{|E|}$ represents the set of all edges with $\boldsymbol{q}_e = 1$. In the remainder of this chapter we refer to pre-screening a transplant as *querying an edge*, in order to be consistent with the literature.

[1] Our experiments use the PICEF formulation, which is more compact and equivalent [109].

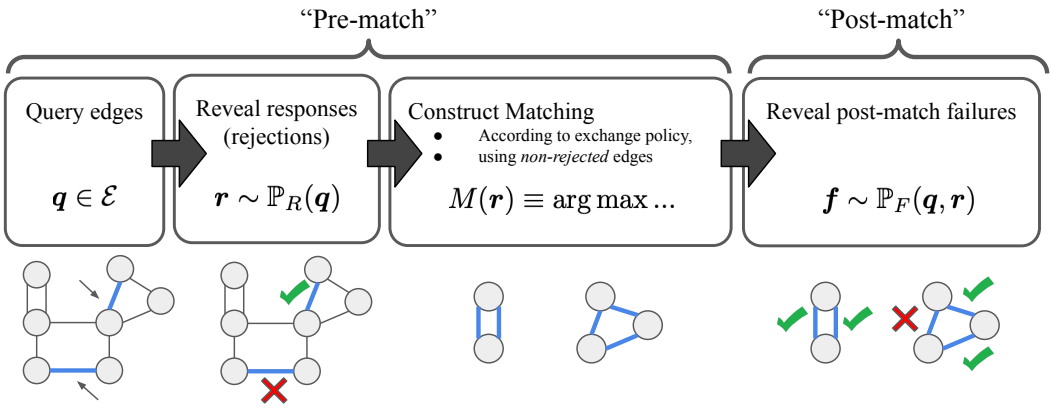



FIGURE 8.2: Single-stage edge selection: First, edges are selected to be queried, and responses revealed. Then, a final matching is constructed according to the exchange's matching policy. Finally, the post-match edge failures are revealed.

**Selecting Edge Queries.** Our setting consists of two phases (see Figure 8.2): during *pre-match*, the decision maker selects edges to query, and each queried edge is either accepted or rejected; then the decision maker constructs a matching using a fixed policy. During *post-match*, each match edge either fails (no transplant) or succeeds (the transplant proceeds). We consider two version of the pre-match phase: in the *single-stage* version, the decision maker selects all queries before observing edge responses (accept/reject); in the *multi-stage* version, one edge is selected at a time and responses are observed immediately.

Unlike most prior work, edges in our model may fail during both the pre- and post-match phase. For example, suppose the decision maker queries an edge from a 60-year-old non-directed donor, to a 35-year-old recipient; if the recipient or their medical team rejects the elderly donor and decides to wait for a younger donor, this is a pre-match rejection. Instead suppose the edge is not queried, and it is included in the final matching; if medical screening reveals that the patient and donor are incompatible, this is a post-match failure. We refer to pre-match failures as *rejections* and post-match failures as *failures*; however we make no assumption about their cause. We represent potential failures and rejections using binary random variables:

$\boldsymbol{r} \in \{0,1\}^{|E|}$ denotes pre-match rejections, where $\boldsymbol{r}_e = 1$ if $e$ is queried and rejected, and 0 otherwise ($\boldsymbol{r}_e = 0$ for all non-queried edges). Similarly $\boldsymbol{f} \in \{0,1\}^{|E|}$ denotes post-match failures, where $\boldsymbol{f}_e = 1$ if edge $e$ fails post-match, and 0 otherwise. We assume that the distribution of rejections $\boldsymbol{r} \sim \mathbb{P}_R(\boldsymbol{q})$ is known, and depends on $\boldsymbol{q}$; we assume the distribution of failures $\boldsymbol{f} \sim \mathbb{P}_F(\boldsymbol{q}, \boldsymbol{r})$ is known, and depends on both $\boldsymbol{q}$ and $\boldsymbol{r}$.

Rejections and failures impact the matching through the *weight* of each cycle and chain. If any cycle edge fails, then *no* transplants in the cycle can proceed; if a chain edge fails, than all edges *following* it cannot proceed.[2] Suppose we observe failures $\boldsymbol{f}$; the *final matching weight* of $c$ is

$$F(c, \boldsymbol{y}) \equiv \begin{cases} \sum_{e \in c} w_e & \text{if } \sum_{e \in c} \boldsymbol{y}_e = 0 \\ 0 & \text{if } c \text{ is a cycle and } \sum_{e \in c} \boldsymbol{y}_e > 0 \\ \sum_{e \in c'} w_e & \text{if } c \text{ is a chain, where } c' \text{ includes all edges up to the first failed edge.} \end{cases}$$

Thus the *post-match expected weight* of matching $\boldsymbol{x}$, due to both rejections $\boldsymbol{r}$ and failures $\boldsymbol{f}$, is

$$W(\boldsymbol{x}; \boldsymbol{q}, \boldsymbol{r}) \equiv \mathop{\mathbb{E}}_{\boldsymbol{f} \sim \mathbb{P}_F(\boldsymbol{q}, \boldsymbol{r})} \left[ \sum_{c \in C} \boldsymbol{x}_c \, F(c, \boldsymbol{r} + \boldsymbol{f}) \right].$$

**Matching Policy** In this chapter we assume that the final matching is constructed using a fixed matching policy, which uses only *non-rejected* edges; we denote this policy by $M(\boldsymbol{r})$. We focus primarily on the *max-weight* policy $M^{\texttt{MAX}}(\cdot)$, which is used by most fielded exchanges, and the *failure-aware* policy $M^{\texttt{FA}}(\cdot)$, which maximizes the

[2] This assumes that chains can be *partially* executed: for example, suppose that the $4^{th}$ edge in a 10-edge chain fails; the first three edges can still be matched, and the post-failure chain weight sums only these three edges. Not all fielded exchanges use this policy: some exchanges cancel the entire chain if one of its edges fails.

expected post-match weight [112]:

$$M^{\texttt{MAX}}(\boldsymbol{r}) \in \arg\max_{\boldsymbol{x}\in\mathcal{M}} \sum_{c\in C} \boldsymbol{x}_c\, F(c,\boldsymbol{r})\,, \qquad M^{\texttt{FA}}(\boldsymbol{r}) \in \arg\max_{\boldsymbol{x}\in\mathcal{M}(\boldsymbol{r})} \mathop{\mathbb{E}}_{f\sim\mathbb{P}_F(\boldsymbol{q},\boldsymbol{r})} \left[\sum_{c\in C} \boldsymbol{x}_c\, F(c,\boldsymbol{r}+\boldsymbol{f})\right].$$

Evaluating this policy requires solving a kidney exchange clearing problem, which is NP-hard [5]. However, state-of-the-art method can solve realistic kidney exchange clearing problems in fractions of a second (e.g., our experiments use the PICEF method of Dickerson et al. [109]); thus, throughout this chapter we treat this policy as a low- or no-cost oracle.

Next we formalize the *edge selection problem*–the main focus of this chapter. We denote by $\mathcal{E}$ the set of "legal" edge subsets, subject to exchange-specific constraints; we assume that $\mathcal{E}$ is a matroid with ground set $E$. For example, the decision maker may limit the number of queries issued to any one medical team (vertex in $G$) or transplant center (group of vertices). We aim to select an edge set $\boldsymbol{q} \in \mathcal{E}$ which maximizes the *expected weight* of the final matching. These edges are selected using only the distribution of future rejections and failures; we take a *stochastic optimization* approach, maximizing the expected outcome over this uncertainty.

**Single-Stage Setting.** The single-stage policy-constrained edge selection problem (henceforth, the *edge selection problem*) is expressed as

$$\max_{\boldsymbol{q}\in\mathcal{E}} V^S(\boldsymbol{q})\,, \qquad \text{with} \quad V^S(\boldsymbol{q}) \equiv \mathop{\mathbb{E}}_{\boldsymbol{r}\sim\mathbb{P}_R(\boldsymbol{q})} \left[\, W\left(M(\boldsymbol{r});\boldsymbol{q},\boldsymbol{r}\right) \,\right], \tag{8.1}$$

where, $M(\boldsymbol{r})$ denotes the matching policy after observing rejections $\boldsymbol{r}$, and $W(\boldsymbol{x};\boldsymbol{q},\boldsymbol{r})$ denotes the post-match expected weight of matching $\boldsymbol{x}$. Exact evaluation of $V^S(\boldsymbol{q})$ is often intractable, as the support of $\mathbb{P}_R(\boldsymbol{q})$ grows exponentially in $|\boldsymbol{q}|$. In experiments we approximate $V^S(\boldsymbol{q})$ using sampling, and these approximations converge for a

moderate number of samples (see Appendix C.1).

**Multistage Setting.** In the multi-stage setting, edge rejections are observed immediately after each edge is queried. The multi-stage problem is expressed as

$$\max_{q^1 \in \mathcal{E}_1} \mathop{\mathbb{E}}_{r^1 \sim \mathbb{P}_R(q^1)} \left[ \max_{q^2 \in \mathcal{E}_1} \mathop{\mathbb{E}}_{r^2 \sim \mathbb{P}_R(q^2)} \left[ \dots \max_{q^K \in \mathcal{E}_1} \mathop{\mathbb{E}}_{r^K \sim \mathbb{P}_R(q^K)} \left[ W\left(M(\boldsymbol{r}); \boldsymbol{q}, \boldsymbol{r}\right) \right] \right] \dots \right], \quad (8.2)$$

where $\boldsymbol{q} \equiv \sum_{i=1}^{K} q^i$ denotes all queried edges, $\boldsymbol{r} \equiv \sum_{i=1}^{K} r^i$ denotes all rejections, and $\mathcal{E}_1 \subseteq \mathcal{E}$ be denotes the legal edge subsets containing only one edge. First, we observe that Problems 8.1 and 8.2 require evaluating a matching policy $M(\boldsymbol{r})$. In the case of kidney exchange, evaluating both the max-weight policy $M^{\texttt{MAX}}(\cdot)$ and the failure-aware policy $M^{\texttt{FA}}(\cdot)$ require solving NP-hard problems; thus Problems 8.1 and 8.2 are at least NP-hard as well.

However, regardless how difficult the matching policy is, the question remains whether *edge selection* is is hard. We observe that while these problems are difficult in principle, experiments (§ 8.4) show that they are easy in practice. Proofs of the following propositions can be found in Appendix C.3.

**Proposition 8.1.** *With matching policy $M^{\texttt{MAX}}(\cdot)$, the objective of Problem 8.1 is non-monotonic in the number of queried edges, even with independent edge distributions.*

In other words, querying additional edges can sometimes lead to a *worse* outcome. This is somewhat counter-intuitive; one might think that providing additional information to the matching policy would strictly improve the outcome. This is a worst-case result–and in fact our experiments demonstrate that querying edges almost always leads to a better final matching weight.

**Proposition 8.2.** *With matching policy $M^{\texttt{MAX}}(\cdot)$, the objective of Problem 8.1 is non-submodular in the set of queried edges.*

In other words, certain edges are *complementary* to each other–and querying complementary edges simultaneously can yield a greater improvement than querying them separately. Taken together, these propositions indicate that single-stage edge selection with matching policy $M^{\texttt{MAX}}(\cdot)$ is a challenging combinatorial optimization problem. On the other hand, using the failure-aware matching policy $M^{\texttt{FA}}(\cdot)$ allows us to avoid some of these issues under mild assumptions.

**Assumption 8.1.** *Let $\boldsymbol{q}, \boldsymbol{r} \in \{0,1\}^{|E|}$ denote initial edge queries and responses. Let $\boldsymbol{q}'$ be* additional *edges, such that $\boldsymbol{q} + \boldsymbol{q}' \in \{0,1\}^{|E|}$ denotes an augmented edge set; let $\boldsymbol{r}' \in \{0,1\}^{|E|}$ denote the responses to edges $\boldsymbol{q}'$ only. We assume that for any such $\boldsymbol{q}$, $\boldsymbol{r}$, and $\boldsymbol{q}'$,*

$$\mathbb{E}\left[\boldsymbol{r} + \boldsymbol{f} \mid \boldsymbol{q}, \boldsymbol{r}\right] \geq \mathbb{E}\left[\boldsymbol{r} + \boldsymbol{r}' + \boldsymbol{f} \mid \boldsymbol{q} + \boldsymbol{q}', \boldsymbol{r}\right] \ .$$

Intuitively, Assumption 8.1 excludes distributions where queries arbitrarily increase edge failure or rejection. For example, Assumption 8.1 disallows the following distribution: suppose all edges are independent; all queried edges are accepted ($P(\boldsymbol{r}_e = 1 \mid \boldsymbol{q}) = 0$ for all $\boldsymbol{q}$), all accepted edges have failure probability 0.5 ($P(\boldsymbol{f}_e = 1 \mid \boldsymbol{q}_e = 1, \boldsymbol{r}_e = 0) = 0.5$), and all non-queried edges have failure probability 0.1 ($P(\boldsymbol{f}_e = 1 \mid \boldsymbol{q}_e = \boldsymbol{r}_e = 0) = 0.1$). In this case, if an edge is not queried, then it has overall rejection or failure probability 0.1 (i.e., $\mathbb{E}[\boldsymbol{r}_e + \boldsymbol{f}_e \mid \boldsymbol{q}, \boldsymbol{r}] = 0.1$ with $\boldsymbol{q}_e = 0$); if this edge is queried, then it has rejection or failure probability 0.5 (i.e., $\mathbb{E}[\boldsymbol{r}_e + \boldsymbol{r}'_e + \boldsymbol{f}_e \mid \boldsymbol{q} + \boldsymbol{q}', \boldsymbol{r}] = 0.5$ with $\boldsymbol{q}'_e = 1$).

We also assume that edge failures are *independent*.

**Definition 4** (Edge Independence)**.** Two edges $e, e' \in E$ are *independent* if (a) their rejection distributions are conditionally independent, given whether or not they were queried:

$$\boldsymbol{r}_e \perp\!\!\!\perp \boldsymbol{r}_{e'} \mid \boldsymbol{q}_e \quad \text{and} \quad \boldsymbol{r}_e \perp\!\!\!\perp \boldsymbol{r}_{e'} \mid \boldsymbol{q}_{e'}$$

and (b) their failure distributions are conditionally independent, given whether or not they were queried and rejected:

$$f_e \perp\!\!\!\perp f_{e'} \mid q_e, r_e \quad \text{and} \quad f_e \perp\!\!\!\perp f_{e'} \mid q_{e'}, r_{e'}\,.$$

**Proposition 8.3.** *If edges are independent and Assumption 8.1 holds, then with a failure-aware matching policy the objective of Problem 8.1 is monotonic in the set of queried edges.*

While Propositions 8.1 and 8.2 state that single-stage edge selection is challenging in the worst case, our computational results suggest that these problems are often easier on realistic exchanges.

### 8.2.1 Using the Max-Weight Matching Policy as a Baseline

It might seem as though our edge pre-screening procedure is simply compensating for a flawed matching policy ($M^{\texttt{MAX}}(\cdot)$). If matching policy $M^{\texttt{MAX}}(\cdot)$ performs poorly in practice, then why not use $M^{\texttt{FA}}(\cdot)$? Indeed, $M^{\texttt{FA}}(\cdot)$ can directly account for edge failure, and Proposition 8.3 states that the edge selection problem is in fact "easy" when using this policy. However this assumes that the edge failure distribution is accurately known. The failure-aware matching policy $M^{\texttt{FA}}(\cdot)$ is very sensitive to the specified edge failure distribution, and if the assumed distribution is *incorrect* then this policy can perform very poorly. This is in fact a reason that $M^{\texttt{FA}}(\cdot)$ is not used in practice: edge failure distributions are not accurately known, and exchange programs are hesitant—with good reason—to guide their matching policy with a noisy estimation of edge failure.

In our edge pre-screening setting, the assumed edge distribution may also be noisy, however this does not directly affect the matching algorithm. In the worst case, an incorrect edge failure distribution will lead us to pre-screen edges that do

not provide useful information to the exchange (this is the status quo). This is in stark contrast to $M^{\mathtt{FA}}(\cdot)$: in the worst case, a bad estimation of the edge failure distribution will cause $M^{\mathtt{FA}}(\cdot)$ to match risky cycles and chains, potentially decreasing the number of transplants.

### 8.2.2 A Note on Match Run Frequency

In the real world kidney exchange is a *dynamic* process: patients are constantly entering and leaving the pool, and participants are matched every few weeks. One way to deal with edge failure uncertainty is to match *more frequently*. For example if we match patients and donors once every month, then each failed edge adds one month to the waiting time for every patient who relied on the failed transplant. If we match participants every few hours, this increase in waiting time is far less severe.

However there are good reasons to match *infrequently*. First, exchanges benefit with the addition of more patients and donors—new edges "thicken" the compatibility graph, enabling more cycles and chains. Second, each transplant center participating in exchange needs a transplant coordinator to manage individual patients and donors. Many small hospitals do not have a full-time staff for organ exchange, and they cannot deal with frequent match offers. For these reasons, most exchanges match patients and donors very infrequently: the UK national kidney exchange matches patients and donors every quarter[3] (once every three months); the Canadian national exchange matches participants once every four months[4]; UNOS currently matches patients once every week. [5]

[3] https://nhsbtdbe.blob.core.windows.net/umbraco-assets-corp/24443/pol274-4.pdf

[4] https://profedu.blood.ca/en/organs-and-tissues/programs-and-services/kidney-paired-donation-kpd-program

[5] https://unos.org/wp-content/uploads/unos/KPD_emanual_how-matching-works.pdf

## 8.3 Solving the Policy-Constrained Edge Query Problem

First we propose an exhaustive tree search which returns an optimal solution to Problem 8.1 given enough time. Building on this, we propose a Monte Carlo Tree Search algorithm and a simple greedy algorithm. Our multi-stage approaches are very similar to these, and can be found in Appendix C.4.

Our optimal exhaustive search uses a *search tree* where each tree node corresponds to an edge subset in $\boldsymbol{q} \in \mathcal{E}$. The children of node $\boldsymbol{q}$ correspond to any $\boldsymbol{q}' \in \mathcal{E}$ which are equivalent to the parent $\boldsymbol{q}$, but include one additional edge: $C(\boldsymbol{q}) \equiv \{(\boldsymbol{q} + \boldsymbol{q}') \; \forall \boldsymbol{q}' \in \mathcal{E} : |\boldsymbol{q}'| = 1 \;\mid\; (\boldsymbol{q} + \boldsymbol{q}') \in \mathcal{E}\}$. We say that edge sets (or tree nodes) containing $L$ edges are on the $L^{th}$ *level* of the tree. We refer to nodes with no children as *leaf nodes*. Unlike other tree search settings, the optimal solution to Problem 8.1 may be at *any* node of the tree, not only leaf nodes; this is a consequence of non-monotonicity (see Proposition 8.1). The tree defined by root node $\boldsymbol{q} = \mathbf{0}$ and child function $C(\boldsymbol{q})$ contains all legal edge subsets in $\mathcal{E}$, when $\mathcal{E}$ is a matroid. Thus, *any* exhaustive tree search algorithm (such as depth-first search) will identify an optimal solution, given enough time and memory.

Of course exhaustive search is only tractable if $\mathcal{E}$ is small. Consider the class of *budgeted* edge sets $\mathcal{E}(\Gamma)$ used in our experiments: $\mathcal{E}(\Gamma) \equiv \{\boldsymbol{q} \in \{0,1\}^{|E|} \;\mid\; |\boldsymbol{q}| \leq \Gamma\}$ (edge sets containing at most $\Gamma$ edges). The number of edge sets in $\mathcal{E}(\Gamma)$ grows roughly exponentially in $\Gamma$ and $|E|$, and is impossible to enumerate even for small graphs. Suppose a graph has 50 edges and we have an edge budget of five: there are over two million edge sets in $\mathcal{E}(5)$. Even small exchange graphs can have thousands of edges, and thus $\mathcal{E}(\Gamma)$ cannot be enumerated. Therefore, we propose search-based approach.

**Monte Carlo Tree Search for Edge Selection (`MCTS`):** We propose a tree-search algorithm for single-stage edge selection, `MCTS`, based on Monte Carlo Tree Search (MCTS), with the Upper Confidence for Trees (UCT) algorithm [186]. Our approach keeps track of a *value* (the objective value of Problem 8.1) and a UCB value estimate for each node, and these values are updated during sampling. The formula used to estimate a node's UCB value is

$$\frac{\frac{U}{N} - V^{min}}{V^{max} - V^{min}} + \sqrt{N^P / N}$$

where $U$ is the "UCB value estimate" calculated by `MCTS`, $N$ is the number of visits to the node, $N^P$ is the number of visits to the node's parent, and $V^{max}$ and $V^{min}$ are the largest and smallest node values encountered during search.

When the set of tree nodes is too large to enumerate UCT can use a huge amount of memory, by storing values for each visited node. To limit both memory use and runtime, we incrementally search the tree from a temporary root node. Beginning from the root (the the empty edge set), we use UCB sampling on the next $L$ levels of nodes—where $L$ is a small fixed integer. After a fixed time limit, sampling stops and we set the *new* root node to the current root's best child according to its UCB estimate—using the method of [186]. This process repeats until we reach the final level of the search tree. Algorithm 3 gives a pseudocode description of `MCTS`, which uses Algorithm 4 as a submethod. While often successful, MCTS requires extensive training and parameter tuning. As a simpler alternative, we propose a greedy algorithm.

**Single-Stage Greedy Algorithm (`Greedy`).** Like `MCTS`, our greedy algorithm begins with the empty edge set as the root node, and iteratively searches deeper levels of the

tree. However unlike `MCTS`, `Greedy` simply selects the child node with the greatest objective value in Problem 8.1–that is, *greedily* improving the objective value; see Appendix C.4 for a pseudocode description.

**Algorithm 3** `MCTS`: Tree Search for Single-Stage Edge Selection

**Require:** $\mathcal{E}$: legal edge sets, $K$: maximum size of any legal edge set, $T$: time limit per level, $L$: number of look-ahead levels

$\boldsymbol{q}^R \leftarrow \mathbf{0}$ root node (no edges)

$\boldsymbol{q}^* \leftarrow \mathbf{0}$ the best visited node

$V^* \leftarrow$ objective value of $\boldsymbol{q}^*$

**for** $N = 1, \ldots, K$ **do**

$M \leftarrow \min\{N + L, K\}$

$Q \leftarrow$ all nodes in levels $N$ to $M$

$U[\boldsymbol{q}] \leftarrow 0 \; \forall \boldsymbol{q} \in Q$ UCB value estimate

$V[\boldsymbol{q}] \leftarrow 0 \; \forall \boldsymbol{q} \in Q$ objective value

$N[\boldsymbol{q}] \leftarrow 0 \; \forall \boldsymbol{q} \in Q$ number of visits

**while** less than time $T$ has passed **do**

`Sample`$(\boldsymbol{q}^R, M)$

$\boldsymbol{q}^R \leftarrow \arg\max_{\boldsymbol{q} \in C(\boldsymbol{q}^R)} U[\boldsymbol{q}]$

Delete $U[\cdot]$, $V[\cdot]$, and $N[\cdot]$

**return** $\boldsymbol{q}^*$

**Algorithm 4** `Sample`: Sampling function used by `MCTS`

**Require:** $q$, $M$

$N[q] \leftarrow N[q] + 1$

$V[q] \leftarrow$ objective of edge set $q$ in Problem 8.1

**if** $V[q] > V^*$ **then**

$q^* \leftarrow q$, $V^* \leftarrow V[q]$

**if** $q$ has no children **then return** $V[q]$

**if** $q$ has children **then**

**if** $|q| < M$ **then**

$q' \leftarrow \arg\max_{q \in C(q^R)} U[q] + \text{UCB}[q]$

$U[q] \leftarrow U[q] +$ `Sample`$(q', M)$

**else**

$q' \leftarrow$ a random descendant of $q$ at any level

$V' \leftarrow$ objective value of $q'$ in Problem 8.1

**if** $V' > V^*$ **then**

$q^* \leftarrow q'$, $V^* \leftarrow V'$

$U[q] \leftarrow U[q] + V'$

**Runtime.** Our methods rely on an "oracle" to solve the NP-hard kidney exchange matching problem; while state-of-the-art methods solve real-sized instances of these problems in fractions of a second, there is no guaranteed bound for absolute runtime. Instead, we can report the *number of calls* to this oracle for each method as a measure of complexity. Both benchmark methods (max-weight matching and failure-aware [112]) as well as `IIAB` [56] use exactly one oracle call; i.e., they are $O(1)$. Both `Greedy` and `MCTS` use a fixed number of samples ($M$) to evaluate the objective of an edge set. `Greedy` evaluates the objective of an edge set exactly $\Gamma$ times; thus, `Greedy`

is $O(M \cdot \Gamma \cdot |E|)$. Finally, MCTS can in theory visit all potential edge sets of size at most $\Gamma$ (i.e., an exhaustive search), which is $O(M \cdot \sum_{\gamma=1}^{\Gamma} \binom{|E|}{\gamma})$. Since this version of MCTS is intractable in both runtime and memory, Algorithm 3 imposes reasonable limits on our implementation.

## 8.4 Computational Experiments

We conduct a series of computational experiments using both synthetic data, and real kidney exchange data from UNOS; all code for these experiments is available online.[6] In these experiments, "legal" edge sets are the budgeted edge sets defined as $\mathcal{E}(\Gamma) \equiv \{\boldsymbol{q} \in \{0,1\}^{|E|} \mid |\boldsymbol{q}| \leq \Gamma\}$. In Sections 8.4.2 and 8.4.3 we present results in the single- and multi-stage edge selection settings, respectively. We use both real data and synthetic data for our experiments.

### 8.4.1 Data

**Real Data.** We use exchange graphs from the United Network for Organ Sharing (UNOS), representing UNOS match runs between 2010 and 2019. Some of these exchange graphs only have the trivial matching (no cycles or chains), or they have only one non-trivial matching. We ignore these graphs because the matching policy is a "constant" function (to return the one feasible matching) and edge queries cannot change the outcome. Removing these, we are left with 324 UNOS exchange graphs.

These exchange graphs are relatively small and sparse: most graphs have fewer than 250 vertices, and fewer of half of these can be matched via a legal cycle or chain (with cycle cap $K = 3$ and chain cap $L = 4$); we refer to these vertices as *matchable*. The number of NDDs (who can initiate chains) is extremely small: most exchanges

[6] https://github.com/duncanmcelfresh/kpd-edge-query

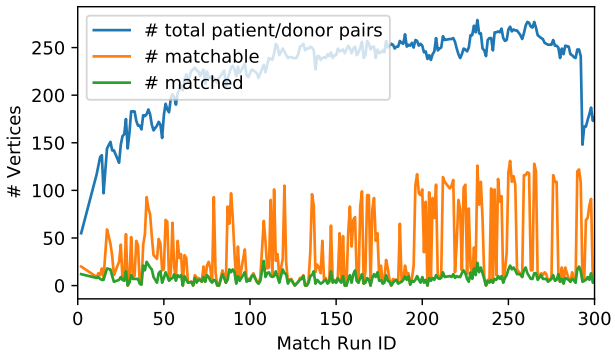



FIGURE 8.3: Number of patient donor pairs in each exchange, the number of matchable vertices (who can participate in a legal cycle or chain), and the number of vertices matched by $M^{\texttt{MAX}}(\cdot)$ in simulation.

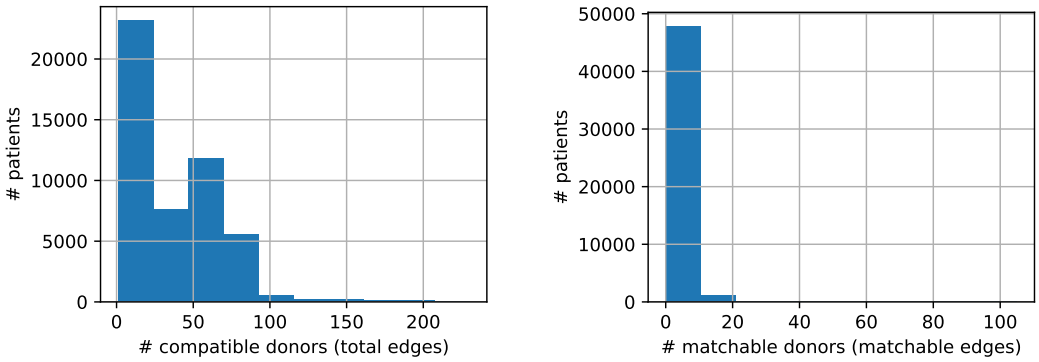



FIGURE 8.4: Left: Histogram of the number of *compatible* donors (edges) for each recipient, over all UNOS graphs. Right: Histogram of the number of *matchable* edges for each recipient, over all UNOS graphs.

have zero or 1 NDD. Figure 8.3 shows the number of vertices, the number of matchable vertices, and the number matched by $M^{\texttt{MAX}}(\cdot)$ (in simulation). These graphs are also sparse: Figure 8.4 (left) shows the number of edges (compatible donors) for each patient over all UNOS graphs. The median number of edges per patients is 29, and 14% have only one edge. Furthermore, while many patients have multiple *compatible* edges, very few of these edges are *matchable* (see Figure 8.4 (right)): 72% of all patients cannot be matched via a legal cycle or chain; of the *matchable* patients (with one or more matchable donor), the median number of matchable donors is 3.

**Synthetic Data.** We generate random kidney exchange graphs based on directed Erdős-Rényi graphs defined using parameters $N$ and $p$: let $V$ be a fixed set of $N$

vertices; for each pair of vertices $(V_1, V_2)$ there is an edge from $V_1$ to $V_2$ with probability $p$, and an edge from $V_2$ to $V_1$ with probability $p$ (independent of the edge from $V_1$ to $V_2$). Any vertices with no incoming edges are considered NDDs. In simulations we generate graphs which are smaller and more-sparse than the average UNOS graph ($N \leq 100$ and $p = 0.01$). These are meant to represent a toy-model of kidney exchange, rather than a realistic imitation of exchange graphs. Please see Appendix C.2 for results on these random graphs.

In these experiments edge rejections and failures are independently distributed for each edge $e$; let $P_R$ be the rejection probability, $P_Q$ is the post-match success probability if $e$ is queried/accepted, and $P_N$ is the success probability if $e$ is not queried. To simulate edge rejections and failures we use two synthetic edge distributions: *Simple* and *KPD*. In the *Simple* distribution, $P_R = 0.5$, $P_Q = 1$, and $P_N = 0.5$ for all edges. The *KPD* distribution is inspired by the fielded exchange setting from which we draw our real underlying compatibility graphs. According to UNOS, about 34% of all edges are rejected by a donor or recipient pre-match [198]; we draw $P_R$ uniformly from $U(0.25, 0.43)$ for each edge. Edges ending in highly-sensitized patients (who are often less healthy and more likely to be incompatible) are considered high-risk; for these edges we draw $P_Q$ from $U(0.2, 0.5)$ and $P_N$ from $U(0.0, 0.2)$. For other edges we draw $P_Q$ from $U(0.9, 1.0)$ and $P_N$ from $U(0.8, 0.9)$.

### 8.4.2 Single-Stage Edge Selection Experiments

In this section we compare against the baseline of a max-weight matching *without* edge queries (using policy $M^{\texttt{MAX}}(\cdot)$). Many fielded kidney exchanges use a variant of this matching policy, so by comparing against this baseline we are illustrating the impact of edge queries on the state-of-the-art matching policies used in many

real exchanges. Let $V_X$ be the objective[7] of Problem 8.1 achieved by method $X$, we calculate $\Delta^{\texttt{MAX}}$ (the relative difference from baseline) as $\Delta^{\texttt{MAX}} \equiv (V_X - V^S(\mathbf{0}))/V^S(\mathbf{0})$. A value of $\Delta^{\texttt{MAX}} = 0$ means that method $X$ did not improve over the baseline, a value of $\Delta^{\texttt{MAX}} = 1$ means that $X$ achieved an objective 100% greater than the baseline, and so on. Furthermore a value of $\Delta^{\texttt{MAX}} > 0$ means that method $X$ *increases* the objective by querying edges, while $\Delta^{\texttt{MAX}} < 0$ means that method $X$ decreases the objective by querying edges.

**Result: `Greedy` is essentially Optimal with small random graphs.** First we investigate the *difficulty* of edge selection. Using random graphs, we compare `Greedy` to the *optimal* solution to Problem 8.1, found by exhaustive search (`OPT`). We generate three sets of 100 random graphs with $N = 50$, 75, and 100 vertices, and each with $p = 0.01$. For all graphs we run both `OPT` and `Greedy` with edge budget 3; we calculate the *optimality gap* of `Greedy` as $\texttt{\%OPT} \equiv 100 \times (V_{\texttt{OPT}} - V_{\texttt{Greedy}})/V_{\texttt{OPT}}$, where $V_X$ denotes the objective achieved by method $X$. ($V_{\texttt{OPT}} > 0$ in all graphs used in these experiments.) If $\texttt{\%OPT} = 0$ then `Greedy` returns an optimal solution, and $\texttt{\%OPT} > 0$ means that `Greedy` is not optimal. Table 8.1 shows the number of random graphs binned by %`OPT`, as well as the maximum %`OPT` over all graphs. For each $N$, `Greedy` returns an optimal solution for at least 90 of the 100 graphs; the *maximum* %`OPT` over all graphs is 2.8.

In other words, `Greedy` always returns an *optimal* or nearly-optimal set of edges to query for small random graphs. This is somewhat unexpected, since the edge selection problem is both non-monotone and non-submodular (see Section 8.2).

[7] All objective values are estimated using up to 1000 sampled rejection scenarios (see Appendix C.1), as it is intractable to evaluate the exact objective of large edge sets.

[†] We use an approximation of Fail-Aware for the *KPD* distribution which assumes a uniform edge failure probability; *true* Fail-Aware should always have $\Delta^{\texttt{MAX}} > 0$.

TABLE 8.1: Optimality gap for `Greedy`, over 100 random graphs with $p = 0.01$ and various $N$, with edge budget $\Gamma = 3$; bottom row shows the maximum value of %`OPT` over all graphs.

| | Num. Graphs (out of 100) | | |
|---|---|---|---|
| %`OPT` | $N = 50$ | $N = 75$ | $N = 100$ |
| $[0, 0.1]$ | 93 | 93 | 90 |
| $(0, 1]$ | 5 | 4 | 9 |
| $(1, 2]$ | 1 | 3 | 1 |
| $(2, 100]$ | 1 | 0 | 0 |
| Max %`OPT` | 2.8 | 1.5 | 1.0 |

TABLE 8.2: Single-stage results on UNOS graphs using the variable IIAB edge budget (top rows), and the failure-aware method (bottom row). Columns $P_X$ indicates the $X^{th}$ percentile of $\Delta^{\texttt{MAX}}$ over all UNOS graphs.

| | *Simple* edge dist. | | | *KPD* edge dist. | | |
|---|---|---|---|---|---|---|
| **Method** | $P_{10}$ | $P_{50}$ | $P_{90}$ | $P_{10}$ | $P_{50}$ | $P_{90}$ |
| `MCTS` | 0.40 | 0.67 | 1.11 | 0.05 | 0.45 | 3.44 |
| `Greedy` | 0.47 | 0.64 | 1.00 | 0.02 | 0.47 | 3.44 |
| Random | 0.00 | 0.10 | 0.46 | −0.11 | 0.00 | 0.63 |
| `IIAB` | 0.21 | 0.45 | 0.89 | −0.27 | 0.12 | 2.24 |
| Fail-Aware | 0.00 | 0.09 | 0.23 | $-0.27^{\dagger}$ | $0.00^{\dagger}$ | $2.17^{\dagger}$ |

**Result: `Greedy` is essentially monotonic with UNOS graphs.** We test `Greedy` on real UNOS graphs, using maximum budget 100. Figure 8.5a shows the median $\Delta^{\texttt{MAX}}$ over all UNOS graphs, with shading between the $10^{th}$ and $90^{th}$ percentiles. Larger edge budgets almost never decrease the objective achieved by `Greedy`, and `Greedy` *never* produces a worse outcome than the baseline. Thus–in our setting–single-stage edge selection is effectively monotonic in our setting, and `Greedy` is an effective method.

**Result: `MCTS` and `Greedy` are nearly equivalent with UNOS graphs.** We compare all methods on UNOS graphs, using smaller, more-realistic edge budgets from 1 to 10. For `MCTS` we use a 1-hour time limit per edge ($\Gamma$ hours total). Figures 8.5b and 8.5d compare $\Delta^{\texttt{MAX}}$ for `MCTS`, `Greedy`, and random edge selection, for the *Simple* and *KPD* edge distributions, respectively. We draw two conclusions from these results: (1) `MCTS` and `Greedy` produce almost identical results, further suggesting that `Greedy` is nearly optimal in our setting; (2) in our setting, edge selection is *effectively* monotonic, as $\Delta^{\texttt{MAX}}$ almost never decreases. However Figure 8.5d gives an example of non-monotonicity for both `Greedy` and Random: in some cases, querying edges can lead to a *worse* outcome than querying no edges.

**Result: Both `MCTS` and `Greedy` outperform benchmarks from the literature.** We also compare against two state-of-the-art approaches: the edge selection approach of [56] (`IIAB`), which uses a *variable* edge budget that depends on the graph structure; and and the failure-aware matching policy of [112] (Fail-Aware[8]), which does not query edges To our knowledge, `IIAB` is the only edge selection method in the literature. We compare against the Fail-Aware method because it is a state-of-the-art

[8]For the *KPD* distribution we use an approximation of Fail-Aware, which assumes a uniform edge failure probability.

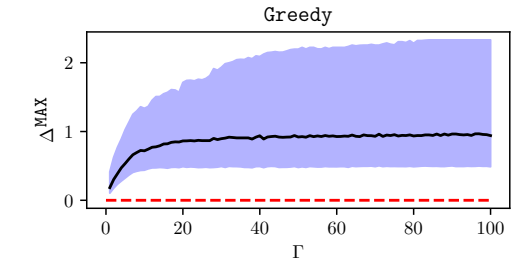
kidney exchange matching policy which aims to maximize the expected matching weight, under a similar edge failure model to ours; we compare against this approach to further illustrate the utility of querying edges.

Table 8.2 shows a comparison of all edge-selection methods–each using the variable edge budget of IIAB; the bottom row shows results for Fail-Aware. Both MCTS and Greedy achieve greater $\Delta^{\texttt{MAX}}$ (in distribution) than both benchmark methods. This is expected in both cases: IIAB uses a heuristic to select edges to query, which does not consider the final matching weight—the objective of our edge selection problem; on the other hand, both MCTS and Greedy are designed to maximize this objective. We do not expect Fail-Aware to out-perform any edge selection methods, since Fail-Aware does not have access to information revealed after edge queries.

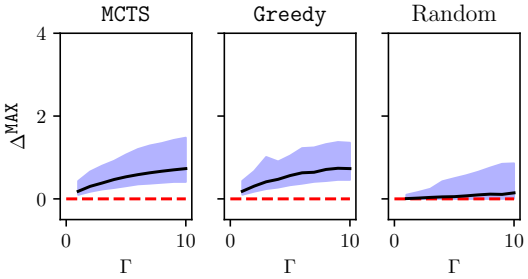
It is notable that Greedy performs better than MCTS (in distribution). This likely means that MCTS is *under-trained*—that the time and memory limits used in our implementation are too restrictive; alternatively, this indicates that Greedy is simply very effective in our setting.

### 8.4.3 Multi-Stage Edge Selection Experiments on UNOS Graphs

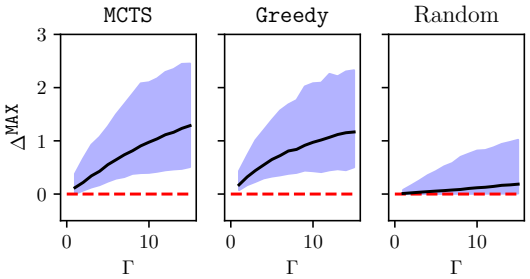
We run initial multi-stage edge selection experiments on all UNOS graphs with the *Simple* edge distribution. For each graph we test our multi-stage variants of MCTS and Greedy, and compare with a baseline of random edge selection; as before, MCTS uses a 1-hour training time per level. It is substantially harder to evaluate the multi-stage objective, as each edge edge-selection method changes depending on rejections observed in prior stages. Similarly, the MCTS search tree is orders of magnitude larger in the multi-stage setting: each node in tree corresponds to both an edge set *and* a rejection scenario (see Appendix C.4).



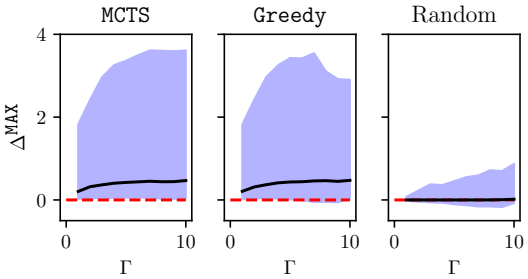

(A) Single-stage, *Simple* distribution.




(B) Single-stage, *Simple* distribution.




(C) Multi-stage, *Simple* distribution.




(D) Single-stage, *KPD* distribution.

FIGURE 8.5: Results for UNOS graphs. Right: edge budget up to 10 for the *Simple* distribution (top) and the *KPD* distribution (bottom). Top-left: `Greedy` with edge budget up to 100, for the simple distribution. Bottom-left: multi-stage methods using the *Simple* distribution. In all plots, a solid line indicates median $\Delta^{\texttt{MAX}}$ over all UNOS graphs, and shading is between the $10^{th}$ and $90^{th}$ percentiles; a dotted line indicates the baseline.

In these initial experiments we evaluate each method on 10 edge rejections *realizations* (only a small subset). We estimate $\Delta^{\texttt{MAX}}$ for each method and each graph by averaging the final matching weight over all realizations. Figure 8.5c shows the results of these experiments.

These initial multi-stage results are quite similar to our single-stage results. However it is notable that the objective value in the multi-stage setting is somewhat higher than in the single-stage setting–even using the simple method `Greedy`. Further, this suggests that more can be gained by developing a more sophisticated multi-stage edge selection policy. We leave this for future work.

## 8.5 Discussion

Many planned kidney exchange transplants *fail* for a variety of reasons; these failures greatly reduce the number of transplants that an exchange can facilitate, and increase the waiting time for many patients in need of a kidney. Avoiding transplant failures is a challenge, as exchanges are often constrained by policy and law in how they match patients and donors. We consider a setting where exchanges can *pre-screen* certain transplants, while still matching patients and donors using a fixed policy. We formalize a multi-stage optimization problem based on realistic assumptions about how transplants fail, and how exchanges match patients and donors; we emphasize that these important assumptions are not included in prior work. While this problem is challenging in theory, we show that it is much easier in practice–with computational experiments using both synthetic data and real data from the United Network for Organ Sharing. In experiments, we find that pre-screening even a small number of potential transplants (around 10) significantly increases the overall quality of the final match–by more than 100% of the original match weight.

Our initial study of the pre-screening problem suggests several areas for future work. First we assume that the distribution of transplant failures is known, when in reality only rough approximations of these distributions are available. Second, we assume that exchange participants (donors, recipients, hospitals) are not strategic. In reality, strategic behavior plays a substantial role in real exchanges [7]; we expect that participants might behave strategically when responding to pre-screening requests. Third, our model does not account for equitable treatment of different patients [215]. For example, it may be the case that pre-screening a transplant decreases the likelihood of the transplant being matched. That might disproportionately impact highly-sensitized patients, which are both sicker and more difficult to match than other patients.

## 8.6 Authors and Publication

This chapter was written by Duncan C McElfresh, Michael Curry, Tuomas Sandholm, and John P Dickerson, and it appeared at NeurIPS'20 [214]. Many thanks to Ruthanne Leishman, Sarah Booker, Morgan Stuart, and Darren Stewart for providing insight and motivation for this project.

# Chapter 9: Matching Algorithms for Blood Donation

## 9.1 Introduction

Blood is a scarce resource; its donation saves the lives of those in need. Countries approach blood donation in different ways, running the gamut from privately-run to state-run programs, with or without monetary compensation, and with varying degrees of public campaigns for action.[1] As such, blood donation rates differ across different countries; for example, approximately 3.2%, 1.5%, 0.8%, and 0.5% of the population donates in high-, upper-middle-, lower-middle-, and low-income countries, with varying rates of voluntary versus paid donors [319]. Yet demand for blood still far exceeds supply, and unmet need is greatest in low- and middle-income countries [263]. Thus, experts suggest that the blood supply chain—collection, testing, processing, storage, and distribution—be managed at a national level [263, 319].

Optimization-based approaches to blood supply chain management have a rich history in the operations research and health care management literature. [235] reviews over 100 publications in this space since 1963. The supply chain is roughly split into collection, testing & processing, storage & inventory, and distribution & transfusion [236]. Substantial research effort has gone into each of those segments [113, 120, 178, 251, 327]. Yet, we note that most optimization-based research in the initial

[1] Some examples follow. China maintains state control of its donation centers, which take a mix of captive-, quota-, and voluntary-based donations [152]. The US mixes state- and private-run donation that is primarily sourced via voluntary donations [235]. Brazil has seen a recent shift from remunerated to non-remunerated (aka voluntary) donation at its initially state-run, and now Federally-run, centers [73].

*collection* stage of the blood supply chain has focused on *prediction* of blood supply (e.g., during a crisis). In this work, we instead focus on the *creation* and *coordination* of blood supply via automated social prompts, subject to the expressed preferences and constraints of potential donors and the overall donation system. That is, we focus on the *donor recruitment* stage of the blood supply chain (see Figure 9.1).

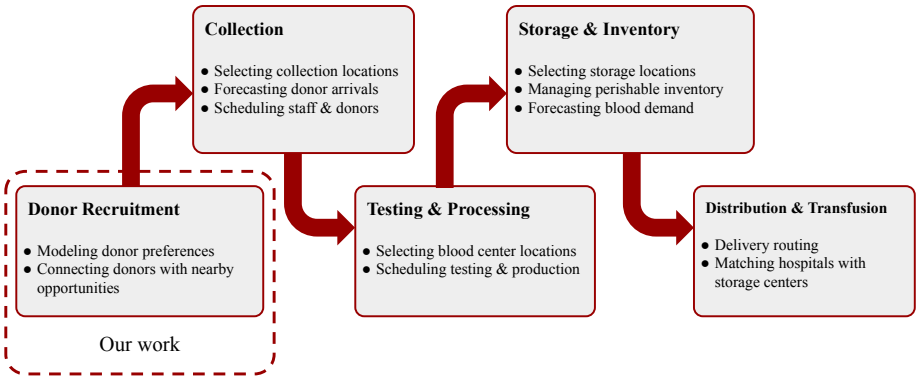
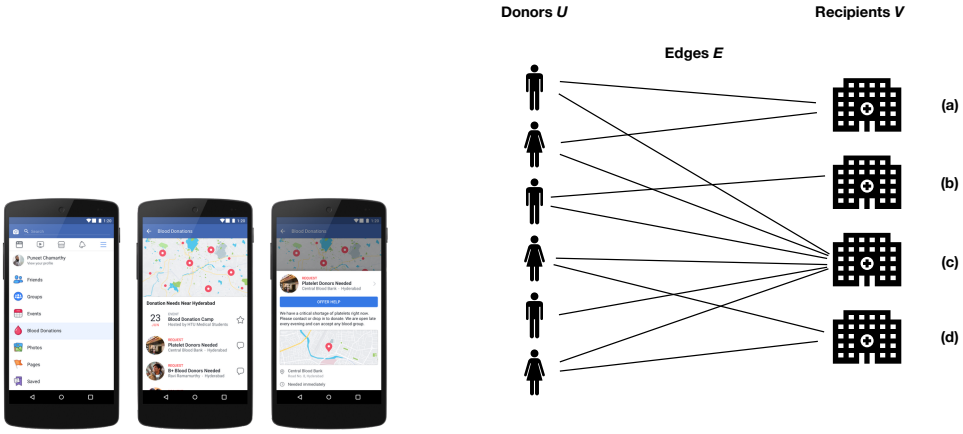



FIGURE 9.1: Stages of the blood supply chain. Our work—donor recruitment—precedes the four stages of the blood supply chain as described in [235].

Donor recruitment has also been a topic of study for decades. Factors like social pressure [289], empathetic messaging [260], and non-monetary incentives [78] can increase donation rates. Negative past experiences, and real or perceived barriers to donation, can also impede donation rates [93, 149, 309]. Most importantly, this body of work suggests that *different donors are motivated by different factors*. In other words, personalized recruitment strategies—which respect diverse donor motivations, preferences, and perceived barriers to donation—should be more effective than a uniform recruitment strategy.

Our work leverages the widespread use of web-based applications (apps) and social media platforms, which already play a substantial role in blood donor recruitment. The American Red Cross, which provides about 40% of transfused blood




(A) (B)

FIGURE 9.2: (a) The Facebook Blood Donation tool interface, where users can search for donation opportunities, and opt in to receive notifications about nearby opportunities as they arise. (Source: https://about.fb.com/news/2018/06/making-it-easier-to-donate-blood.) (b) an example matching graph, with donors (Facebook users who opt in to receive notifications about nearby opportunities), recipients (e.g., hospitals and blood banks), and edges (potential notifications that can be sent to donors).

in the United States,[2] recently launched an app to connect blood donors with donation opportunities.[3] A review by [237] identifies 169 free mobile apps for blood donation; though many of these apps have usability and privacy issues that may prevent widespread use. In a survey of donors at a German hospital, [297] finds that social media platforms Jodel and Facebook are a major motivation for donation—especially for first-time donors. Similar studies find that WhatsApp and Twitter help promote donation in Saudi Arabia [10] and India [3].

Herein we propose a personalized donor recruitment strategy using the recently developed Facebook Blood Donation tool,[4] which connects millions of potential blood donors with opportunities to donate, in several countries around the world.

[2] https://www.redcrossblood.org/donate-blood/how-to-donate/how-blood-donations-help/blood-needs-blood-supply.html

[3] https://www.redcrossblood.org/blood-donor-app.html

[4] https://socialgood.fb.com/health/blood-donations/

Users of this tool can opt in to receive notifications about nearby donation opportunities. Our strategy aims to notify donors about opportunities they are *more likely* to take action on. We frame this notification scenario as an online bipartite matching problem [176]—a well-studied paradigm which has been applied to a variety of settings including online advertising [221] and rideshare services [111, 206, 315]. We demonstrate, both in computational simulations and in a real A/B test, that even a simple matching policy can substantially increase the likelihood of donor action.

## 9.2 Online Platform: the Facebook Blood Donation Tool

The advent of global social networks offers a unique opportunity to recruit and coordinate massive numbers of donors, in order to meet a large and unpredictable demand for donor blood. The Facebook Blood Donation Tool aims to seize this opportunity—leveraging the widespread use of its online platform to connect blood donors with nearby recipients (see Figure 9.2a). Donors can also opt in to receive *notifications* about nearby donation opportunities. This tool is available in several countries around the world[5]; as of December 2020, more than 85 million people have registered with this tool.[6]

In this chapter we focus on a small but important feature of the Blood Donation Tool: automatic donor notifications. Our primary goal is to *increase the number of blood donations around the world* by carefully selecting *which opportunity* to notify each donor about, and *when* to notify them. We frame this question of donor notifications as an *online matching problem*. One might ask whether such a complicated approach is warranted in this setting—perhaps it does not matter how and when donors are

[5] As of February 2021, the Blood Donation Tool has been approved in Bangladesh, Brazil, Burkina Faso, Chad, Cote d'Ivoire, Egypt, England, Guinea, Hong Kong, India, Kenya, Mali, Mexico, Mongolia, Namibia, Netherlands, Niger, Northern Ireland, Pakistan, Peru, Rwanda, Senegal, South Africa, the United States, Taiwan, Wales and Zimbabwe (see `https://socialimpact.facebook.com/health/blood-donations/`).

[6] `https://socialimpact.facebook.com/health/blood-donations/`.

notified. To better motivate our approach, we first answer the question: how can we tell whether a Facebook user donates blood after we notify them?

### 9.2.1 Measuring Donation: Meaningful Action.

To design notifications that effectively encourage blood donation, it is necessary to know *when* donations occur. However social networking platforms like Facebook cannot directly observe a user's action outside the platform. As a proxy, we instead observe when a donor takes *meaningful action* toward donation after being notified. In our context, Meaningful Actions (MA) include user behaviors such as creating a reminder to donate, or calling a blood bank; note that these actions are only observed if taken within the Facebook platform.

It is beyond the scope of this study to validate MA as a proxy for actual donation, however initial results indicate that MA is a reliable indicator. For example, a 2018 Facebook study with its partner donation sites in India and Brazil found that 20% of donors said that Facebook influenced their decision to donate blood.[7] In the remainder of this chapter, we focus on increasing the number of donor MAs as a proxy for increasing the number of donations. Our goal is to design a notification policy that chooses both (a) *when* to notify a donor, and (b) *which donation opportunity* to notify them about. The next step in designing this policy is to understand *which* notifications are likely to prompt donor MA. We begin with some high-level observations.

As an initial analysis we consider all notifications sent to donors using the Facebook Blood Donation tool over a one-month period.[8] Below we describe some high-level observations; we leave a deeper analysis to future work.

[7] Ibid.
[8] Hundreds of millions of notifications.

1. **Users rarely take meaningful action in response to notifications:** between 3% and 4% of all notifications lead to meaningful action.

2. **More-engaged donors are more likely to take meaningful action:** Donors who tend to use Facebook every day are about 43% more likely to take meaningful action in response to a notification than those who use Facebook about once per week.

3. **New users are more likely to take action:** donors who joined Facebook within the last year are about 35% more likely to take action in response to a notification that those who have been users for longer.

4. **Older donors are more likely to take action:** donors over 30 years old are about 22% more likely to take action in response to a notification than donors under 30.

5. **Donors are more likely to take action if they are notified about a nearby opportunity:** Donors who are notified about opportunities less than 3km away are 20% more likely to take action than those who are notified about further-away opportunities.

6. **Donors are more likely to take action if they haven't been notified recently:** Donors who haven't been notified about a donation opportunity in the past 60 days are about 12% more likely to take action in response to a notification than those who have been notified in the past 60 days.

We emphasize that several of these observations have been reflected in prior studies: (1) reflects the observation of [289] and [297] that social pressure and influence from family or friends can increase donation rates. (5) reflects the finding of

[309] and [149] that logistical barriers to donation can impede donation rates. (6) reflects the finding of [325] that blood donors can be burdened by receiving too many notifications.

The likelihood of donor MA varies significantly across several features of both the blood donor (e.g., when they were last notified) and donation opportunity (e.g., location). To better understand these dependencies we train a predictive model for estimating likelihood of donor MA, using all available data from prior notifications. This model is used in both our offline and online experiments.

### 9.2.2 Machine Learning Model for Donor Action

To develop a machine learning (ML) model of donor action, we use all prior notifications sent by the Facebook Blood Donation tool. This model takes an individual notification as input, and predicts the *probability* that the donor will take action. Each notification is represented by a set of *features* of both the donor and the donation opportunity (i.e., the independent variables); the dependent variable is binary (i.e., whether or not the donor took MA). Before being deployed, this ML model and application passed through Facebook's internal review process to protect user privacy.

Prior to training this model, we use industry-standard feature selection techniques to identify the most important features for predicting donor MA; these features are (in decreasing order of importance, with importance percentage in parenthesis): (1) whether the donor recently took meaningful action (18%), (2) donor age (8.5%), (3) donor city (7.5%), (4) the number of Facebook friends the donor has (7.3%), (5) the distance between donor and recipient (6.8%). Other relevant features include the number of local donors (6.5%), number of times a donor has viewed the hub in the last 30 days, and the number of days since the donor's last notification.

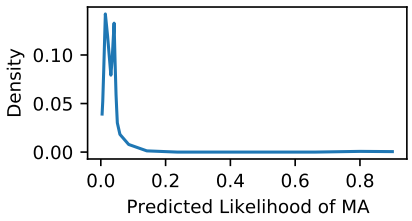



FIGURE 9.3: Density of estimated likelihood of MA, for all notifications in the training data.

Using the selected features, we train a gradient boosted decision tree (GBDT) model. We use standard parameter-sweep techniques to obtain the learning rate of 0.1, 120 trees, a maximum tree depth of 5 and a maximum number of leaves of 120. This model is trained using 10-fold cross-validation on 80% of the the training data and an additional 10% for validation; it achieves an AUC of 0.66 and logistic loss of 0.45, averaged over all training folds. Training this model is particularly challenging because of the small number of "positive" examples (i.e., the number of donor MAs). Figure 9.3 shows the density of prediction scores returned from this model, over all training data. Most prediction scores are between 0-10%, with an average of 3.43%—quite close to the observed likelihood of MA.

We use this model to estimate *how likely* it is that a donor will take action, when notified about a particular donation opportunity. Next we describe how this model is used to design a notification policy: by framing blood donor recruitment as a *matching* problem.

## 9.3 Matching Framework for Blood Donation

We represent a blood donation problem as a weighted bipartite *donation graph* $G = (U, V, E)$, with donors $u \in U$ and donation opportunities (or *recipients*) $v \in V$.[9]

[9]We use the terms "donors" and "recipients" as shorthand for *prospective* donors and recipients. Facebook does not make any determination about a person's eligibility to donate blood; these are potential donors who sign up to receive notifications of blood donation opportunities.

Each vertex has a set of *attributes* (e.g., blood type, geographical location, and so on), and these attributes determine whether a donor $u$ can donate to a recipient $v$—i.e., whether $u$ and $v$ are *compatible*. Compatible pairs $(u, v)$ are connected by edges $e = (u, v) \in E$; we denote all edges adjacent to vertices $u \in U$ ($v \in V$) as $E_{u:}$ ($E_{:v}$). All omitted proofs for the theoretical results in this section are given in Appendix D.3.

If an edge $e = (u, v)$ exists, then donor $u$ can be *notified* about $v$.[10] We discretize time into days $t \in \mathcal{T} \equiv \{1, \dots T\}$, with a finite-time horizon $T$. In our setting both donors an recipients are dynamic, in the sense that some donors and recipients are available at certain time steps. This notion of dynamism is designed specifically to represent a blood donation setting.

We assume that donors may receive only one notification at each time step, however *any number* of donors may be notified about the same recipient on any time step. Thus, our setting more-closely resembles $b$-matching [19] than traditional bipartite matching.

**Edge Weights:** Each edge $(u, v)$ has weight equal to the *probability* that donor $u$ donates to recipient $v$ once notified (i.e., the predicted MA likelihood, see §9.2.2); we assume that edge weights $w_{et}$ are indexed by edge $e$ and time step $t$. In other words, some edges (notifications) are more likely than others to result in donation: for example, certain people may be more likely than others to donate (e.g., people who have donated frequently in the past, as observed by [150]) and people may prefer to donate on specific days more than others.

**Recipients:** We consider both *static* recipients $S \subseteq V$, such as blood banks and hospitals, and *dynamic* recipients (or *events*) $D \subseteq V$, such as blood drives or emergency

[10] In this initial work, we assume the set of potential donors and donation centers do not change, although this *longer-term dynamism* is certainly interesting to consider as future research.

requests. Static recipients are available during *all* time steps, and edges into these recipients are always available. *Events* arrive in an online manner, and are available only during certain time steps. We assume that the *distribution* of recipient availability is known and defined by $p_{vt} \in [0, 1]$: the probability that recipient $v$ is available at time $t$. The distribution of recipient arrivals $p_{vt}$ is assumed to be known; this is a primary input to our matching algorithms. We use $\hat{p}_{vt}$ to denote a *realization* of recipient arrivals, which is 1 if donor $v$ is available at time $t$ and 0 otherwise. We assume that realized recipient arrivals $\hat{p}_{vt}$ are revealed on each time step $t$. In other words, at time step $t'$ all realized arrivals $\hat{p}_{vt}$ are known for time steps $t$ with $1 \leq t \leq t'$.

**Donors:** After a donor signs up with the Facebook Blood Donation Tool, we say they are *available* to receive notifications (i.e., to be matched) at any time. While there is essentially no limit on the number of notifications that can be sent on via online platform, there is a legal limit on how often people can donate blood. This limit is meant to protect donor health, and is often set by local governments or health authorities.[11] Thus, due to legal and health considerations, and out of respect for donors' time and attention, we limit how often each donor is notified: this limit is one notification every $K \in \mathbb{Z}_+$ days. Since not all notifications lead to donation, it is reasonable to set $K$ to 7 or 14 days—much shorter than the donation rate limit.

**Balancing Priorities:** In general there are several priorities when matching blood donors and recipients: we aim to increase the number of active blood donors, maximize the number of donations, respect donor privacy and preferences, satisfy recipients' needs, and so on. Deciding which of these policies is most important is a matter of policy, and is beyond the scope of this chapter. Here we consider two priorities which we believe are relevant to any blood donor matching platform: (a)

[11] Typically 8 weeks or longer; see https://www.redcrossblood.org/faq.html.

increasing the overall number of donations from a fixed donor pool, and (b) treating recipients equitably. While the justification for priority (a) is perhaps obvious, priority (b) requires more discussion.

### 9.3.1 Equitable Treatment of Recipients

In an online blood donor matching platform, notification policies have a far greater potential to impact recipients than donors. From a donor's perspective, a change in notification policy might mean that they receive notifications at a slightly different rate, or that they are encouraged to donate to a different recipient. (Recall that donors can always browse for opportunities using the Blood Donation tool; they need not pay attention to notifications.) However from a recipient's perspective, a change in notification policy can drastically impact the number of notifications encouraging donors to visit their facility. For example if predictive models suggest that edge weights to centrally-located hospitals are high, while edge weights to rural hospitals are near zero, then a simple edge-weight-maximizing policy would never notify donors about rural hospitals (indeed we report a similar distance-based effect in Section 9.5). Furthermore, two-sided matching platforms—such as the Facebook Blood Donation tool—are most effective when both sides of the market benefit from participating. If donors are never notified about rural recipients then these recipients might choose to leave the platform, which is a strictly worse outcome for everyone. For these reasons we consider the *fairness* of different notification policies.

Our approach is inspired by the problem of *fair division* in economics [296], and specifically the notion of weighted proportional fair division [94]. In weighted proportional fair division, a finite set of resources is divided among agents such that each agent values their allocation proportional to their *weight*—where greater weight represents greater endowment or priority. In our setting, different recipients have

different numbers of compatible donors (e.g., due to their location), or different edge weights (e.g., due to donor preferences or recipient accessibility); it may not be reasonable to, for example, guarantee that each recipient is matched with the same total edge weight. Instead we endeavor to match each recipient with edge weight proportional to their *normalization score*—where normalization scores are provided as input to the matching policy. Furthermore, since individual edges cannot be divided between recipients, it is not always possible to guarantee exact proportionality for all recipients. Instead we use a relaxed notion of proportionality, based on the normalized edge weight matched with each recipient.

**Definition 5** ($\gamma$-Proportional Matching)**.** Let $Y_v$ be the total weight matched with recipient $v$ over time horizon $\mathcal{T}$, and let $m_v$ be the normalization score for $v$. This matching is $\gamma$-proportional for $\gamma \in (0, 1]$ if

$$\gamma \frac{Y_{v'}}{m_{v'}} \leq \frac{Y_v}{m_v}$$

for each $v, v' \in V$.

In other words, a matching is $\gamma$-proportional if the normalized matched weight for recipient $v$ is at least fraction $\gamma \in (0, 1]$ of the normalized matched weight for all other recipients. Note that with $\gamma = 1$, all recipients receive the same normalized matched weight.

By this definition, it is always $\gamma$-proportional to allocate *zero* matched weight to all recipients (i.e., $Y_v = 0$ for all $v \in V$); we refer to this the *empty* allocation. We are interested in non-empty allocations; thus, one might wonder how hard it is to find *any* $\gamma$-proportional allocation which matches at least one edge. We refer to this as the $\gamma$-proportional allocation problem.

**Definition 6** ($\gamma$-Proportional Allocation Problem)**.** Input: $\gamma \in (0, 1]$, donation graph $G = (U, V, E)$, edge weights $w_e \in \mathbb{R}_+$ for each $e \in E$, and normalization scores $m_v \in \mathbb{R}_+$ for each $v \in V$. All recipient availability is known ahead of time. *Does there exist a non-empty set of edges in $E'$, with $E' \subseteq E$, which covers each donor at most once, and is $\gamma$-proportional to all recipients?*

**Theorem 9.1.** *The $\gamma$-proportional allocation problem is NP-hard for every $\gamma \in (0, 1]$.*

In other words, it is intractable to identify a $\gamma$-proportional allocation when recipient availability is known. Furthermore, recipient availability is often unknown: some recipients may host regular week-long blood drives, and others may only accept donation in response to patient needs. Instead we focus on proportionality in *expectation*—over all possible realizations of recipient availability.

## 9.4 Matching Policies

We aim to match donors with recipients such that we maximize edge weight (maximize the number of MAs), such that the outcome is $\gamma$-proportional for recipients. Here we define matching policies which trade off between both of these goals. These policies assume that donor availability is *fixed*, that is, we are given as input the time steps in which each donor can be notified. This is a natural constraint for fielded notification systems, which may only notify donors, for example, on certain days of the week. In Appendix D.4 we briefly discuss policies which also select *when* to notify each donor.

Each matching policy takes as input a bipartite graph $G = (U, V, E)$ with edge weights $w_{et}$, normalization scores $m_v$, recipient arrival distribution $p_{vt}$, and time horizon $\mathcal{T}$. At each time step $t$, all observed demand realizations $\hat{p}_{vt'}$ for all $t' \leq t$ are "revealed" to the policy, and may be used as input.

We use parameters $a_{ut}$ to denote the (exogenous) donor availability on each time step: donor $u$ may be matched on time step $t$ only if $a_{ut} = 1$. We denote the set of available edges for recipient $u$ on time $t$ by $E^t_{u:} \equiv \{(u', v') \in E \mid u' = u, a_{ut} = \hat{p}_{v't} = 1\}$.

In order to benchmark practical matching policies, we compare them with an unrealistic *offline optimal* policy, which has complete knowledge of the "true" demand realization $\hat{p}_{vt}$. The offline optimal policy is defined using any optimal solution to Problem 9.1.

$$\begin{aligned}
\max \quad & \sum_{t\in\mathcal{T}} \sum_{e\in E} w_{et} x_{et} \\
\text{s.t.} \quad & x_{et} \in \{0,1\} && \forall e \in E\ t \in \mathcal{T} \\
& s_v \in \mathbb{R} && \forall v \in V \\
& x_{et} \le \hat{p}_{vt} a_{ut} && \forall e = (u,v) \in E,\ t \in \mathcal{T} \\
& \sum_{e\in E^t_{u:}} x_{et} \le a_{ut} && \forall u \in U,\ t \in \mathcal{T} \\
& s_v = \tfrac{1}{m_v} \sum_{t\in\mathcal{T}} \sum_{e\in E^t_{:v}} x_{et} w_{et} && \forall v \in V \\
& \gamma s_v \le s_{v'} && \forall v, v' \in V,\ v \neq v'.
\end{aligned} \tag{9.1}$$

Here variables $x_{et}$ are 1 if edge is matched at time $t$ and 0 otherwise; auxiliary variables $s_v$ denote the normalized matched weight for recipient $v$. An offline optimal policy for this setting is defined using an optimal solution to Problem 9.1.

**Definition 7** (Offline Optimal Policy $\texttt{OPT}(\gamma)$)**.** Let $x^*_{et}$ be an optimal solution to Problem 9.1, for demand realization $\hat{p}_{et}$. At each time $t \in \mathcal{T}$, $\texttt{OPT}(\gamma)$ matches all edges $e \in E$ such that $x^*_{et} = 1$. Policy $\texttt{OPT}(0)$ refers to the offline-optimal matching policy without proportionality constraints.

**Corollary 9.1.1.** *It is NP-hard to identify policy $\texttt{OPT}(\gamma)$, for every $\gamma \in (0, 1]$.*

As a direct corollary of Theorem 9.1, Problem 9.1 is NP-hard for every $\gamma \in (0, 1]$. Thus, even if the demand realization is known, it is computationally hard to find an optimal matching. Of course, in realistic settings the demand realization is not known. Instead, our proposed policies use distributional information (exogenous parameters $p_{vt}$) to match donors and recipients. We compare these realistic policies to $\texttt{OPT}(\gamma)$ using two evaluation metrics:

**Competitive Ratio.** Let $E[\texttt{OPT}(0)]$ be the expected matched weight by $\texttt{OPT}(0)$, over all demand realizations. Let $E[\texttt{ALG}]$ be the expected matched weight by matching policy $\texttt{ALG}$, over all demand realizations and (if $\texttt{ALG}$ is stochastic) all policy realizations. The competitive ratio is

$$CR \equiv \min_{G,p,a} \frac{E[\texttt{ALG}]}{E[\texttt{OPT}(0)]},$$

where the minimization is over all possible matching graphs, demand distributions, and donor availability. In other words, $CR$ is the *worst-case* ratio of expected matching weight over all possible matching scenarios.

**Expected Proportionality.** Let $E[Y_v]$ be the expected weight matched by an a matching policy, over all demand realizations and (if $\texttt{ALG}$ is stochastic) all policy realizations. The expected proportionality of policy $\texttt{ALG}$ is

$$EP \equiv \min_{G,p} \max_{\gamma \in [0,1]} \{\gamma E[Y_v]/m_v \leq E[Y_{v'}]/m_{v'} \ \forall (v, v') \in V, \ v \neq v'\},$$

where as before $m_v$ is a fixed normalization score for recipient $v$, and the minimization is over all possible graphs, demand distributions, and donor availability. In other words, if policy $\texttt{ALG}$ is guaranteed to be $\gamma$-proportional in expectation then

$EP = \gamma$. Note that $EP$ may be 0, meaning that there is no $\gamma > 0$ such that the expected outcome is $\gamma$-proportional.

For the remainder of this section we assume that agent normalization scores are determined by a uniform random notification policy, defined below.

**Definition 8** (Uniform Random Policy `Rand` (fixed-time))**.** At each time step $t \in \mathcal{T}$, for each available donor $u$: `Rand` matches $u$ using an edge in $E^t_{u:}$ chosen uniformly at random.

**Definition 9** (Normalization Score $m_v$)**.** Let $E[Y_v]$ be the expected weight matched with recipient $v$, over all recipient demand realizations and (for randomized policies) over all policy realizations. The scaling factor for recipient $v$ is $m_v \equiv E[U_v]$.

Using these normalization scores we imply that policy `Rand`, and its outcome, are "fair"; we emphasize that this is only one choice of normalization scores, and in practice the notion of fairness/proportionality should be defined by stakeholders.

Metrics $CR$ and $EP$ help us characterize the expected performance of fixed-time matching algorithms. In the following two sections we analyze two classes of policies: *myopic* policies use only information from the current time step to make matching decisions (this includes both policies implemented in our online experiments); *non-myopic* policies take into account the demand distribution for future time steps.

**Myopic Policies** only take into account the information available at each time step. We consider two simple baseline myopic policies, `Max` and `Rand` (defined above). Policy `Max` is defined below.

**Definition 10** (Max-Weight Policy `Max`)**.** At each time step $t \in \mathcal{T}$, for each available donor $u$: let $W \equiv \max_{e \in E^t_{u:}} w_{et}$ be the maximum edge weight for any of $u$'s available

edges at time $t$. Max matches $u$ using any edge in $E^t_{u:}$ with edge weight $W$, and if multiple edges have weight $W$ then one is chosen randomly.

First, note that Rand has $EP = 1$ by definition. On the other hand, Max does not.

**Lemma 9.2.** *Max is $EP = 0$; that is, in the worst case Max is 0-proportional in expectation.*

Intuitively Max ignores normalization weights $m_v$, meaning that it does not guarantee proportionality. In the worst case, Max can leave some recipients can unmatched, meaning that $EP = 0$. On the other hand, Max *always* maximizes matched weight.

**Lemma 9.3.** *Max achieves competitive ratio $CR = 1$. Further, without proportionality constraints ($\gamma = 0$), Max is equivalent to an offline-optimal policy (OPT(0)).*

On the other hand, since Rand ignores edge weight, its worst-case competitive ratio is low.

**Lemma 9.4.** *Rand achieves a competitive ration of at most $CR = 1/N$ when there are $N$ recipients.*

Baseline policies Max and Rand represent two ends of a spectrum: on one side, Max prioritizes maximizing edge weight, at the cost of proportionality for recipients; on the other side, Rand treats all recipients "fairly" (for one specific notion of fairness), but does not prioritize edge weights. To balance these objectives in a principled way, we might randomly choose between Max and Rand at each time step, for each donor. This is the purpose of myopic policy RandMax, defined below.

**Definition 11** (Hybrid Policy RandMax($\gamma$))**.** At each time step $t \in \mathcal{T}$, and for each available donor $u \in U$, this policy randomly chooses to (a) match the donor using policy Max (with probability $1 - \gamma$), or (b) match the donor using policy Rand (with probability $\gamma$).

Since this policy randomly mixes `Max` (which is equivalent to an offline-optimal policy with $\gamma = 0$), and `Rand` (which is a "perfectly" proportional policy in this setting), this hybrid policy effectively balances the objectives of maximizing matched weight and proportionality for recipients.

**Lemma 9.5.** *`RandMax`$(\gamma)$ has $CR = 1 - \gamma$ and $EP = \gamma$, for all $\gamma \in [0, 1]$.*

In other words, this hybrid policy strikes a balance between matched weight ($CR$) and proportionality ($EP$), set by parameter $\gamma$. However this hybrid policy may not be Pareto optimal: for $\gamma \in (0, 1)$ there may be another policy with stronger guarantees on both proportionality $EP$ and competitive ration $CR$.

We leave the task of identifying a Pareto optimal policy to future work; instead we propose a class of stochastic policies with moderate guarantees on $CR$ and $EP$, though their performance is far better than these guarantees in computational experiments.

The policies introduced in this section are based on the optimal solution to an LP formulation of our matching problem. As a baseline for these policies we use an LP relaxation of the offline optimal MILP, Problem 9.1. We refer to this relaxation as Problem 9.1-LP (not stated explicitly). This problem is nearly identical to Problem 9.1, with two differences: (1) variables $x_{et}$ are continuous (on interval $[0, 1]$) rather than binary, and (2) demand realization $\hat{p}_{vt}$ is replaced by demand distribution $p_{vt}$.

Before defining matching policies based on Problem 9.1-LP, we make some important observations. First, Problem 9.1-LP yields a valid upper bound for Problem 9.1

**Lemma 9.6.** *Let $Z_{LP}$ denote the optimal objective of Problem 9.1-LP for a matching problem defined by $U, V, E, m_v, p_{vt}, \mathcal{T}$ and $\gamma \in [0, 1]$. Let $E[\texttt{OPT}(\gamma)]$ be the expected objective of the*

*offline-optimal policy, over all demand realizations. Then,* $Z_{LP} \geq E[\texttt{OPT}(\gamma)]$.

This result lets us use Problem 9.1-LP as an upper-bound on the matched weight for any matching policy; we use this as a baseline for which to compare other matching policies.

We consider two classes of LP-based policies: *non-adaptive* policies (which pre-commit to a set of edges that may be matched), and *adaptive* policies (which may change their matching decisions at each time step).

### 9.4.1 Non-adaptive Policies

We consider a class of non-adaptive policies which *pre-match* at most one edge for each donor at each time step—that is, matching decisions may not adapt at each time step as new information is revealed. At each time step, if the donor is pre-matched to an edge and the edge's recipient is available, then this edge is matched; otherwise the donor remains unmatched during this time step. Of course, this does not guarantee that all donors are matched at each time step—and therefore the competitive ratio can be quite low.

**Warm-Up: Policies based on Problem 9.1.** First we consider a non-adaptive policy based on an optimal solution for Problem 9.1-LP.

**Definition 12** ($\texttt{NAdapLP}(\alpha, \gamma)$)**.** Let $x^*_{et}$ denote an optimal solution to Problem 9.1-LP with proportionality parameter $\gamma \in [0,1]$ and $\alpha \geq 0$. For each time step $t \in \mathcal{T}$ and each donor $u \in U$, edge $e \in E_{u:}$ is pre-matched with probability $\alpha x^*_{et} / p_{vt}$, and the donor is not pre-matched with probability $1 - \alpha \sum_{e=(u,v) \in E_{u:}} x^*_{et} / p_{vt}$. At each time step, all donors are matched using their pre-matched edge, if the pre-matched donor is available.

In this policy, parameter $\alpha$ is a scaling factor used to ensure that each edge assignment distribution is *valid*—that is, that $\alpha \sum_{e=(u,v)\in E_{u:}} x^*_{et}/p_{vt} \leq 1$ for all $u \in U$. Note that this policy can only be implemented if each of these distributions are valid. Conveniently, the probability that any edge is matched by $\texttt{NAdapLP}(\alpha, \gamma)$ is expressible in terms of the optimal solution to Problem 9.1-LP used to define this policy.

**Lemma 9.7.** *Let $x^*_{et}$ be the optimal solution used in policy $\texttt{NAdapLP}(\alpha, \gamma)$. The unconditional probability that edge $e$ is matched at time $t$ by policy $\texttt{NAdapLP}(\alpha, \gamma)$ is $\alpha x^*_{et}$.*

Lemma 9.7 leads to some additional observations about this policy.

**Corollary 9.7.1.** *$\texttt{NAdapLP}(\alpha, \gamma)$ has competitive ratio $CR = \alpha$.*

**Corollary 9.7.2.** *$\texttt{NAdapLP}(\alpha, \gamma)$ is always $\gamma$-proportional in expectation, that is, $EP = \gamma$.*

Both corollaries follow directly from Lemma 9.7 and the constraints of Problem 9.1-LP. These results suggest that we can arbitrarily increase the weight matched by $\texttt{NAdapLP}(\alpha, \gamma)$ by increasing $\alpha$; however these policies are not guaranteed to be *valid*. This policy can only be implemented if $\alpha$ is small enough that each edge assignment distribution is valid.

**Lemma 9.8.** *Policy $\texttt{NAdapLP}(1/D, \gamma)$ is always valid and achieves a competitive ratio of $CR = 1/D$ and $EP = \gamma$ for all $\gamma \in [0, 1]$, where $D$ is the maximum degree of any donor: $D \equiv \max_{u\in U} |E_{u:}|$.*

In other words, Policy $\texttt{NAdapLP}(1/D, \gamma)$ is always implementable; thus there always exists a non-adaptive policy which achieves expected proportionality $EP = \gamma$ and competitive ratio $CR = 1/D$ for all $\gamma \in [0, 1]$. This competitive ratio guarantee is quite weak, and we might ask whether a better non-adaptive policy exists. Indeed it does, and we discuss this policy next.

**Optimal $\gamma$-Fair Non-Adaptive Policies** Here we aim to identify a policy which is $\gamma$-proportional in expectation ($EP = \gamma$), and also maximizes matched weight (and thus $CR$); we refer to this as an *optimal* $\gamma$-proportional non-adaptive policy. To identify this policy, we first observe that *any* non-adaptive policy can be characterized by the probability that it pre-matches edge $e$ at time $t$: $y_{et} \in [0,1]$; using these statistics, the unconditional probability that $e = (u,v)$ is matched at time $t$ is $y_{et}p_{vt}$. Note that for any non-adaptive policy, the probability that donor $u$ is pre-matched at time $t$ is at most 1 if $u$ is available and 0 otherwise; thus, statistics $y_{et}$ must satisfy conditions $\sum_{e \in E_{u:}} y_{et} \leq a_{ut}$ for all $u \in U$, and $t \in \mathcal{T}$. If a non-adaptive policy is $\gamma$-proportional, then $y_{et}$ must satisfy conditions

$$\gamma s_v \leq s_{v'} \quad \forall v, v' \in V$$

with

$$s_v = \frac{1}{m_v} \sum_{t \in \mathcal{T}} \sum_{e \in E^t_{:v}} y_{et} p_{vt} w_{et} \quad \forall v \in V.$$

Aggregating these conditions, we observe that the statistics $y_{et}$ of any $\gamma$-proportional non-adaptive policy is a feasible solution to the following LP.

$$\begin{aligned}
\max \quad & \sum_{t \in \mathcal{T}} \sum_{e \in E} w_{et} y_{et} p_{vt} & \\
\text{s.t.} \quad & y_{et} \in [0,1] & \forall e \in E \; t \in \mathcal{T} \\
& s_v \in \mathbb{R} & \forall v \in V \\
& \sum_{e \in E_{u:}} y_{et} \leq a_{ut} & \forall u \in U, \, t \in \mathcal{T} \\
& s_v = \tfrac{1}{m_v} \sum_{t \in \mathcal{T}} \sum_{e \in E^t_{:v}} y_{et} p_{vt} w_{et} & \forall v \in V \\
& \gamma s_v \leq s_{v'} & \forall v, v' \in V, \, v \neq v'.
\end{aligned} \tag{9.2}$$

Furthermore, a solution to Problem 9.2 corresponds to a non-adaptive policy; we use an optimal solution to this problem to define a $\gamma$-proportional non-adaptive policy.

**Definition 13** (`NAdapOpt`($\gamma$))**.** Let $y^*_{et}$ be an optimal solution to Problem 9.2. For each time step $t \in \mathcal{T}$ and each donor $u \in U$, a pre-matched edge is drawn with probability $y^*_{et}$; with probability $1 - \sum_{e \in E^t_{u:}} y^*_{et}$, no edge is pre-matched. At each time step $t$ and for each available donor $u$, if the donor is pre-matched with an available recipient, then the pre-matched edge is matched.

**Lemma 9.9.** *`NAdapOpt`($\gamma$) achieves expected proportionality $EP = \gamma$ and maximal competitive ratio over all non-adaptive policies, with $CR \geq 1/D$ (where $D$ is the maximum degree of any donor).*

Both non-adaptive policies described in this section are $\gamma$-proportional in expectation ($EP = \gamma$), thought their competitive ratio guarantee is somewhat weak. This is expected, since non-adaptive policies cannot change their matching decisions between time steps—they pre-match at most one edge for each donor at each time step. Some pre-matched edges will in fact be unavailable, depending on the particular demand realization (which is not known in advance).

### 9.4.2 Adaptive Policies

Adaptive policies can use any available information in order to make matching decisions—including observed demand realizations, prior matching decisions, and the distribution of future demand. We leave a general characterization of adaptive policies to future work; here we consider a simple class of adaptive policies that naturally extends the non-adaptive policies from the previous section. This policy class, `AdaptMatch`, takes as input the set of edges pre-matched by a non-adaptive policy, denoted by $M$, where $M_{ut} = e \in E$ if $u$ is pre-matched along edge $e$ at time $t$, and

$M_{ut} = \emptyset$ if $u$ remains unmatched at time $t$. `AdaptMatch` uses pre-matched edges when possible, and if a pre-matched edge is not available it matches donors using either `Rand` (with probability $\gamma$) or `Max` (with probability $1 - \gamma$). Algorithm 5 gives a pseudocode description of this matching algorithm.

**Algorithm 5** `AdaptMatch`: Adaptive matching policy

**Require:** donors $V$, recipients $U$, edges $E$, time steps $\mathcal{T}$, donor availability, pre-matched edges $M_{ut}$, parameter $\gamma \in [0, 1]$

**for** each time step $t \in \mathcal{T}$ **do**

  **for** each available donor, $u$ **do**

    **if** $u$ has a pre-matched edge $M_{ut}$, and this edge is available **then**

      Match $u$ using pre-matched edge $M_{ut}$

    **else**

      Flip a weighted coin with "heads" probability $\gamma$

      **if** heads **then**

        Match $u$ with policy `Rand`

      **else**

        Match $u$ with policy `Max`

**return** Matched edges for each time step

Note that this adaptive policy matches strictly more edges (in expectation) than their non-adaptive counterparts. Thus, expected matched weight (and $CR$) is strictly larger for `AdaptMatch` than the non-adaptive policy it is based on.

While competitive ratio is at least as large for these policies ($CR \geq 1/D$) as for their non-adaptive counterparts, there is no meaningful guarantee on expected proportionality. We leave more sophisticated adaptive policies to future work. However, while these approximate adaptive policies do not have strong guarantees on

*CR* or *EP*, they perform far better than these guarantees well in computational experiments (see § 9.5.1).

## 9.5 Results

Prior to deploying new matching policies in an online setting, it is important to assess their performance in simulations. Section 9.5.1 outlines computational simulations with real data from the Facebook Blood Donation Tool, using our proposed matching policies; Section 9.5.2 describes our online experiment with the Facebook blood donation tool. In Appendix D.1 we also present results using synthetic, publicly available data.

### 9.5.1 Computational Simulations

We developed open-source simulation code for these simulations, which implements each of our proposed policies; details of these simulations are discussed in Appendix D.1. All code used in these simulations is available in the supplementary material, and on GitHub.[12] Data related to the Facebook blood donation tool cannot be released due to concerns for user privacy. We test each matching policy from the previous section using data from the Facebook Blood Donation tool, and we ran separate simulations for 12 major cities around the world. For each city we create a blood donation graph, consisting of donors $V$ and recipients $U$ registered with the Blood Donation tool; edges are created between donors and recipients within 15km of each other, and edge weights are calculated by the GBTD models described in Section 9.2.2. Each of these cities has on the order of 1000 donors, 100 recipients, and 100,000 edges.

[12] https://github.com/duncanmcelfresh/blood-matching

We require that donors are notified exactly once every $K = 14$ days, and the first day each donor is notified is chosen randomly from $t \in \{1, \ldots, 13\}$; recipient availability parameter $p_{vt}$ are determined from past notifications. The realized recipient availability used in these experiments is randomly drawn using parameters $p_{vt}$, and this realization is fixed for the remainder of the experiment. Each simulation runs for 60 days, so each donor is notified exactly 4 times. Since policies `Rand` and `AdaptMatch` are random, we run 50 independent trials with these policies. We define recipient normalization scores $m_v$ as the average weight matched to $v$ over all 50 trials of `Rand`.

For policy `Max` we calculate the total matched weight, and for `Rand` and `AdaptMatch` we calculate the average matched weight over all trials. We also calculate the (average) weight matched to each recipient, $Y_v$. Using the recipient weights we calculate a measure of proportionality *Gamma*, defined as

$$Gamma \equiv \max\{\gamma \in [0,1] \mid \gamma Y_v / m_v \leq Y_{v'} / m_{v'} \forall v, v' \in V\}.$$

**Simulation Results** Simulation results for all 12 cities are shown in Figure 9.4. For each city we simulate matching using policies `Max`, `Rand`, and `AdaptMatch`. We implement several versions of `AdaptMatch`: each uses a fixed parameter $\gamma \in \{0.0, 0.1, \ldots, 1.0\}$, and pre-matched edges $M \equiv \texttt{NAdapOpt}(\gamma)$. These plots in Figure 9.4 illustrate the trade-off between overall matched weight and proportionality (or fairness) for recipients. While `Max` maximizes matched weight in this setting, it does not guarantee a proportional outcome: in all cities except for City 1 and City 9, *Gamma* is zero for `Max`, meaning that some recipients are never matched by this policy. On the other hand, `Rand` is proportional by definition (and $Gamma = 1$), though this policy does not maximize matched weight. However, `Rand` always matches at least 90% of the

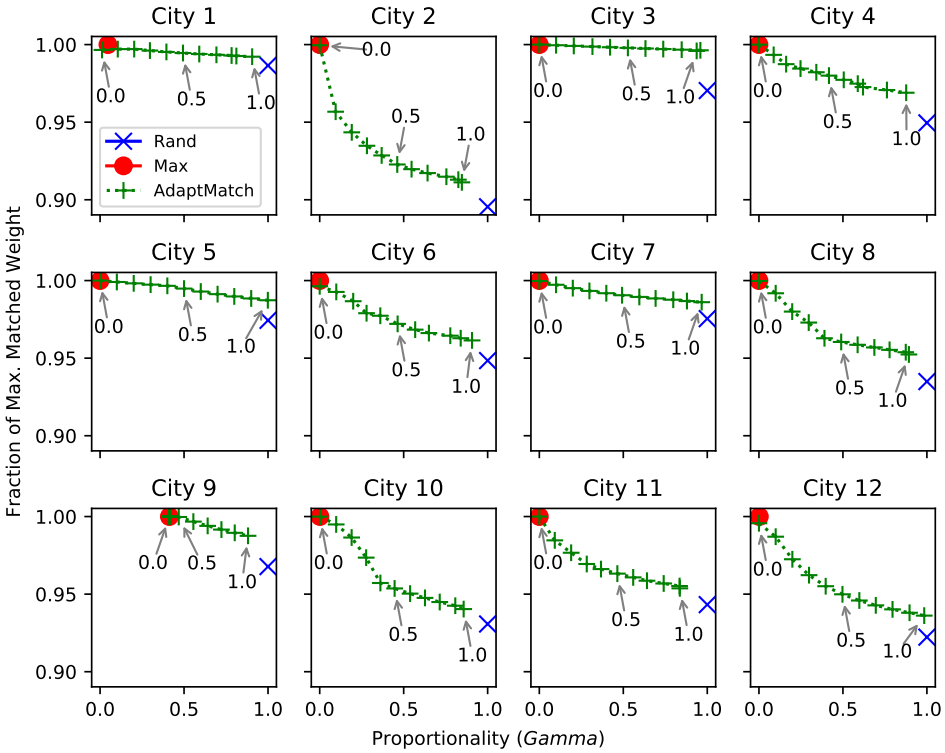



FIGURE 9.4: Simulation results for 12 cities around the world. Each plot corresponds to one 60-day trial in each city. The vertical axis shows the fraction of matched weight, compared to `Max`; the horizontal axis shows proportionality metric *Gamma*. Policy `Max` is shown as a red circle, `Rand` is a blue "×", and `AdaptMatch` is a green "+" (for $\gamma = 0.0, 0.1, \ldots, 1.0$. Arrows on each plot indicate the values of $\gamma$ used by `AdaptMatch`.

maximum possible matched weight in all simulations, and more than 95% in five out of the 12 cities.

While policy `AdaptMatch` does not have strong guarantees on matched weight or proportionality, it mediates smoothly between the extremes of `Rand` and `Max`, according to parameter $\gamma$. In some cases, this policy matches more weight than `Rand`, while still achieving a nearly-proportional outcome (*Gamma* equal to 1), as in Cities 3, 5, and 7.

### 9.5.2 Online Experiments

As a proof-of-concept, we compare the max-weight matching policy (`Max`) to the random baseline policy (`Rand`, which is similar in behavior to the notification policy currently used by the Facebook Blood Donation tool), in an online experiment. The goal of this experiment is to answer the question: *can we increase the overall number of donor meaningful actions* by carefully selecting *which recipient* to notify each donor about. Both of these policies notify donors once every 14 days; they only differ in *which recipient* each donor is notified about. `Rand` selects a nearby recipient at random, while `Max` selects a nearby recipient with the greatest likelihood of donor MA—according to our predictive model.

To compare these policies we design a randomized an online experiment, including hundreds of thousands of donors registered with the Facebook Blood Donation tool. We randomly partition these donors into a control group (who were notified using policy `Rand`) and a test group (who were notified using policy `Max`). As in our simulations, we include *only* static recipients (e.g., hospitals and large blood banks), who are always available to receive donations.

**Potential Impact on Donors and Recipients.** This experiment was approved by an internal review board. We emphasize that the impact of these experiments is minimal: the only difference between the test and control group in this experiment is *which* donation opportunity the donor is notified about. The impact on blood recipients is less clear: due to our experimental design we cannot effectively measure the proportionality of each notification policy in a meaningful way. However it is possible that any optimization-based matching policy (e.g., `Max` or `AdaptMatch`) prioritizes certain recipients over others. This may marginalize recipients in rural areas or those with a limited Facebook presence. More thorough analysis of these impacts is necessary before more widespread adoption of these policies.

**Online Experiment Results** This experiment ran from Nov. 23 to Dec. 17, 2019 (25 days); in total, 1,359,980 donors were notified using either policy `Rand` or `Max`. In this experiment many donors had only one compatible recipient—in this case, the donor was *always* notified about this recipient, regardless of the notification policy. For clarity, we distinguish between notifications sent to donors who had only one compatible recipient (1R), and those sent to donors with two or more compatible recipients (+2R). Thus we only expect to observe a difference between control and test groups for +2R notifications; we expect the same outcome for (1R) notifications. Table 9.1 shows the number of notifications and meaningful actions for notifications of each type (1R and +2R), in both the test and control group. Note that only +2R notifications are relevant for comparing the test and control groups, though we report both for transparency. The key result in these tables is the percentage of notifications that led to meaningful action (%MA, a number on $[0, 100]$). We report the Wilson score interval for %MA as $C \pm R/2$, where $[C - R/2, C + R/2]$ is the 95% confidence interval.

TABLE 9.1: Online Experiments - Number of notifications (#Notifs) and meaningful actions (#MA), over the online experiment. Notifications are separated into those sent to donors with only one compatible recipient (1R), and those sent to donors with two or more compatible recipients (+2R). Wilson score intervals are for the percentage of notifications that lead to MA are presented as $C \pm R/2$, where the 95% confidence interval is $[C - R/2, C + R/2]$.

| Notif. Group | Control (Rand) | | | Test (Max) | | |
|---|---|---|---|---|---|---|
| | #MA | #Notifs | %MA | #MA | #Notifs | %MA |
| 1R | 10,534 | 215,544 | $4.7 \pm 0.1$ | 10,755 | 214,841 | $4.8 \pm 0.1$ |
| +2R | 15,551 | 420,230 | $3.7 \pm 0.1$ | 16,054 | 412,387 | $3.9 \pm 0.1$ |

In the remaining discussion we consider only the +2R notifications, as there is no difference between the test and control group for 1R notifications. For the overall experiment, %MA is about 5% higher for `Max` than for `Rand`. To better understand the differences between the control and test groups, we use two statistical tests to compare the notifications sent by `Max` and `Rand`.

**Overall Comparison** We use both a two-sided and one-sided Chi-square test to compare %MA (+2R notifications only) for the control and test groups, over all notifications sent during this experiment. Let $P_{\texttt{Rand}}$ and $P_{\texttt{Max}}$ represent %MA for the control (`Rand`) and test (`Max`) groups, respectively. The two-sided test checks the null hypothesis **H0:** $P_{\texttt{Rand}} = P_{\texttt{Max}}$, with alternative $P_{\texttt{Rand}} \neq P_{\texttt{Max}}$; the one-sided test checks null hypothesis **H0:** $P_{\texttt{Rand}} = P_{\texttt{Max}}$, with alternative $P_{\texttt{Rand}} < P_{\texttt{Max}}$. We can reject *both* of these null hypotheses with $p \ll 0.01$. In light of the results presented in Table 9.1, these statistical test suggests `Max` achieves a small ($\sim$ 5%) but significant improvement over `Rand` in terms of overall %MA. In the next set of statistical tests we compare each *day* of the experiment as a separate trial.

**Daily Paired Comparison** Next we treat day of the experiment as a set of *paired measurements* of both $P_{\texttt{Rand}}$ and $P_{\texttt{Max}}$. For each day of the experiment (26 days in

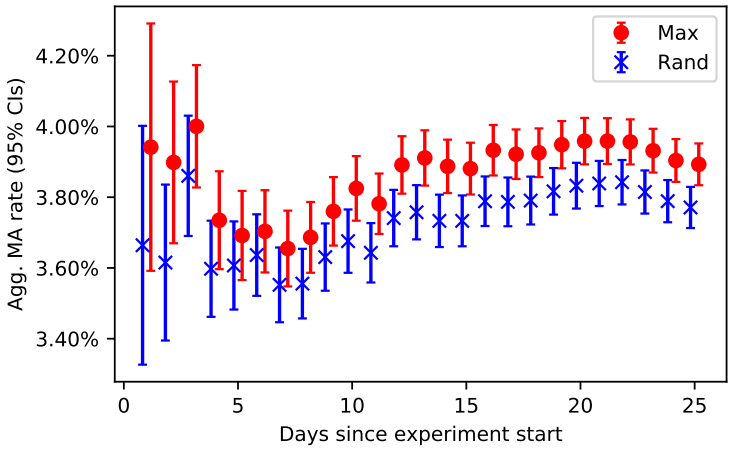



FIGURE 9.5: Aggregate MA rate for both `Rand` and `Max`, for each day in the experiment. Rates are calculated using the cumulative number of notifications and MAs at each day in the experiment. Error bars show the 95% confidence interval (Wilson score interval), and points indicate the center of the interval.

total) we calculate sample estimates of $P_{\texttt{Rand}}$ and $P_{\texttt{Max}}$—i.e., the 100 times the ratio of MAs to overall notifications. Note that donors are notified once every 14 days, meaning that the set of donors notified on any particular day is nearly disjoint from the donors notified on any other day of the experiment; for this reason we treat the measurements of $P_{\texttt{Rand}}$ and $P_{\texttt{Max}}$ on different days as independent.

We use a two-sided Wilcoxon signed-rank test to check the null hypothesis **H0:** the median difference between daily $P_{\texttt{Max}}$ and $P_{\texttt{Rand}}$ is zero. We reject this null hypothesis ($p \ll 0.01$), further confirming that notification policy `Max` yields a higher MA rate than `Rand`. For illustration, Figure 9.5 shows the 95% confidence intervals for $P_{\texttt{Rand}}$ and $P_{\texttt{Max}}$, using the aggregated number of notifications and MAs for each day of the experiment. In the Appendix D.2 we show the results for each individual day, as well as the cumulative rates.

## 9.6 Discussion

We introduce the problem of connecting blood donors with demand centers in a time-dependent setting, with uncertain demand. We formalize this as an online matching problem, with the priorities of *efficiency* (maximizing the number of donations) and *fairness* (proportionality) for recipients. We propose a class of stochastic policies for this setting, to which we compare a realistic randomized baseline. In simulations we see a clear trade-off between the overall number of donations and proportionality (Figure 9.4); the particular trade-off between these objectives depends on the notification policy used. Policy `Max` (which maximizes edge weight/expected donations) results in a 5-10% increase in the overall number of expected donations, compared to a random baseline (`Rand`). However `Max` tends to favor certain recipients over others. In our simulations, `Max` completely ignores some recipients in 11 out of the 12 cities tested—presumably because these recipients are associated with lower edge weights. On the other hand, `Rand` always sends a "fair" amount of notifications to each recipient, regardless of edge weight (according to the definition of fairness and proportionality used in this study). To mediate between the extremes of `Rand` and `Max`, we propose a class of stochastic policies (`AdaptMatch`); in simulations these policies effectively control the balance between the overall expected number of donations and proportionality across recipients, using parameter $\gamma$.

As a proof-of-concept we run an online experiment via the Facebook Blood Donation Tool, comparing notification policies `Rand` and `Max`. We find that `Max` results in about 5% more meaningful actions (a proxy for donations) than `Rand`. In relative terms this improvement seems small, however the implications are quite meaningful. This experiment investigated *one small improvement* to the notification strategy used by the Facebook Blood Donation Tool, i.e., whether the donor is notified about

a nearby donation opportunity at random (`Rand`), or notified about a particular opportunity selected by a predictive model (`Max`). Several other modifications to the notification policy might yield similar improvements: for example by changing *how often* each donor is notified, by more carefully planning for *future donation needs*, or by tailoring notifications to each donor's unique preferences and values.

The potential impact of this work is considerable. Indeed, if our observed results generalize to the entire community of Facebook blood donors, then a 5% increase in donor action corresponds to at about $160,000$[13] *more* donors taking meaningful action toward donation when notified. Even if few of these meaningful actions lead to actual donation, the increase is still substantial.

Before implementing these policies at a large scale in practice, it is important to understand their potential impacts on both blood donors and recipients. In this study impact on donors is minimal; the only difference between notification policies is *which donation opportunity* they are notified about. However our simulation results indicate that blood recipients may face significant impacts from changes in notification policy. For example policies that prioritize edges with a high likelihood of meaningful action (e.g., policy `Max`) may ignore certain recipients—such as rural hospitals or small donation centers with a limited web presence. This observation is particularly troubling if low-weight recipients are *already* unlikely to recruit donors, which we expect is the case. Of course, this potential injustice is exactly the motivation for our stochastic policy `AdaptMatch`.

Blood donation is a global challenge, and has been the focus of many dedicated organizations and researchers for decades. In this chapter we investigate a new

[13]Our results reported in Table 9.1 suggest that policy `Max` leads has a meaningful action rate of 3.9%, compared to 3.7% for policy `Rand`. The difference is 0.2%—or $160,000$ of the estimated 85 million donors registered with the Blood Donation Tool (https://socialimpact.facebook.com/health/blood-donations/).

opportunity to recruit and coordinate a massive network of blood donors and recipients, enabled by the widespread use of social networks. We formalize a matching problem around matching blood donors with recipients, and test these policies in both offline simulations and an online experiment using the Facebook Blood Donation Tool. Our findings suggest that a matching paradigm can significantly increase the overall number of donations, though it remains a challenge to do so while treating recipients equitably.

## 9.7 Authors and Publication

This chapter was written by Duncan C McElfresh, Christian Kroer, Sergey Pupyrev, Eric Sodomka, Karthik Abinav Sankararaman, Zack Chauvin, Neil Dexter, John P Dickerson; an earlier version of this chapter appeared at EC'20 [217].

# Part III

# Human-Algorithm Interactions

# Chapter 10: Toward Participatory Algorithm Design

Part III of this thesis steps away from the technical details of algorithm design, and turns toward questions of how people use and perceive of algorithms. I focus here on real, deployed algorithmic systems that help people make important decisions—such as finding the most efficient use of a scarce medical resource, screening job applications, or estimating the risk of criminal recidivism. In these applications there is broad agreement that deployed algorithms should satisfy certain ethical standards, however concrete definitions of these standards remain elusive. There is little consensus what ethical standards algorithms *ought to* meet, how compliance with these standards should be measured and enforced, and how algorithms should be designed to meet them.[1] There has been some progress on this front: over the past decade, the computer science community has highlighted *fairness*, *accountability*, *transparency*, *interpretability*, and *explainability* as especially important properties for deployed algorithms. New academic groups, conferences, and funding sources have emerged around these topics, which is perhaps a mark of progress. Yet there is no consensus on how these properties should be defined or implemented, and the literature is full of conflicting definitions. I believe that this research is currently undermined by its technical bias—focusing on the *algorithm* rather than the *stakeholders* it is intended for.

[1] One notable exception is *privacy*, which has been an active research area for decades [142].

Take for example the research domain currently referred to as eXplainable Artificial Intelligence (XAI)—which includes the study of machine learning interpretability and explanations. Presently, the XAI literature is largely based on *computer scientists'* beliefs of what "good" explanations look like, and this narrow focus has drawn criticism from both within [223] and without [20] the computer science community. Moreover, XAI research is largely focused on technical problems and technical solutions. For example, nearly all foundational machine learning explanation methods explain output from complex algorithms using output from *additional* complex algorithms [209, 261, 281, 298]. In these papers, explanations are deemed useful if they satisfy certain mathematical properties (symmetry, additivity, fidelity, and so on), which serve as abstractions for the needs of an actual stakeholder or user. Some recent methods incorporate user feedback—such as *personalized explanations* [278], the use of social transparency in XAI [119], and the context-specific explanations developed in Chapter 12 of this thesis. This is a step in the right direction, however I believe that even deeper stakeholder engagement is warranted.

Most human-focused AI and ML research, including XAI, centers the *algorithm* rather than the *stakeholder*. The stakeholder is often absent from the actual research process, save for a nod in the introduction or conclusion.[2] In order for our research to realize benefit, and to avoid harm, I believe that it is necessary to engage directly with people who use, and who are impacted by, our research—i.e., *stakeholders*. Since stakeholders are saddled with the benefits and burdens of AI and ML, they are best-equipped to make normative judgments about these systems: such as whether a system is behaving well or poorly, or whether it should be deployed at all. It may well be that the AI and ML literature aligns with stakeholder interests (we address this question in Chapter 14), however this cannot be assumed: in some cases AI and

[2]There are certainly exceptions [316, 321].

ML systems are demonstrably misaligned with stakeholder interests [234]. However it is not clear how to facilitate collaboration between stakeholders and computer scientists; I'll refer to this endeavor as *Participatory Algorithm Design* (PAD). The phrase *participatory algorithm design* was coined by [197] to describe their WeBuildAI framework for algorithm design; this is one of the only examples of participatory algorithm design in the computer science literature, and will serve as an inspiration for my future work.

As the phrase suggests, PAD is inspired by *Participatory Design* (PD): a practice that includes stakeholders directly in the innovation and development of new technologies. PD emerged in the late 20th century as a way for workers and labor unions to influence how computers would impact the workplace [57]; today PD falls under the umbrella of Human-Computer Interaction (HCI). There is a wide variety of PD methodologies for facilitating stakeholder interaction, and the PD literature has long grappled with the challenges of ethics and power that will inevitably arise in algorithm design [310]. PD would be a natural starting point for PAD research, however PD techniques are not commonly used by the applied AI and ML communities. These communities have instead developed a variety of stakeholder collaboration models, unrelated to PD, which are described either explicitly or implicitly in the literature. Most AI and ML literature assumes a "top-down" approach, where stakeholders determine system requirements, and technicians design algorithms to meet them; this is true of most XAI and ML fairness research. Alternatively, a "bottom-up" approach is guided by stakeholder input or by observing stakeholder behavior, for example through a survey. [131]. Hybrid approaches can involve both aspects of top-down and bottom-up design [11]. Yet another approach defines a division of labor between technologists and stakeholders—where each group has different responsibilities in the design process [201]. Formalizing the goals and methods of

PAD is an important next step; I discuss this in greater detail in Chapter 16.

Each of the following chapters addresses a different aspect of Participatory Algorithm Design: Chapter 11 investigates how (artificial) AI suggestions can impact real decisions; Chapter 12 explores a framework for user-driven explanations of ML models; Chapter 13 highlights the importance of *indecision* in human and algorithmic decision making; and Chapter 14 studies when people understand (and misunderstand) algorithmic definitions of ML fairness.

# Chapter 11: AI-influenced Decisions

## 11.1 Introduction

As AI systems are increasingly used to inform—and sometimes make—important decisions, it is increasingly important to understand how AI suggestions impact human decision making. We investigate this question using a hypothetical decision making scenario, where participants are shown an (artificial) AI-generated *prediction* of their future decisions; some participants are shown (artificial) predictions from a human expert, and a control group is shown no predictions. Our research questions are: Do people *follow* predictions made about their future decisions? And does this depend on whether the prediction came from an AI system or a human?

To address these questions we conducted three studies on the effect of "Artificial Artificial Intelligence" (AAI) assessment, in which random statements about users' values were (falsely) presented as AI-generated feedback. In each study, participants received an AAI assessment of their morality before they were presented with a series of hypothetical questions related to kidney allocation. We found that AAI assessments had an effect on participants' allocation choices between patients. Under some conditions, this effect was slightly altered if participants were first asked whether they *agreed* with the assessment. We also found differences between the effect on people who believed the assessment to be AI-generated, compared to those who believed that it was from human experts.

As our studies build on research across several disciplines including computer science, moral psychology, and social psychology, part of our effort is to suggest how relevant concepts can be reasonably translated across these frameworks. From the perspective of computer science, our question about AI assessment and moral decision making can be interpreted as determining the effect of intervening on preferences by introducing a prediction, which is presented to the user as an assessment in the sense described above. The question of how most effectively to learn preferences is the focus of *preference elicitation*, a field with broad applications in several fields including medicine [317], marketing [169], and auction design. [88] Preference elicitation is also a primary concern in *social choice* [21]—the study of how to aggregate preferences for collective decision making; social choice has received significant attention from computer scientists [63], raising further questions about automation and AI in human decision making.

However, to our knowledge, the effect of the *perception* of these predictions has not been covered in the computer science or preference elicitation literature. To study this, our AAI goes one step further: after making an assessment of a user, it presents this assessment *directly to the user*. Does this impact the user's moral decision making, interpreted as an expressed preference? This is an area in which research from psychology could be informative.

## 11.2 Methodology

We used a custom online platform to study the effect of AAI assessment on human decision making. Each decision took the form of a *pairwise comparison* [61]—a

decision format used widely in many disciplines, where an agent selects their most-preferred item from two options. Using a participant's answers to these comparisons, we learned an approximate *decision function* to describe their choices. In what follows, we briefly explain our methodology by describing the basic scenario we presented to participants, the decision function used for analysis, and the custom platform that simulates the decision making environment.

**Scenario: Moral Decisions on Kidney Allocation** In order to elicit preferences efficiently from our participants, we designed a simplified choice scenario with a small class of easily-measurable preferences, motivated by a real life-and-death decision made every day: allocating donor kidneys. Many of the features that the general population considers important in determining kidney allocation go beyond objective medical facts and enter into ethical opinions. In the US, these decisions are guided strictly by UNOS policy[1]. However, the general population may think other features to be relevant to these decisions. For example, people are often unwilling to allocate organs to patients with features that do not contribute to organ failure or prognosis [303]. This makes kidney allocation a valuable avenue for comparing attitudes toward different ethically relevant characteristics, and for studying differences between informal attitudes and lay opinions about what should be included in formal policies.

We use *life expectancy* (henceforth, "LifeExp") and *number of dependents* (henceforth, "Dep") as the two criteria for comparing patients in need of a kidney; both of these features have been demonstrated to be of importance to the general population for ethically relevant but seemingly distinct reasons [131]. We also included patient

[1] https://optn.transplant.hrsa.gov/media/1200/optn_policies.pdf

age, which is 40 years for all patients. In these studies we presented each participant with several pairwise comparisons, where the alternatives are two patients in need of a single kidney; this is akin to the problem of allocating a single deceased-donor kidney to one of two patients. In each study, these features were explained to participants as follows:

**Life Expectancy** How many years the patient is expected to live if they receive the kidney transplant, if the patient makes no lifestyle changes.

**Dependents** The number of children under the age of 18 for whom the patient is responsible for providing at least half the necessary support, including food, shelter, and clothing.

**Age** The current age of the patient. All patients in the scenarios are 40 years old. This feature does not vary.

Because LifeExp and Dep convey different types of value on a donor kidney, those two features may be varied independently without either implying anything about the other. We held age constant at 40 years to limit further the assumptions participants could make about the patients from the target features.

**Measuring Participant Decision Functions: Feature Dominance** Each participant may have arbitrarily complicated preferences in this setting. One participant may only allocate kidneys to patients with LifeExp greater than 10 years, and choose randomly otherwise. Another participant might only care about a different feature (such as the patient's age) and completely ignore both LifeExp and Dep. To avoid this problem we constructed a set of pairwise comparisons that essentially ask which *feature* the participant cares most about.

In each comparison, one patient *always* had greater LifeExp and less Dep than the other patient. We assume that participants answer each comparison by selecting the patient with either greater LifeExp or Dep. Formally, we assume that each participant has a simple *decision model*: with probability $p$ they prefer the patient with greater life expectancy, and with probability $1 - p$ they prefer the patient with more dependents. For ease of exposition, we express these probabilities as percentages, where

$$\%\text{Life} \equiv 100 \times p$$

Because participants only considered a patient who was always strictly greater in life expectancy and fewer in dependents to another, this is simply *the percentage of comparisons where the participant selects the patient with greater LifeExp* :

$$\%\text{Life} \equiv 100 \times \frac{\text{\# of LifeExp Favoring Decisions}}{\text{\# of Total Decisions}}$$

To measure the impact of an AAI assessment on participant decision functions, we first learned %Life for each participant. We then compared the effect AAI assessments had on each intervention group by aggregating the participants' %Life from respective groups.

**Custom Online Platform** We created a custom online platform to facilitate data collection, in a style similar to The Moral Machine Project.[2] The core component of the online platform is the sequential display of a set of a hypothetical decision making scenarios in which participants choose one of two patients to receive a donor kidney. In each study, every participant received the same set of scenarios, but the

[2] http://moralmachine.mit.edu/

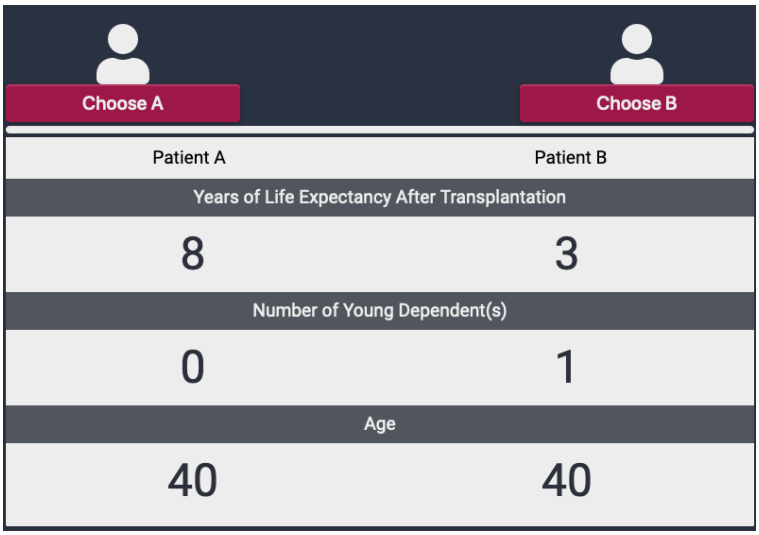



FIGURE 11.1: Decision making screen

display of each scenario was randomized: each scenario appeared in a different order for each participant, and each hypothetical recipient in a scenario was randomly selected to be presented on either the left or the right side. Participants were given a chance to review their answer, and the option to change their minds as needed.

For each study below, participants were first given a brief description of the decision making scenario (kidney allocation). They were informed that if one patient received the kidney, the other would *not* receive one, and that if a patient did not receive the kidney they were expected to live less than a year. Furthermore, it was made explicit that all transplants were likely to be successful. Figure 11.1 shows an example screen shot of our platform.

From pilot testing, we expected that participants recruited online would maintain attention on the kidney allocation task for between 20 and 30 patient profile pairs. As such, all studies asked participants to respond to 20 pairs to ensure they remained focused. Further, pilot testing suggested that decision making time decreased substantially after the first three pairs, indicating that participants took about three decisions to become familiar with the task. Therefore, all studies used at least 10 pairs to ensure that most of the participants' decisions were made after becoming

familiar with the task.

## 11.3 Study 1

### Method

114 participants were recruited on Amazon Mechanical Turk (MTurk) in a single cohort (on a Monday afternoon); only United States residents were used. After data collection was completed, 17 participants were excluded from analyses for failing an attention check which required them to report the assessment they received. Six participants were removed due to participants with the same IP address making multiple attempts. This leaves a final sample of N=91 (41% self-reported females and 59% self-reported males; mean age=37.7, SD=11.2, 76% white).

Participants were presented with background information on kidney allocation and about the patient features in this survey. On our online platform, participants were asked to make decisions on a set of 10 scenarios. To limit decision complexity, we further simplified the scenarios by keeping *all but one patient feature* the same for each comparison. One patient always had a life expectancy of 20 years and 0 dependents, while the other patient had 4 dependents. The only variable feature was the life expectancy of the latter patient, ranging from 1 to 19 years. After these 10 decisions, an "assessment" screen was displayed with the intervention text. Participants were randomly assigned to view one of the following AAI assessments:

**LifeFavor** "According to our AI model, you care a lot about the life expectancy of the patients when making decisions about who will get a kidney."

**DepFavor** "According to our AI model, you care a lot about how many dependents patients have when making decisions about who will get a kidney."

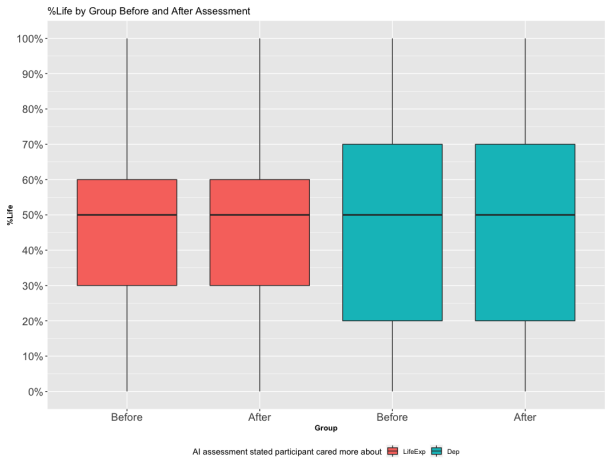



FIGURE 11.2: Study 1: medians and first/third quartiles for %Life, before and after assessment for each participant

Immediately afterwards, participants were prompted to make 10 more decisions. While the comparisons were the same as the initial 10, participants were not explicitly informed of this. Furthermore, the sequence in which the queries were shown was shuffled, and the sides on which patient profiles in each comparison were displayed were randomly switched. After completing all kidney patient allocation choices, participants responded to a survey, which included demographic information and a question on whether they agreed with the AAI's assessment. They were then debriefed, which included telling them that the feedback was actually random.

## Results

After exclusions, 32 participants received the LifeExp intervention and 59 received the DepFavor intervention. %Life was created as a summary variable to capture the proportion of life expectancy-favoring decisions for each participant. Figure 11.2 displays the visualization of the result.

We use a one-sided Wilcoxon rank sum test to determine if the %Life for participants in each group moved toward their respective assessment after it was given.

For the LifeExp group we test if the median %Life for the identical set of comparison was *higher* after assessment than before it; we cannot reject the null hypothesis ($p = 0.3$). For the DepFavor group we test whether median %Life was lower after the assessment (i.e., participants favored patients with more dependents); we cannot reject the null hypothesis here either ($p = 0.5$).

**Discussion** Surprisingly, both pre-assessment and post-assessment groups are nearly identical: two important observations came from this exploratory study. First, in the process of generating the initial 10 decisions as the input for the "AI," participants may have formed explicit decision rules that made them resistant to intervention. This would have been simple to do, since one patient always had 20 years of life expectancy with no dependents in every comparison. Second, the lack of the effect of DepFavor compared to LifeFavor could be attributed to the fact to that Dep was invariant across patients, while one patient's LifeExp varied between comparisons.

## 11.4 Study 2

In this study we modified the design of Study 1 to amplify potential effects of the intervention. First, we changed the position of the intervention relative to the allocation decisions. This change is motivated by two behavioral observations: (1) choice tendencies that are constructed through repetition are more resistant to change than choice tendencies developed through contextual cues [14]; (2) presenting self-referential information before a task tends to cause behaviors consistent with that self-referential information [238]. To avoid pre-assessment heuristic development, and to promote the AAI assessment as self-relevant information, we presented the AAI intervention *before* the kidney allocation task. We also asked participants to answer questions on

which the AAI's assessment could plausibly be based; these questions are described in the following section.

Second, we added two conditions: in one condition participants are asked if they agree with their assessment *before* the decision task, and in the other condition they are asked *after* their decision task. In Study 1, participants could develop an opinion about their assessment while engaged in the allocation task, and report their end agreement or disagreement based on their experience. We hypothesize that judging the accuracy of the assessment before the task would amplify either incorporation of the assessment into participants' self-referential beliefs (if they agreed), or reactions in opposition to the assessment (if they disagreed).

### Method

350 participants were recruited on MTurk in two cohorts (one around midday on a Monday, one during afternoon on a Wednesday), although they were randomly assigned among all groups within each cohort. As in Study 1, only US residents were used. 11 participants failed either or both attention checks, 7 participants could not accurately report assessment received, and 11 failed an attention check by choosing to allocate a kidney to a patient with what we considered dominated features (0 dependents, 1 year life expectancy, versus 0 dependents, 20 years life expectancy). 11 participants failed to finish the task, and six attempted the task twice. Some participants belonged to more than one of the categories above. The final sample after exclusions was therefore N=322 (41% female and 58% males; mean age=35.8, SD=10.2, 79% white).

As in Study 1, participants first received background information about the task and patient features. They were also told that they would answer a series of questions that an AI agent would use to make an assessment about what they found most

important in the kidney allocation task.

Participants rated, on a 5-point Likert-type scale, the extent to which they agreed or disagreed with 14 statements about the importance of using example features to determine who should receive a kidney (e.g., *"I feel that race is important in determining which patient should receive a kidney"*). These example features *did not* include life expectancy, number of dependents, or age. Afterwards, participants were randomly assigned to one of five conditions, in which all but control received AAI assessments that used contrastive language (e.g., *"you care more about the life expectancy of the patients than how many dependents they have"*):

- **Control:** no assessment ($N = 65$)
- **LifeFavor :** participants "assessed" by AAI to prioritize life expectancy over number of dependents ($N = 66$)
- **LifeFavorQ :** participants "assessed" by AAI to prioritize life expectancy over number of dependents, and asked immediately upon viewing the assessment if they agreed with it ($N = 60$)
- **DepFavor :** participants "assessed" by AAI to prioritize number of dependents over life expectancy ($N = 64$)
- **DepFavorQ :** participants "assessed" by AAI to prioritize number of dependents over life expectancy, and asked immediately upon viewing the assessment if they agreed with it ($N = 67$)

Following intervention, all participants responded to 20 curated patient comparisons in random order. As before, each comparison had one patient with greater Dep and one with greater LifeExp; unlike Study 1, both LifeExp and Dep varied by

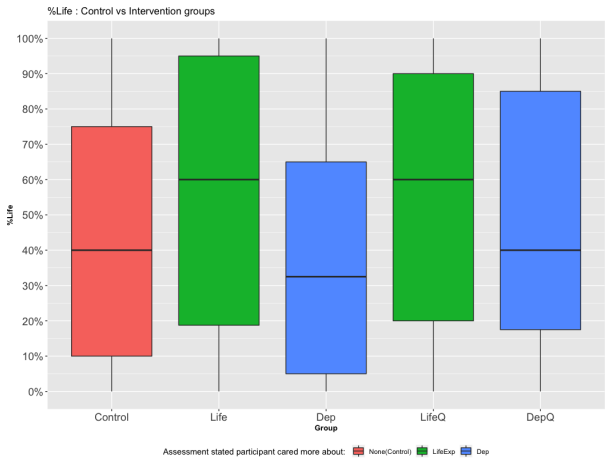



FIGURE 11.3: Study 2: medians and first/third quartiles for %Life of participants in each group

patient and pair. Participants then completed the same post-task survey as in Study 1, and were debriefed as in Study 1.

## Results

As before we calculate %Life as the percentage of comparisons in which they decided to allocate the kidney to the patient with higher life expectancy. Figure 11.3 shows a box plot of %Life for each group, aggregated over all comparisons.

Intervention groups LifeFavor (M=56, SD=10) and LifeFavorQ (M=56, SD=10) both had greater mean %Life than Control (M=45, SD=13). Also consistent with our main hypothesis, DepFavor (M=40, SD=10) had lower mean %Life than Control. These results are suggestive of the hypothesis that an AAI assessment influences people to make decisions aligned with the assessment. The exception, however, is that DepFavorQ (M=49,SD=13) had a *higher* %Life than Control.

We used the one-sided Wilcoxon rank sum test to compare %Life for each participant group. Compared to the control group, we have some reasonable evidence suggesting that median %Life was higher for LifeFavor ($p = 0.056$) and LifeFavorQ ($p = 0.057$). However, we saw no evidence suggesting that the median %Life

are lower for DepFavor ($p = 0.307$) and DepFavorQ ($p = 0.85$) than for Control.

**Discussion** Overall, we may reasonably suggest that the modification of experimental conditions from Study 1 to Study 2 enhanced the effect of AAI assessments, as evidenced by the moderate directional results. In Study 1, the LifeExp assessment had virtually no impact on decision making, but, in Study 2, participants in groups that received a LifeFavor and LifeFavorQ assessment favored patients somewhat more heavily on the basis of LifeExp than control. This could be due to participants receiving their assessments *before* having the chance to develop decision preferences, but also could be due to the increased complexity of the comparisons. The result is not overwhelming, however: in comparison, there seems to be no evidence that DepFavor assessment had an impact on participants' decision making. Even more curiously is the high $p$-value (0.85) from the comparison between DepFavorQ %Life and Control %Life.

## 11.5 Study 3

Our third study compares the influence of AAI assessments to that of supposedly human assessments: we examine the effect of AAI assessments relative to assessments believed to be generated by *human experts*.

### Method

450 participants were recruited on MTurk in two cohorts (one between a Thursday evening and a Friday around midday, and one around midday on a Saturday), all participants were randomly assigned to an experiment group; as in Studies 1 and 2, only US residents were included. Exclusion procedures were similar to procedures

in Study 2, except here we also checked if they could report the source of the assessment. 59 participants were excluded because they could not accurately report the assessment received or its source, 22 were excluded because of failed attention checks or failure to complete the task. The final sample after exclusions was therefore N=369 (43% female, 56% male, and 1% other/not indicated; mean age=38, SD=11.3, 73% white).

The study design and artificial assessments were similar to in Study 2. Each participant was randomly assigned to five groups:

- **Control:** no assessment ($N = 77$)
- **LifeFavorAI:** participants were given an AAI assessment stating that they care more about life expectancy than number of dependents ($N = 80$)
- **DepFavorAI:** participants were given an AAI assessment stating that they care more about number of dependents than life expectancy ($N = 75$)
- **LifeFavorPsy:** participants were informed that, based on a test made by "expert psychologists," they care more about life expectancy than number of dependents ($N = 74$)
- **DepFavorPsy:** participants were informed that, based on a test made by "expert psychologists," they care more about the number of dependents than life expectancy ($N = 63$)

After the assessment, participants were told that their responses to comparisons would be used either to train an AI that models their decision making, or by expert psychologists to develop a psychological test. The post-task survey was similar to the post-task survey used in Studies 1 and 2, as was the debriefing information.

### Results

As in Study 2, we calculated %Life for each participant over all comparisons. Both groups that received LifeExp assessments—LifeFavorAI (M=45,SD=11) and LifeFavorPsy (M=60, SD=11)—had higher %Life than Control (M=40,SD=12). Inconsistent with Study 2, however, was that the %Life were also higher for the groups that received dependent-favoring assessments—DepFavorAI (M=42, SD=11) and DepFavorPsy (M=42, SD=11). Figure 11.4 shows a box-plot of %Life for subjects in each group.

We use a two-sided Wilcoxon rank sum test to determine whether assessments from AAI had a difference in effect from assessments perceived to be from human experts. We reject the null hypothesis for life expectancy, but not for dependents: the median %Life for LifeFavorAI was not equal to LifeFavorPsy ($p = 0.01$), but there is no significant difference between DepFavorAI and DepFavorPsy ($p = 0.93$).

Next we use the one-sided Wilcoxon rank sum test for comparison between intervention groups and the control: (1) LifeFavorPsy has a statistically significant higher %Life than Control ($p < 0.001$), (2) %Life for LifeFavorAI is not significantly greater than for the control ($p = 0.15$). We found no significant difference between DepFavorPsy ($p = 0.75$) and DepFavorAI ($p = 0.67$) and control.

### 11.5.1 Discussion

We found strong evidence that, for the more impactful of our two patient features (LifeExp), there was a difference between the perception of "AI" and "expert human" assessments. Although life expectancy-favoring AAI assessments influenced participants to make allocation decisions in the expected direction (as they did in Study 2) compared to control, participants responded more strongly to the "expert"

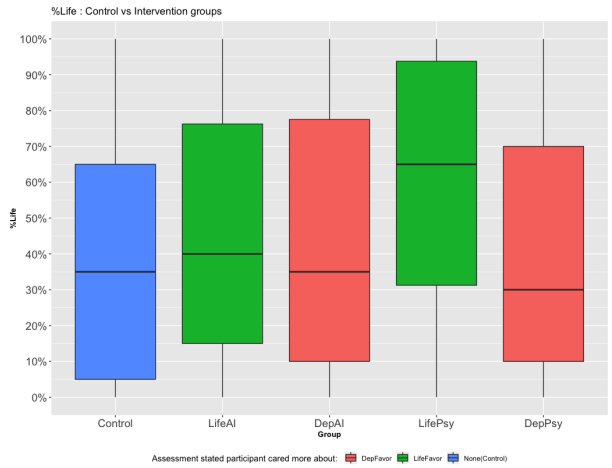



FIGURE 11.4: Study 3: medians and first/third quartiles for %Life for over all comparisons

life expectancy-favoring assessment by this metric than to any other condition. Furthermore, the results of the LifeFavorAI and LifeFavorPsy groups were significantly different from each other—suggesting that there is a difference between the perception of "AI" and "expert human" assessments. Dependent-favoring assessments did not notably impact decision making, so we cannot draw conclusions about these interventions.

## 11.6 Discussion

Overall, we derive three central findings from these studies: First, life expectancy-favoring assessments had modest directional results on participants' decision making. Second, dependent-favoring assessments had little notable effect. Third, the largest effect was that of the life expectancy favoring-assessment from "expert psychologists," not that of any AAI assessment.

It is important to keep in mind that the decision scenarios used in this study are highly stylized; more accurate, genuine AI assessments might have a stronger effects than artificial assessments. Future research should compare the effects of an artificial AI assessments with those from a "true" AI or ML system. Other modifications will

also likely impact participant responses to these assessments—such as the wording used, whether they align with participants' self perception, how much information is provided and when in the decision process the assessments are presented; we observed some of these effects in Study 1 and Study 2.

It is possible that participants were less receptive to the dependent-favoring assessment than the life expectancy-favoring assessment because it was a less complex variable and therefore easier to form an opinion about: each patient could have only up to four dependents, whereas they could have up to twenty years' life expectancy. Comparisons of different patient features could also yield different results.

The stronger influence of assessment from an "expert psychologists" compared to the AAI assessment in the life expectancy-favoring condition could be interpreted in several ways. As evidence-supported feedback is more effective in belief revision than unsupported feedback [262], participants might have believed that evaluation about decision making by a psychologist was better evidence of their preferences than an evaluation by AI. Alternatively, participants might have assigned more credibility to "expert psychologists" than to an "AI" simply because of the word "expert."

## 11.7 Authors and Publication

This chapter was written by Lok Chan, Kenzie, Doyle, Duncan C McElfresh, Vincent Conitzer, John P Dickerson, Jana Schaich Borg, and Walter Sinnott-Armstrong; it appeared at AIES'20 [76].

# Chapter 12: Learning Useful Explanations

## 12.1 Introduction

In the previous chapter we explored how (artificial) AI output influences peoples' behavior. In this chapter instead study how people *interpret* the output of ML models—and specifically, output from post-hoc explanation methods.

Researchers have proposed a variety of methods to explain the output of ML models, and a variety of properties to characterize their performance. Common properties of explanation methods include *fidelity* (how well the explanations match the underlying model), *stability* or *consistency* (whether similar inputs or similar models result in similar explanations), *comprehensibility* (whether the explanations are understood by an average person), and *computational complexity*, among others [69, 153, 290, 323].

These properties are important, but they cannot indicate whether the resulting explanations are *useful* to the user. Furthermore, whether or not an explanation is *useful* varies across users and decision making environments. We refer to the combination of user, the ML model, and the decision making environment as *context*. For example, an engineer debugging a marketing model might use complex explanations to understand aberrant behavior—this is one context. A policy analyst might interrogate the same marketing model to determine if it discriminates against certain customers—this is another context.

Context is defined variously in the computer science community, but largely focuses on *information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves* [4, 66, 101, 102, 271]. Our definition fits this general definition of context.

Prior work has considered aspects of "good" ML explanations from a philosophical perspective [207], and through the lens of performance metrics [172]; another related concept is *personalized explanations* [278]. However, most prior work focuses on *fixed* explanation methods for a particular context; we consider a setting where the context and user are not specified ahead of time. In this setting, the "best" explanation method is likely to vary across contexts. We propose a framework (Figure 12.5) for *learning* the most useful explanation(s) for each context through context-specific user feedback.

### Our Contributions

- (Section 12.2) We conduct an online survey study where participants are asked to complete a "downstream task" with a ML model, in a computer vision setting; the downstream task is to detect "bugs" in the ML model. We aim to answer the question *can users identify ML explanations that will help them complete their task?* In the first part of this survey each participant is shown several example input-output pairs for the model, along with visualizations of several common explanation methods. For each example, participants are asked to rank these explanations according to how useful they are for a downstream task. Participants are also asked which explanation method they found most useful for this task (their "preferred" method). Then, each participant is asked to complete this downstream task for a series of new inputs, using a single

explanation method; some participants are shown their preferred explanation method, and a control group is shown a non-preferred method. To vary context, we use two different classification settings, and two different types of bugs.

- (Section 12.3) We find that participants rate the *usefulness* of explanation methods differently in different contexts—this is reflected both in participants' usefulness ratings, and in the method they select as most-useful. We also find that participant usefulness ratings are *not correlated* with their ability to complete the downstream task. This suggests that following user preferences is not necessarily an effective way to develop useful explanations.

- (Section 12.4) We propose a theoretical framework for context-aware explanations, guided by feedback from a user completing a downstream task. We emphasize that the quality of a model explanation is necessarily tied to a downstream "task," such as debugging an object detector for a self-driving car, or a classifier trained to diagnose an illness. Accordingly, our explanations should strive to help users accomplish this task.

- (Section 12.4.1): As a proof-of-concept for our framework, we simulate an automated ML explanation system using the data from our survey study.

### Related Work

**Algorithmic Explainable AI (XAI)** Many methods have been proposed by the ML community that generate post-hoc explanations of black box-models [191, 209, 261] and complicated models such as deep neural networks [181, 282, 285, 287, 294]. Another common method for explaining predictions of a ML model is to have models that are interpretable by design, so that one can examine the output of a model by

simply inspecting the model itself [12, 74, 79, 190, 199]. There are also methods which provide explanations that can serve as recourse for individuals receiving the explanations [308, 312] as well as methods that impose an "interpretability prior" during the training phase of a ML model [188, 266]. Since all of these methods differ significantly in the kind of explanation they provide, many measures have been proposed to evaluate explanations. These include fidelity, computational complexity, comprehensibility, sensitivity, among others [69, 290, 323]. It has often been argued that explainability evaluation should inherently be human-centered [116]. The goal of this chapter is not to design a new explanation method or a new evaluation metric. We also do not aim to change the model. Instead, we aim to build on the vast body of prior work to propose a framework that can be used to assess the type of explanation(s) that will be most useful in a given decision making *context*.

**Human-Centered XAI** Many recent studies have taken a human-centered approach to evaluating explainability methods [173, 179, 189, 250]. However, these studies do not vary the context in the evaluation of an explanation method. We take a different stance and argue that usefulness of an explanation changes based on the context in which it's being used. This is inspired by Miller's work which argues that research in explainable AI should build on findings in social sciences which have studied how humans explain their decisions to other humans [224]. One key idea formalized by Miller is that explanations for AI systems are highly contextualized. We build on this and propose a framework that returns context-aware explanations, by eliciting user preferences using feedback mechanisms. Additionally, Arya et al. [22] acknowledge that different scenarios require different kinds of explanations and propose a taxonomy of questions that can help people choose an appropriate explainability method. However, our work differs from such a taxonomy in that we automatically infer which explainability method is most useful in a given context

from task-specific feedback. Moreover, our results show that it's often hard for users to know exactly the kind of explanation is needed to accomplish some downstream task, thus showing the flexibility of our framework over a pre-defined taxonomy. Many current XAI methods have been criticized for being too focused on algorithmic solutions, without taking into account the stakeholders that will receive the explanation [224]. As a result recent works have focused on characterizing stakeholders of XAI methods [299] and expanding the epistemic boundaries of XAI to also incorporate the social context in which these methods are situated [118]. Our work adds to the scholarship on broadening the scope of XAI research to include the context in which an explanation is needed.

## 12.2 Empirical Study: Users Cannot Identify Helpful ML Explanations

The goal of this study is to determine whether non-expert users of an ML model can identify model *explanations* that are helpful for them in a downstream task that involves the ML model. Since there is a wide range of applications for deployed ML models, and model users, we expect that the most useful explanations will depend on the particular context. We aim to test the following hypotheses:

**H1:** The usefulness of an explanation method varies across users and contexts.

**H2:** Users are better able to complete downstream tasks that depend on a ML model if they are provided with explanations they see as useful.

### 12.2.1 Study Design

This study was approved by our institution's ethical review board, and a full transcript of our survey can be found in Appendix E.3. Survey participants agreed to an

online consent form, which was approved by our ethical review board; this consent form is also available in Appendix E.3. Some personally identifiable information was collected during this survey through the online platform, and it is not used in our analysis. All code used in this study is available on GitHub. [1] All participants were recruited via Amazon Mechanical Turk, and were compensated US$3 for their time, with the opportunity for a small bonus (described below).

**Survey Scenario: Self-Driving Cars and ML Bugs.** To test our hypotheses we use four online surveys, each with a different decision scenario. Each survey centers on a hypothetical ML application, where a self-driving car uses a ML model to classify (250x250-pixel) images. To simulate a downstream task, participants are told that the ML model is sometimes "buggy", and that their task is to identify whether or not there is a bug. To encourage careful consideration of this task, participants who complete this task with more than 60% accuracy are given a US$2 bonus (total compensation US$5). To identify bugs, participants are shown a model input (an image), output (class prediction), and a model explanation (a masked image). Figure 12.1 shows an example input-output pair and ML model explanations.

**Explanation Methods.** Four different explanation methods are used throughout all surveys; these methods are commonly used in the ML literature, and we use publicly-available implementations of them for our experiments. In this study we aim to focus on user perceptions and behavior rather than specific explanation methods; for this reason we refer to these methods as ("A", "B", "C", and "D") in both the survey and our analysis. Complete descriptions of these explanations can be found in Appendix E.2.

[1] https://github.com/duncanmcelfresh/learning-explanations

Each explanation method is a masked version of the target image, where *unmasked* regions are "important" to the positive class label; Figure 12.1 shows an example of each explanation method.

**Survey Design.** Each survey consists of two parts. In Part I participants are shown eight examples, each consisting of an input-output pair, all four explanations of the output (A, B, C, D), and whether or not this model is buggy—i.e., "ground truth" for the downstream task. Half (4) of these examples are buggy, and half are not. For each example participants are asked how useful each explanation is at determining whether or not there is a bug, using a 5-point scale (5 = Extremely useful, 1 = Not at all useful). At the end of Part I, participants are asked which explanation method (A, B, C, D) they found most useful for detecting bugs.

In Part II, participants are asked to *identify* bugs for 10 additional examples using an input, output, and a single model explanation; each participant sees the same explanation method for all 10 examples. In Part II, we use a between-groups design to test **H2** with two groups, `Control` and `Test`. Participants in group `Control` are shown explanations from a randomly-chosen method that they did *not* select as most-useful. In other words, if a `Control` participant chose method B at the end of Part I, they are shown either method A, C, or D during Part II. Participants in group `Test` are shown explanations from the method they chose as most-useful. Participants who correctly identify whether or not there is a bug for at least 7 of the 10 examples during Part II are given a US$2 bonus. In addition to the 10 examples, we include a single attention check, where participants are told whether or not there is a bug; responses to the attention check did not impact compensation.

After Part II, participants are asked basic demographic questions, and general questions about whether or not they feel comfortable and confident with the bug

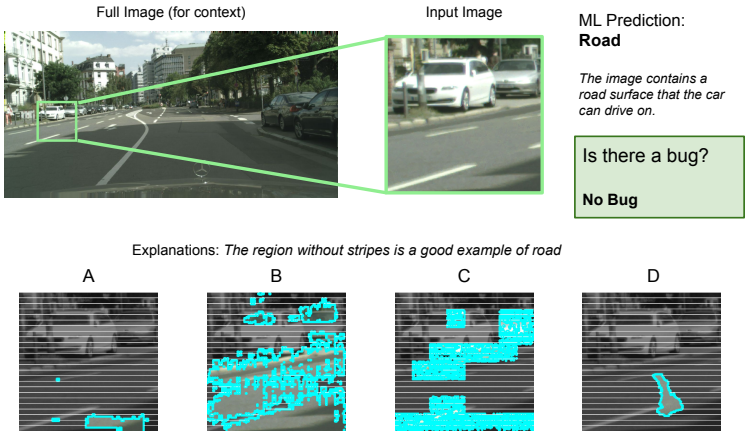



FIGURE 12.1: Example of model input and output (top) and explanations (bottom) used in Part I of all surveys. During Part II, participants are asked to *predict* whether or not there is a bug, given only one explanation (A, B, C, or D).

detection task.

**Data, Classification Tasks, and Models.** All input images in our survey are random 250x250-pixel crops of Cityscapes[2] images [91]; each image is a scene of a street, from a car driver's point of view. We use two classification tasks: in *Road*, images are labeled 1 if they at least 5% of their pixels show a road surface, and 0 otherwise. Similarly in task *Vehicles*, images are labeled 1 if at least 5% of their pixels show a vehicle (e.g. a car or truck). Ground truth labels are determined using pixel-level semantic maps from the Cityscapes dataset. For both tasks we train a classifier by modifying a pre-trained Resnet50 ([163]) model to have a fully-connected output layer with two output features; each model is trained for 30 epochs using cross-entropy loss, with a random 50% sample of the Cityscapes dataset. Images are normalized during both training and testing, and during training images are randomly flipped horizontally and vertically each with probability 1/2.

[2]The official license allows the use of the dataset for research purposes: https://www.cityscapes-dataset.com/license/

**Bugs.** We use two types of (intentional) bugs in this survey: *Weights* and *Noise*. The *Weights* bug is created by setting the weights in all convolutional layers and the final (fully-connected) layer to uniform-random values. The *Noise* bug is created by adding shot noise (also known as Poisson noise) to each input image [3]. This is a common corruption used to benchmark robustness in deep learning [166].

## 12.3 Experiment Results

First, in Section 12.3.1 we investigate the participant-reported usefulness of each explanation method for each classification task—we find that participants' usefulness ratings for each explanation method vary across context (classification task and bug type). Then, in Section 12.3.2 we find that in Part II participants are largely unable to identify bugs in new examples; furthermore, participants who are shown their preferred explanation method (group `Test`) tend to have *more trouble* identifying bugs than those who are shown one of their non-preferred methods (group `Control`).

**Survey Overview.** We conducted four surveys, one for each classification-bug combination (1: *Road+Weights*, 2: *Road+Noise*, 3: *Vehicle+Weights*, 4: *Vehicle+Noise*). 150 responses were collected for each survey; we discarded "invalid" responses from participants who failed the attention check, spent less than 3 minutes completing the survey, or took the survey multiple times. Table 12.2a shows the number of participants for each survey, for both `Control` and `Test`.

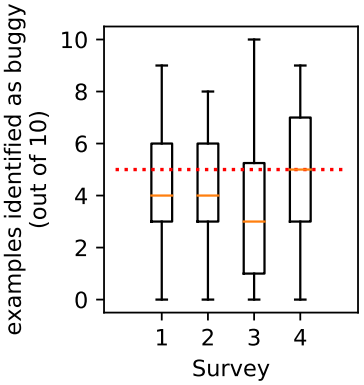
### 12.3.1 Part I: Reported usefulness varies across context

Figure 12.3 shows participant-reported usefulness over all examples. Participants rate the relative usefulness of explanation methods differently for different contexts:

[3] The severity of noise is the highest value used by [166]

| Number Valid (Invalid) Responses | | |
|---|---|---|
| Survey | `Control` | `Test` |
| 1 (*Road+Weights*) | 44 (31) | 44 (31) |
| 2 (*Road+Noise*) | 39 (35) | 40 (36) |
| 3 (*Vehicle+Weights*) | 40 (33) | 43 (34) |
| 4 (*Vehicle+Noise*) | 42 (31) | 47 (30) |

(A)




(B)

FIGURE 12.2: (a), Left: Number of valid responses in each survey, in `Control` and `Test` groups. The number of invalid responses is shown in parentheses. (b), Right: Number of bugs reported by participants over all 10 examples from Part II of each survey. In each survey, five of the 10 examples in Part II are buggy (shown as a dotted line).

for example, D is rated as more useful than C in both Survey 1 and 3, but D is rated as *less* useful in Survey 4. Using a Wilcoxon signed-rank test, each of these differences is significant with $p < .01$.

Similarly, perceived usefulness of some methods changes *across* context. For example, C is rated as more-useful in Survey 2 and 4 than in Survey 1 and 3 (significant by Mann-Whitney U, $p < .01$). However there is no significant difference between ratings for C for each noise type—*Weights* (Surveys 1 and 3) and *Noise* (Surveys 2 and 4).

We also observe some differences in the explanation methods selected by participants at the end of Part I (see Table 12.1). For example, about half of participants in Survey 2 select method B, compared to <20% in Surveys 2 and 3 and 34% in Survey 1. Using a chi-square test of independence, the selected explanation method in Survey 2 differs significantly from those in both Surveys 3 and 4 ($p < .01$); these differences are not significant between the other pairs of surveys.

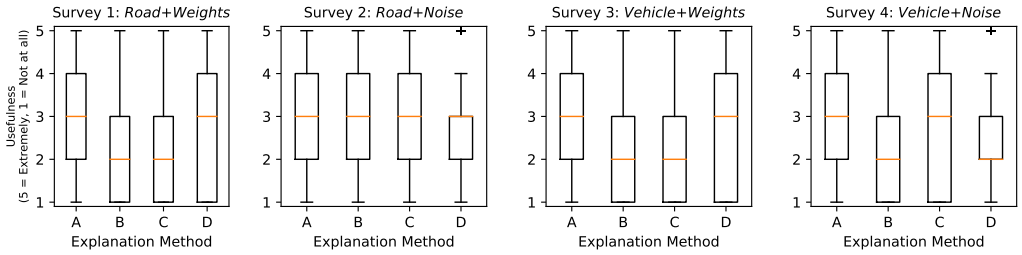
TABLE 12.1: Left: Number (percentage) of participants who selected each method at the end of Part I as the most-useful for identifying bugs in each survey. Right: Usefulness rank of the explanation method selected by participants at the end of Part I of the survey. The usefulness rank for each explanation method is the best (lowest) rank of the *average* usefulness ratings (1 = Not at all useful and 5 = Extremely useful) over all examples from Part I.

| | Selected Explanation Method | | | | Usefulness Rank of Selected Method | | | |
|---|---|---|---|---|---|---|---|---|
| Survey | A | B | C | D | 1st | 2nd | 3rd | 4th |
| 1 (*Road+Weights*) | 16 (18%) | 30 (34%) | 15 (17%) | 27 (31%) | 36 | 28 | 13 | 11 |
| 2 (*Road+Noise*) | 20 (25%) | 38 (48%) | 9 (15%) | 12 (11%) | 22 | 23 | 16 | 18 |
| 3 (*Vehicle+Weights*) | 17 (20%) | 16 (19%) | 8 (10%) | 42 (51%) | 33 | 31 | 13 | 6 |
| 4 (*Vehicle+Noise*) | 19 (21%) | 14 (16%) | 13 (15%) | 43 (48%) | 18 | 28 | 30 | 13 |
| **Overall** | 72 (21%) | 89 (29%) | 45 (13%) | 124 (37%) | 109 (32%) | 110 (32%) | 65 (19%) | 55 (16%) |

**Participants select methods that they rate as more-useful.** A quick glance at these results suggests that participants select explanations methods that *do not align* with their usefulness ratings; for example according to Table 12.1, about half of Survey 2 participants selected method B, yet the same participants rate B as no more-or less useful than A or C on average. However, by comparing relative rankings we find that most participants select a method that they ranked (relatively) highly. For each participant, we find the average usefulness rating (1 = Not at all useful and 5 = Extremely useful) for each explanation method, and then *rank* each method for each participant, where 1 is the method they rank as most-useful, and 4 is least-useful.[4] Table 12.1 shows the usefulness *rank* of the method selected by each participant at the end of Part I. For each survey, *most* participants selected the method they rated as 1st or 2nd most-useful.

Our main observations from Part I are that (a) people prefer *different explanations* in *different contexts*, and (b) the perceived usefulness of an explanation method *varies* across contexts—for example, see method C in Figure 12.3 . We believe that these expressed preferences are authentic, especially since at the end of Part I participants

[4]We calculate the *best* ranking of each method, meaning that if two methods are tied with the second-highest score, they both have rank 2.




(A) Survey 1 (B) Survey 2 (C) Survey 3 (D) Survey 4

FIGURE 12.3: Box plots of participant-reported usefulness for each explanation method, over all examples shown in Part I of the survey.

tend to select an explanation method that they rated as (relatively) useful (see Table 12.1).

There are multiple interpretations for our results: participants' usefulness ratings may be authentic—for example, participants in Survey 3 may feel that method A is more helpful than C for identifying bugs. On the other hand, these ratings may simply reflect the visual appeal of each explanation method, or perhaps other features of the input images. Regardless of what participants' preferences *mean*, it is clear that preferences vary both across *context* and *individual*. To investigate whether usefulness ratings relate to the downstream decision task, we turn to Part II of our survey.

### 12.3.2 Part II: User-Selected Explanations are Not Helpful for Identifying Bugs

We calculate a *bug score* for each participant—the number of examples (out of 10) in Part II where they correctly identify whether or not a buggy model produced the output and explanation. The bug-detection tasks in our survey proved to be difficult: the median bug score across all surveys is 5 (equivalent to flipping a coin), and the median score for all `Control` and `Test` groups is also 5, with the exception of `Control` in Survey 1 (median score = 6).

Furthermore, we cannot confirm **H2**: using a Mann-Whitney U test, we cannot reject the null hypothesis that scores of `Control` and `Test` groups come from different distribution; this is true for Survey 2 ($p = .48$), Survey 3 ($p = .13$), and Survey 4 ($p = .49$). However we *can* reject this null hypothesis for Survey 1 ($p < .01$)—where scores for group `Test` tend to be *greater* than for `Control`.

That is, in Surveys 2, 3, and 4, participants who use their preferred explanation method are just as (un)able to detect bugs as those who use a non-preferred explanation.

One possible reason that participants have such low bug detection scores is that they under-identify bugs. Figure 12.2b shows the number of examples in Part II (out of 10) where participants predicted that the model was buggy. In each survey five of the 10 examples in Part II were buggy, however participants tend to identify *fewer* than five examples as buggy.

**Different explanations lead to different bug scores.** There is little difference in the bug scores of participants who see their preferred explanation method (`Test`), and those who do not (`Control`). However, bug scores *do* vary across explanation methods. Figure 12.4 shows the bug scores for participants by the explanation method they used in Part II, for each survey. These results suggest that different explanation methods may be more- and less-effective at helping users identify bugs, and that their effectiveness varies across contexts. For example, in Survey 1 participants who use method C have an average bug score greater than 5, while those with method D have an average score near 4; the reverse is true for Surveys 2, 3, and 4. However we cannot concretely determine the relative usefulness of different explanation methods from these results, for two reasons: (1) the explanation methods used in Part II depend on users' expressed preferences during Part I, and (2) our sample size is too

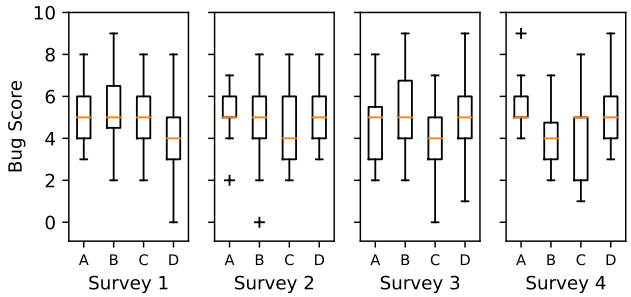



FIGURE 12.4: Bug scores for participants in each survey, by the explanation method used in Part II (A, B, C, or D).

small to draw conclusions.

## 12.4 A Framework for Context-Aware Explanations

Our empirical study demonstrates that following user preferences does not necessarily lead to helpful explanations. To address this issue, we propose a framework for designing explanations that relies on task-specific feedback from the user. Our setting is summarized in Figure 12.5: a ML system and explanation method outputs both *predictions* and *explanations* to a user, who uses this output to complete a downstream task.

We use a mathematical framework to reason in this setting. Let $\mathcal{X}$ and $\mathcal{Y}$ be the sets of input and output labels of the ML system.

Let $f : \mathcal{X} \to \mathcal{Y}$ denote the prediction function of the ML system: $f(x)$ is the predicted label for data point $x$.

Let $\mathcal{M}$ denote the set of possible explanation methods, and let $\mathcal{Z}$ denote the set of possible *explanations* returned by these methods.

Each explanation method $m \in \mathcal{M}$ is itself a function $m : \mathcal{X} \to \mathcal{Z}$, where $m(x)$ is the explanation for data point $x$.[5] Finally, the user considers the input data, the prediction, and the explanation, and determines the *usefulness* of this output.

[5]Explanation methods usually depend on the ML model; in our setting the ML model is fixed, and this dependence is implicit.

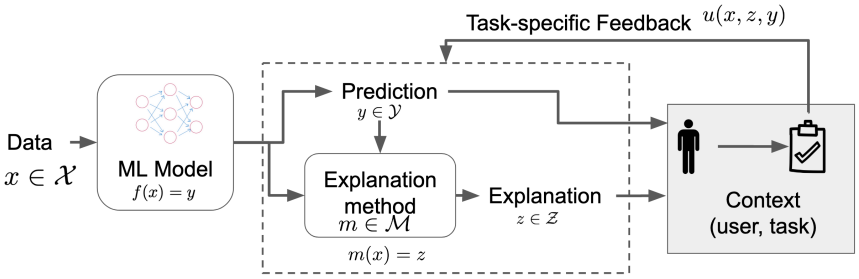



FIGURE 12.5: An overview of our framework for context-aware explanations. Data is passed to a ML model, which outputs a prediction. The user receives the ML prediction and the explanation, and assesses the usefulness in their downstream task (eg: debugging a ML model). Feedback from this task is used to *learn* the preferred explanation method for this particular context.

**Modeling the Context** We represent the usefulness in a given context with a *utility function* $u : \mathcal{X} \times \mathcal{Y} \times \mathcal{Z} \rightarrow [0, 1]$, which is a standard tool for modeling preferences. In a deployed setting we would use feedback from the user performing the downstream task to learn a numerical representation of $u$; here we adapt techniques from *preference elicitation* (e.g., [53]). There are two standard approaches for learning $u$, which use different types of questions. We can learn *relative* utility by using *comparison questions*: we observe which input-output tuple is more useful, $(x, y, z)$ or $(x', y', z')$ to the user for the downstream task. If $(x, y, x)$ is more useful, then we learn the constraint $u(x, y, s) > u(x', y', z')$, otherwise $u(x, y, s) < u(x', y', z')$. Alternatively, we can learn *absolute* utility by observing a user's accuracy on the downstream task. This can map directly to numerical values $u(x, y, z)$.

**Modeling Explanations** We represent each explanation $z$ as a point in an *embedding space*. This space can be characterized by using interpretable dimensions, where each dimension is an existing metric developed by the HCI and ML communities (such as fidelity, comprehensibility, computational complexity etc.).

**Generating Context-Aware Explanations** After learning a model of utility in a decision making context, $u(x, y, z)$, our framework can be used to generate context-aware explanations. When presented with a new data point $x$ and ML output $y$, we identify the *most useful* explanation for the context by solving the problem $\max_{z \in \mathcal{Z}} u(x, y, z)$.

### 12.4.1 Simulation: Recommending Explanations

As a proof-of-concept of our framework, we use a small simulation to demonstrate the utility of downstream task feedback. We simulate an explanation *recommendation* setting, where a user needs to complete a downstream task (determine whether a model is buggy), using a single ML explanation that we recommend to them. Each recommender chooses a single explanation to show the user (A, B, C, or D), with the goal of maximizing the likelihood that the user completes the task correctly.

We consider two recommendation paradigms: *Ratings* and *Task*. Both recommendation methods aim to help the user complete a downstream task, however *Ratings* takes user ratings as input, while *Task* takes other user's performance on the downstream task. In *Ratings* we first ask the user how helpful each explanation is for the downstream task, and we recommend an explanation based on their ratings. In *Task* we recommend an explanation based only on *other users'* accuracy when using each explanation for the same task.

We simulate both the *Ratings* and *Task* paradigms using data from our survey studies. For this simulation we treat users as identical, and each example in Part II is a different "task"; each survey participant completes 10 tasks in Part II, so in each survey there are roughly 700 tasks completed in total. For the *Task* paradigm, we learn a separate recommendation "model" for each individual, $\boldsymbol{p} = [p_A, p_B, p_C, p_D]$, where $p_X$ is the probability that the user will complete the downstream task correctly using explanation method $X$. We learn each model using *other users'* accuracy on

completed tasks: $p_X$ is probability that other users complete the task correctly with explanation $X$ (calculated as the number of correctly-completed tasks divided by the total number of completed tasks).

For method *Ratings* we learn one model for each explanation method $X \in \{A, B, C, D\}$. Each model takes a single parameter as input—the user's average normalized rating for method $X$—and returns the probability that the user will complete the task correctly.

For each survey we generate a dataset with one entry for each Part II example (roughly 700 rows). Columns include the explanation method used by the participant (A, B, C, or D), their normalized average usefulness rating for this method (from Part I), and their *task accuracy*: 1 if they correctly identified the example as bug/no-bug, and 0 if they did not.

We generate 1000 random 80%-20% train-test splits of this dataset. For each split we train the *Task* model, and three *Ratings* models (logistic regression, SVM (RBF), and SVM (linear)) to predict (binary) task accuracy on the training split. We then calculate the accuracy and log loss for each model on the test split.

Table 12.2 shows the results of this simulation: overall, the accuracy of all methods are near chance (0.5). In Survey 3, all *Ratings* methods outperform the *Task*. This suggests that usefulness ratings are in fact predictive of downstream task accuracy. Upon closer examination, participant ratings in Survey 3 are *negatively correlated* with task accuracy—meaning that participants who rate an explanation as *less useful* also complete the downstream task *more accurately* than participants who rate it as more-useful; see Appendix E.1 for more discussion. For both Survey 1 and Survey 4, the *Task* model tends to outperform the *Ratings* models. Overall, this suggests that user ratings are at best loosely related to downstream task performance. In our simulations, recommendations driven by other users' task accuracy—in the

TABLE 12.2: Accuracy and log loss (cross-entropy) on the test set, over 1000 random 80-20 train-test splits for both *Ratings* and *Task* recommender methods. The mean ($\pm$ standard deviation) is reported across all 1000 splits. Models in the *Ratings* paradigm use participant usefulness ratings to predict their accuracy in the task; the *Task* paradigm uses *other participants'* accuracy in the same task.

| | *Ratings* Paradigm | | | | | | *Task* Paradigm | |
|---|---|---|---|---|---|---|---|---|
| | Logistic Regression | | SVM (RBF) | | SVM (Linear) | | | |
| Survey | Accuracy | Log Loss | Accuracy | Log Loss | Accuracy | Log Loss | Accuracy | Log Loss |
| 1 (*Road+Weights*) | $0.47 \pm 0.03$ | $0.70 \pm 0.00$ | $0.53 \pm 0.04$ | $0.70 \pm 0.00$ | $0.47 \pm 0.03$ | $0.69 \pm 0.01$ | $0.52 \pm 0.03$ | $0.69 \pm 0.01$ |
| 2 (*Road+Noise*) | $0.51 \pm 0.04$ | $0.69 \pm 0.00$ | $0.50 \pm 0.03$ | $0.70 \pm 0.00$ | $0.50 \pm 0.03$ | $0.70 \pm 0.04$ | $0.50 \pm 0.04$ | $0.70 \pm 0.01$ |
| 3 (*Vehicle+Weights*) | $0.57 \pm 0.03$ | $0.68 \pm 0.01$ | $0.57 \pm 0.03$ | $0.68 \pm 0.01$ | $0.55 \pm 0.04$ | $0.69 \pm 0.01$ | $0.51 \pm 0.03$ | $0.69 \pm 0.01$ |
| 4 (*Vehicle+Noise*) | $0.50 \pm 0.03$ | $0.69 \pm 0.00$ | $0.51 \pm 0.03$ | $0.70 \pm 0.00$ | $0.50 \pm 0.03$ | $0.71 \pm 0.04$ | $0.56 \pm 0.03$ | $0.69 \pm 0.01$ |

*Task* paradigm—are slightly more predictive of an explanation's usefulness than the user's ratings.

## 12.5 Discussion

ML models are deployed in a wide variety of contexts, with a wide variety of users. We are primarily concerned with applications where a (human) user completes a task with the assistance of a ML model, such as an engineer debugging a computer vision system, or a physician making a diagnosis with a ML assistant. In these scenarios, it is often desirable to *explain* the model's behavior. Computer scientists have developed a wide variety explanation methods, driven by a wide variety of practical metrics—related both to model performance and to user-model interactions. We argue that in deployed settings it is essential to consider the *context* in which a model is deployed, and most importantly the downstream task that the user faces.

In our empirical study we find that participants rate the usefulness of different explanation methods differently depending on the context in which they are used. Furthermore, different users prefer different explanation methods. However, we find that user preferences for explanations are unrelated to the *usefulness* of the explanation in assisting the user in a downstream task.

We argue that, in order for ML explanations to be useful, they should be guided

by downstream feedback from the user's task—not from direct user feedback. Our proposed framework outlines one approach for learning useful explanations, which we term context-aware explanations.

## 12.6 Authors and Publication

This chapter was written by Duncan C McElfresh, Vedant Nanda, and John P Dickerson.

# Chapter 13: Indecision Modeling

## 13.1 Introduction

The previous two chapters consider ways in which humans interact with an AI or ML system. In this chapter we take one step back, and consider the *data* used to train an AI or ML decision support system. In many deployed settings, AI and ML systems are trained to mimic human behavior in a similar task—this is true of ML applications in self-driving cars, hiring, and criminal justice (among others).

A growing body of research views this challenge through the lens of *preference aggregation*. From this perspective, researchers aim to (1) understand the preferences (or values) of the relevant stakeholders, and (2) design an AI system that aligns with the aggregated preferences of all stakeholders. This approach has been proposed recently in the context of self-driving cars [228] and organ allocation [133]. These approaches rely on a mathematical model of stakeholder preferences–which is typically *learned* using data collected via hypothetical decision scenarios or online surveys.[1] There is a rich literature addressing how to elicit preferences accurately and efficiently, spanning the fields of computer science, operations research, and social science.

It is critical that these *observed* preferences accurately represent peoples' *true* preferences, since these observations guide deployed AI systems. Importantly, the way we measure (or *elicit*) preferences is closely tied to the accuracy of these observations.

[1] The MIT Moral Machine project is one example: https://www.moralmachine.net/

In particular, it is well-known that both the order in which questions are asked, and the set of choices presented, impact expressed preferences [98, 99].

Often people choose *not* to express a strict preference, in which case we call them *indecisive*. The economics literature has suggested a variety of explanations for indecision [141]—for example when there are no desirable alternatives, or when all alternatives are perceived as equivalent. Moral psychology research has found that people often "do not want to play god" in moral situations, and would prefer for somebody or something else to take responsibility for the decision [139].

In philosophy, indecision of the kind discussed in this chapter is typically linked to a class of moral problems called symmetrical dilemmas, in which an agent is confronted with the choice between two alternatives that are or appear to the agent equal in value [286].[2] Much of the literature concerns itself with the morality and rationality of the use of a randomizer, such as flipping a coin, to resolve these dilemmas. Despite some disagreements over details [52, 115, 158, 220], many philosophers do agree that flipping a coin is often a viable course of action in response to indecision[3].

The present study accepts the assumption that flipping a coin is typically an expression of one's preference to *not* decide between two options, but goes beyond the received view in philosophy by suggesting that indecision can also be common and acceptable when the alternatives are *asymmetric*. We show that people often do adopt coin flipping strategies in asymmetrical dilemmas, where the alternatives are not equal in value. Thus, the use of a randomizer is likely to play a more complex role in moral decision making than simply as a tie breaker for symmetrical dilemmas.

[2] Sophie's Choice is a well-known example: a guard at the concentration camp cruelly forces Sophie to choose one of her two children to be killed. The guard will kill both children if Sophie refuses to choose. Sophie's reason for not choosing one child applies equally to another, hence the symmetry.

[3] With some exceptions: for example, see [255].

Naturally, people are also sometimes indecisive when faced with difficult decisions related to AI systems. However it is commonly assumed in the preference modeling literature that people always express a strict preference, unless (A) the alternatives are approximately equivalent, or (B) the alternatives are incomparable. Assumption (A) is mathematically convenient, since it is necessary for preference *transitivity*.[4] Since indecision is both a common and meaningful response, strict preferences alone cannot accurately represent peoples' real values. Thus, AI researchers who wish to guide their systems using observed preferences should be aware of the hidden meanings of indecision. We aim to uncover these meanings in a series of studies.

**Our Contributions.**

- We conduct a pilot experiment to illustrate how different interpretations of indecision lead to different outcomes (§ 13.2). Using hypothesis testing, we reject the common assumption (A) that indecision is expressed only toward equivalent—or symmetric—alternatives.

- Then, drawing on ideas from psychology, philosophy, and economics, we discuss several other potential reasons for indecision, drawing (§ 13.3). We formalize these ideas as mathematical indecision *models*, and develop a probabilistic interpretation that lends itself to computation (§ 13.4).

- To test the utility of these models, we conduct a second experiment to collect a much larger dataset of decision responses (§ 13.5). We take a machine learning (ML) perspective, and evaluate each model class based on its goodness-of-fit

[4] My preferences are transitive if "I prefer A over B" and "I prefer B over C" implies "I prefer A over C".

to this dataset. We assess each model class for predicting *individual* peoples' responses, and then we briefly investigate group decision models.

In all of our studies, we ask participants *who should receive the kidney?* in a hypothetical scenario where two patients are in need of a kidney, but only one kidney is available. As a potential basis for their answers, participants are given three "features" of each patient: age, amount of alcohol consumption, and number of young dependents.

We chose this task for several reasons: first, kidney exchange is a real application where algorithms influence—and sometimes make—important decisions about who receives which organ.[5] Second, organ allocation is a *difficult* problem: there are far fewer donors organs than there are people in need of a transplant.[6] Third, the question of *who* should receive these scarce resources raises serious ethical dilemmas [277]. Kidney allocation is also a common motivation for studies of fair resource allocation [8, 213, 215]. Furthermore, this type of scenario is frequently used to study peoples' preferences and behavior [133, 137, 138, 233]. Importantly, this prior work focuses on peoples' *strict* preferences, while we aim to study indecision.

## 13.2 Study 1: Indecision is Not Random Choice

We first conduct a pilot study to illustrate the importance of measuring indecision. Here we take the perspective of a preference-aggregator; we illustrate this perspective using a brief example: Suppose we must choose between two alternatives (X or Y), based on the preferences of several stakeholders. Using a survey we ask all

[5] Many exchanges match patients and donors algorithmically, including the United Network for Organ Sharing (https://unos.org/transplant/kidney-paired-donation/) and the UK national exchange (https://www.odt.nhs.uk/living-donation/uk-living-kidney-sharing-scheme/).

[6] There are around 100,000 people in need of a transplant today (https://unos.org/data/transplant-trends/), and about 22,000 transplants have been conducted in 2020.

stakeholders to express a *strict* preference (to "vote") for their preferred alternative; X receives 10 votes while Y receives 6 votes, so X wins.

Next we conduct the same survey, but allow stakeholders to vote for "indecision" instead; now, X receives 4 votes, Y receives 5 votes, and "indecision" receives 7 votes. If we assume that voters are indecisive only when alternatives are nearly equivalent (assumption (A) from Section 13.1), then each "indecision" vote is analogous to one half-vote for both X and Y, and therefore Y wins. In other words, in the first survey we assume that all indecisive voters choose randomly between X and Y. However, if indecision has another meaning, then it is not clear whether X or Y wins. Thus, in order to make the best decision for our constituents we must understand what meaning is conveyed by indecisive voters. Unfortunately for our hypothetical decision maker, assumption (A) is not always valid.

Using a small study, we test—and reject—assumption (A), which we frame as two different hypotheses, **H0-1**: *if we discard all indecisive votes, then both X and Y receive the same proportion votes, whether or not indecision is allowed.* A second related hypothesis is **H0-2**: *if we assign half of a vote to both X and Y when someone is indecisive, then both X and Y receive the same proportion votes, whether or not indecision is allowed.* We conducted the hypothetical surveys described above, using 15 kidney allocation questions (see Appendix F.1 for the survey text and analysis). Participants were divided into two groups: participants in group *Indecisive* (N=62) were allowed to express indecision (phrased as "flip a coin to decide who receives the kidney"), while group *Strict* (N=60) was forced to choose one of the two recipients. We test **H0-1** by identifying the *majority patient*, "X" (who received the most votes) and the *minority patient* "Y" for each of the 15 questions (details of this analysis are in Appendix F.1). Overall, group *Indecisive* cast 581 (74) votes for the majority (minority) patient, and 275 indecision votes; the *Strict* group cast 751 (149) votes for the majority (minority)

patient. Using a Pearson's chi-squared test we reject **H0-1** ($p < 0.01$). According to **H0-2**, we might assume that all indecision votes are "effectively" one half-vote for both the minority and majority patient. In this case, the *Indecisive* group casts 718.5 (211.5) "effective" votes for the majority (minority) patients; using these votes we reject **H0-2** ($p < 0.01$).

In the context of our hypothetical choice between X and Y, this finding is troublesome: since we reject **H0-1** and **H0-2**, we cannot choose a winner by selecting the alternative with the most votes—or, if indecision is measured, the most "effective" votes. If indecision has other meanings, then the "best" alternative depends on which meanings are used by each person; this is our focus in the remainder of this chapter.

## 13.3 Models for Indecision

The psychology and philosophy literature find several reasons for indecision, and many of these reasons can be approximated by numerical decision models. Before presenting these models, we briefly discuss their related theories from psychology and philosophy.

**Difference-Based Indecision** In the preference modeling literature it is sometimes assumed that people are indecisive only when both alternatives (X and Y) are indistinguishable. That is, the perceived difference between X and Y is too small to arrive at a strict preference. In philosophy, this is referred to as "the possibility of parity" [77].

**Desirability-Based Indecision** In cases where both alternatives are not "good enough", people may be reluctant to choose one over the other. This has been referred to as

"single option aversion" [225], when consumers do not choose between product options if none of the options is sufficiently likable. Zakay [328] observes this effect in single-alternative choices: people reject an alternative if it is not sufficiently close to a hypothetical "ideal". Similarly, people may be indecisive if *both* alternatives are attractive. People faced with the choice between two highly valued options often opt for an indecisive resolution in order to manage negative emotions [208].

**Conflict-Based Indecision** People may be indecisive when there are both good and bad attributes of each alternative. This is phrased as *conflict* by Tversky and Shafir [302]: people have trouble deciding between two alternatives if neither is better than the other in *every way*. In the AI literature, the concept of *incomparability* between alternatives is also studied [245].

While these notions are intuitively plausible, we need mathematical definitions in order to model observed preferences. That is the purpose of the next section.

## 13.4 Indecision Model Formalism

In accordance with the literature, we refer to decision makers as *agents*. Agent preferences are represented by binary relations over each pair of items $(i, j) \in \mathcal{I} \times \mathcal{I}$, where $\mathcal{I}$ is a universe of items. We assume agent preferences are *complete*: when presented with item pair $(i, j)$, they expresses exactly one response $r \in \{0, 1, 2\}$, which indicates:

- $r = 1$, or $i \succ j$: the agent prefers $i$ more than $j$
- $r = 2$, or $i \prec j$: the agent prefers $j$ more than $i$
- $r = 0$, or $i \sim j$: the agent is indecisive between $i$ and $j$

When preferences are complete and transitive,[7] then the preference relation corresponds to a weak ordering over all items [284]. In this case there is a utility function representation for agent preferences, such that $i \succ j \iff u(i) > u(j)$, and $i \sim j \iff u(i) = u(j)$, where $u : \mathcal{I} \to \mathbb{R}$ is a continuous function. We assume each agent has an underlying utility function, however in general we *do not* assume preferences are transitive. In other words, we assume agents can rank items based on their relative value (represented by $u(\cdot)$), but in some cases they consider other factors in their response—causing them to be indecisive. Next, to model indecision we propose mathematical representations of the causes for indecision from Section 13.3.

### 13.4.1 Mathematical Indecision Models

All models in this section are specified by two parameters: a utility function $u(\cdot)$ and a threshold $\lambda$. Each model is based on *scoring functions*: when the agent observes a query they assign a numerical score to each response, and they respond with the response type that has maximal score; we assume that score ties are broken randomly, though this assumption will not be important. In accordance with the literature, we assume the agent observes random iid additive error for each response score (see, e.g., Soufiani et al. [293]). Let $S_r(i, j)$ be the agent's score for response $r$ to comparison $(i, j)$; the agent's response is given by

$$R(i, j) = \arg\max_{r \in \{0,1,2\}} S_r(i, j) + \epsilon_{rij}.$$

That is, the agent has a deterministic score for each response $S_r(i, j)$, but when making a decision the agent observes a noisy version of this score, $S_r(i, j) + \epsilon_{rij}$. We make the common assumption that noise terms $\epsilon_{rij}$ are iid Gumbel-distributed, with scale

[7] Agent preferences are transitive if $i \succ j$ and $i \succ k$ iff $i \succ k$.

$\mu = 1$. In this case, the distribution of agent responses is

$$p(i, j, r) = \frac{e^{S_r(i,j)}}{e^{S_0(i,j)} + e^{S_1(i,j)} + e^{S_2(i,j)}}. \tag{13.1}$$

Each indecision model is defined using different score functions $S_r(\cdot, \cdot)$. Score functions for strict responses are always symmetric, in the sense that $S_2(i, j) = S_1(j, i)$; thus we need only define $S_1(\cdot, \cdot)$ and $S_0(\cdot, \cdot)$. We group each model by their cause for indecision from Section 13.3.

**Difference-Based Models:** `Min-`$\delta$**,** `Max-`$\delta$ Agents are indecisive when the utility difference between alternatives is either smaller than threshold $\lambda$ (`Min-`$\delta$) or greater than $\lambda$ (`Max-`$\delta$). The score functions for these models are

$$\texttt{Min-}\delta: \begin{cases} S_1(i,j) & \equiv u(i) - u(j) \\ S_0(i,j) & \equiv \lambda \end{cases}$$

$$\texttt{Max-}\delta: \begin{cases} S_1(i,j) & \equiv u(i) - u(j) \\ S_0(i,j) & \equiv 2|u(i) - u(j)| - \lambda \end{cases}$$

Here $\lambda$ should be non-negative: for example with `Min-`$\delta$, $\lambda \leq 0$ means the agent is never indecisive, while for `Max-`$\delta$ this means the agent is always indecisive. Model `Max-`$\delta$ seems counter-intuitive (if one alternative is clearly better than the other, why be indecisive?), yet we include it for completeness. Note that this is only one example of a difference-based model: instead the agent might assess alternatives using a distance measure $d : \mathcal{I} \times \mathcal{I} \to \mathbb{R}_+$, rather than $u(\cdot)$.

**Desirability-Based Models:** `Min-`$U$**,** `Max-`$U$ Agents are indecisive when the utility of *both* alternatives is below threshold $\lambda$ (`Min-`$U$), or when the utility of both alternatives is greater than $\lambda$ (`Max-`$U$). Unlike the difference-based models, $\lambda$ here may be

positive or negative. The score functions for these models are

$$\texttt{Min-}U: \begin{cases} S_1(i,j) & \equiv u(i) \\ S_0(i,j) & \equiv \lambda \end{cases}$$

$$\texttt{Max-}U: \begin{cases} S_1(i,j) & \equiv u(i) \\ S_0(i,j) & \equiv 2\min\{u(i),u(j)\} - \lambda \end{cases}$$

Both of these models motivated in the literature (see § 13.3).

**Conflict-Based Model:** `Dom` In this model the agent is indecisive unless one alternative *dominates* the other in all features, by threshold at least $\lambda$. For this indecision model, we need a utility measure associated with each feature of each item; for this purpose, let $u_n(i)$ be the utility associated with feature $n$ of item $i$. As before, $\lambda$ here may be positive or negative. The score functions for this model are

$$\texttt{Dom}: \begin{cases} S_1(i,j) & \equiv \min_{n\in[N]}\left(u_n(i) - u_n(j)\right) \\ S_0(i,j) & \equiv \lambda \end{cases}$$

This is one example of a conflict-based indecision model, though we might imagine others.

These models serve as a class of hypotheses which describe how agents respond to comparisons when they are allowed to be indecisive. Using the response distribution in (13.1), we can assess how well each model fits with an agent's (possibly indecisive) responses. However, in many cases agents are *required* to express strict preferences—they are not allowed to be indecisive (as in Section 13.2). With slight modification the score-based models from this section can be used even when agents are forced to express *only* strict preferences; we discuss this in the next section.

### Indecision Models for Strict Comparisons

We assume that agents may prefer to be indecisive, even when they are required to express strict preferences. That is, we assume that agents use an underlying *indecision* model to express *strict* preferences. When they cannot express indecision, we assume that they either *resample* from their decision distribution, or they choose randomly. That is, we assume agents use a two-stage process to respond to queries: first they sample a response $r$ from their response distribution $p(\cdot, \cdot, r)$; if $r$ is strict (1 or 2), then they express it, and we are done. If they sample indecision (0), then they flip a weighted coin to decide how to respond:

(heads) with probability $q$ they re-sample from their response distribution until they sample a strict response, without flipping the weighted coin again

(tails) with probability $1-q$ they choose uniformly at randomly between responses 1 and 2.

That is, they respond according to distribution

$$p_{strict}(i,j,r) \equiv \begin{cases} q\left(\frac{e^{S(i,j)}+(1/2)e^{S_0(i,j)}}{C}\right) + \frac{1-q}{D}\left(e^{S_1(i,j)}\right) & \text{if } r=1 \\ q\left(\frac{e^{S_2(i,j)}+(1/2)e^{S_0(i,j)}}{C}\right) + \frac{1-q}{D}e^{S_2(i,j)} & \text{if } r=2 \end{cases} \tag{13.2}$$

Here, $C \equiv e^{S_0(i,j)} + e^{S_1(i,j)} + e^{S_2(i,j)}$ , and $D \equiv e^{S_1(i,j)} + e^{S(2(i,j)}$. The (heads) condition from above has another interpretation: the agent chooses to sample from a "strict" logit, induced by only the score functions for strict responses, $S_1(i,j)$ and $S_1(i,j)$. We discuss this model in more detail, and provide an intuitive example, in Appendix F.2.

We now have mathematical indecision models which describe how indecisive agents respond to comparison queries, both when they are allowed to express indecision (§ 13.4.1), and when they are not (§ 13.4.1). The model in this section, and

response distributions (13.1) and (13.2), represent one way indecisive agents might respond when they are forced to express strict preferences. The question remains whether any of these models accurately represent peoples' expressed preferences in real decision scenarios. In the next section we conduct a second, larger survey to address this question.

## 13.5 Study 2: Fitting Indecision Models

In our second study, we aim to *model* peoples' responses in the hypothetical kidney allocation scenario using indecision models from the previous section as well as standard preference models from the literature. The models from the previous section can be used to predict peoples' responses, both when they are allowed to be indecisive, and when they are not. To test both class of models, we conducted a survey with two groups of participants, where one group was were given the option to express indecision, and the other was not. Each participant was assigned to 1 of the 150 random sequences, each of which contains 40 pairwise comparisons between two hypothetical kidney recipients with randomly generated values for age, number of dependents, and number of alcoholic drinks per week. We recruited 150 participants for group *Indecisive*, which was given the option to express indecision[8]. 18 participants were excluded from the analysis for failing attention checks, leaving us with a final sample of N=132. Another group, *Strict* (N=132), was recruited to respond to the same 132 sequences, but without the option to express indecision.

We remove 26 participants from *Indecisive* who never express indecision, because it is not sensible to compare goodness-of-fit for different indecision models when the agent never chooses to be indecisive. This study was reviewed and approved

[8] As in Study 1, this is phrased as "flip a coin."

by our organization's Institutional Review Board; please see Appendix F.1 for a full description of the survey and dataset.

**Model Fitting.** In order to fit these indecision models to data, we assume that agent utility functions are linear: each item $i \in \mathcal{I}$ is represented by feature vector $\boldsymbol{x}^i \in \mathbb{R}^N$; agent utility for item $i$ is $u(i) = \boldsymbol{u}^\top \boldsymbol{x}^i$, where $\boldsymbol{u} \in \mathbb{R}^N$ is the agent's *utility vector*. We take a maximum likelihood estimation (MLE) approach to fitting each model: i.e., we select agent parameters $\boldsymbol{u}$ and $\lambda$ which maximize the log-likelihood (LL) of the training responses. Since the LL of these models is not convex, we use random search via a Sobol process [288]. The search domain for utility vectors is $\boldsymbol{u} \in [-1, 1]^N$, the domain for probability parameters is $(0, 1)$, and the domain for $\lambda$ depends on the model type (see Appendix F.2). The number of candidate parameters tested and the nature of the train-test split vary between experiments. All code used for our analysis is available online, [9] and details of our implementation can be found in Appendix F.2.

We explore two different preference-modeling settings: learning individual indecision models, and learning group indecision models.

### 13.5.1 Individual Models

The indecision models from Section 13.4 are indented to describe how an indecisive agent responds to queries—both when they are given the option to be indecisive, and when they are not. Thus, we fit each of these models to responses from both participant groups: *Indecisive* and *Strict*. For each participant we randomly split their question-response pairs into a training and testing set of equal size (20 responses each). For each participant we fit all five models from Section 13.4, and

[9] https://github.com/duncanmcelfresh/indecision-modeling

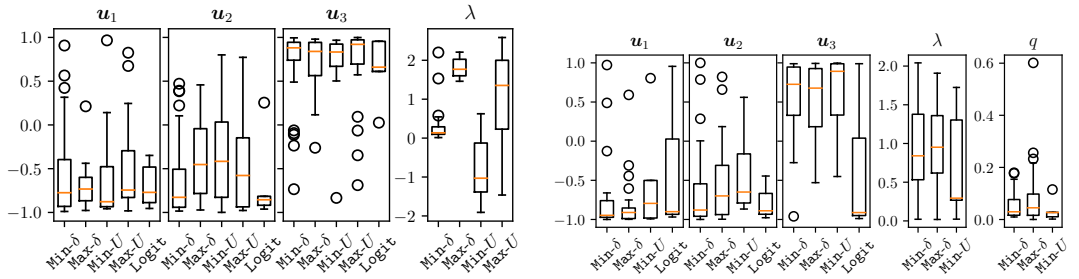
| | Group *Indecisive* (indecision & strict responses) | | | | Group *Strict* (only strict responses) | | | |
|---|---|---|---|---|---|---|---|---|
| Model | #1st | #2nd | #3rd | Train/Test LL | # 1st | # 2nd | # 3rd | Train/Test LL |
| Min-$\delta$ | 29 (27%) | 23 (22%) | 13 (12%) | -0.82/-0.85 | 26 (20%) | 53 (40%) | 34 (26%) | -0.44/-0.47 |
| Max-$\delta$ | 11 (10%) | 12 (11%) | 19 (18%) | -0.81/-0.90 | 31 (23%) | 57 (43%) | 25 (19%) | -0.44/-0.47 |
| Min-$U$ | 8 (8%) | 32 (30%) | 17 (16%) | -0.83/-0.88 | 1 (1%) | 5 (4%) | 20 (15%) | -0.53/-0.56 |
| Max-$U$ | 22 (21%) | 23 (22%) | 12 (11%) | -0.81/-0.83 | 1 (1%) | 5 (4%) | 15 (11%) | -0.53/-0.55 |
| Dom | 0 (0%) | 3 (3%) | 9 (8%) | -0.88/-0.95 | 2 (2%) | 4 (3%) | 3 (2%) | -0.57/-0.58 |
| Logit | 5 (5%) | 12 (11%) | 31 (29%) | -0.84/-0.90 | 4 (3%) | 5 (4%) | 27 (20%) | -0.53/-0.55 |
| Rand | 1 (1%) | 0 (0%) | 3 (3%) | -1.10/-1.10 | 6 (5%) | 0 (0%) | 1 (1%) | -0.69/-0.69 |
| MLP | 30 (28%) | 1 (1%) | 2 (2%) | -0.04/-1.15 | 61 (46%) | 3 (2%) | 7 (5%) | -0.03/-0.49 |

TABLE 13.1: Best-fit models for individual participants in group *Indecisive* (left) and *Strict* (right). The number of participants for which each model has the largest test log-likelihood (#1st), second-largest test LL (#2nd), as well as third-largest (#3rd) are given for each model, and the median training and test LL over all participants.

two baseline methods: Rand (express indecision with probability $q$ and chooses randomly between alternatives otherwise), MLP (a multilayer perceptron classifier with two hidden layers with 32 and 16 nodes). We use MLP as a state-of-the-art benchmark, against which we compare our models; we use this benchmark to see how close our new models are to modern ML methods.

For group *Indecisive* we estimate parameter $q$ for NaiveRand from the training queries; for *Strict* $q$ is 0. For MLP we train a classifier with one class for each response type, using scikit-learn [241]: for *Indecisive* responses we train a three-class model ($r \in \{0, 1, 2\}$), and for *Strict* we train a two-class model ($r \in \{1, 2\}$).

**Goodness-of-fit.** Using the standard ML approach, we select the best-fit models for each agent using the training-set LL, and evaluate the performance of these best-fit models using the test-set LL. Table 13.1 shows the number of participants for which each model was the 1st-, 2nd-, and 3rd best-fit for each participant (those with the greatest training-set LL), and the median test and train LL for each model. First we observe that *no indecision model* is a clear winner: several different models appear in the top 3 for each participant. This suggests that different indecision models fit different individuals better than others — there is not a single model that reflects everyone's choices. However, some models perform better than others: Min-$\delta$ and




(A) Participant best-fit model parameters for *Indecisive*

(B) Participant best-fit model parameters for *Strict*

FIGURE 13.1: Best-fit parameters for each indecision model, for participants in group *Indecisive* (top) and *Strict* (bottom). Elements of the agent utility vector correspond to patient age ($u_1$), alcohol consumption ($u_2$), and number of dependents ($u_3$); the interpretation of $\lambda$ depends on the model class. Only participants for which the model is the 1st-best-fit are included (see Table 1).

`Max-`$\delta$ appear often in the top 3 models, as does `Max-`$U$ for group *Indecisive*.

It it is somewhat surprising the `Max-`$\delta$ fits participant responses, since this model does not seem intuitive: in `Max-`$\delta$, agents are indecisive when two alternatives have *very different* utility—i.e., one has much greater utility than the other. It is also surprising the `Max-`$U$ is a good fit for group *Indecisive*, but not for *Strict*. One interpretation of this fact is that some people use (a version of) `Max-`$U$ when they have the option, but they *do not* use `Max-`$U$ when indecision is not an option. Another interpretation is that our modeling assumptions in Section 13.4.1 are wrong—however our dataset cannot definitively explain this discrepancy.

Finally, `MLP` is the most common best-fit model for all participants in both groups, though it is rarely a 2nd- or 3rd-best fit. This suggests that the `MLP` benchmark accurately models *some* participants' responses, and performs poorly for others; we expect this is due to overfitting. While `MLP` is more accurate than our models in some cases, it does not shed light on why people are indecisive.

It is notable that some indecision models (`Min-`$\delta$ and `Max-`$\delta$) outperform the standard logit model (`Logit`), both when they are learned from responses including indecision (group *Indecisive*), and when they are learned from only strict responses

(group *Strict*). Thus, we believe that these indecision models give a more-accurate representation for peoples' decisions than the standard logit, both when they are given the option to be indecisive, and when they are not.

Since these indecision models may be accurate representations of peoples' choices, it is informative to examine the best-fit parameters. Figure 13.1 shows best-fit parameters for participants in group *Indecisive* (top) and *Strict* (bottom); for each indecision model, we show all learned parameters for participants for whom the model is the 1st-best-fit (see Table 13.1). Importantly, the best-fit values of $\boldsymbol{u}_1$, $\boldsymbol{u}_2$, and $\boldsymbol{u}_3$ are similar for all models, in both groups. That is, *in general*, people have similar relative valuations for different alternatives: $\boldsymbol{u}_1 < 0$ means younger patients are preferred over older patients, $\boldsymbol{u}_2 < 0$ means patients who consume less alcohol are preferred more; $\boldsymbol{u}_3 > 0$ means that patients with more dependents are preferred more. We emphasize that the indecision model parameters for group *Strict* (bottom panel of Figure 13.1) are learned using only strict responses.

These models are fit using only 20 samples, yet they provide useful insight into how people make decisions. Importantly, our simple indecision models fit observed data better than the standard logit—both when people can express indecision, and when they cannot. Thus, contrary to the common assumption in the literature, not all people are indecisive *only* when two alternatives are nearly equivalent. This assumption may be true for some people (participants for which `Min-`$\delta$ is a best-fit model), but it is not always true.

### 13.5.2 Group Models

Next we turn to group decision models, where the goal is for an AI system to make decisions that reflect the values of a certain group of humans. In the spirit of the social choice literature, we refer to agents as "voters", and suggested decisions as

"votes". We consider two distinct learning paradigms, where each reflects a potential use-case of an AI decision making system.

The first paradigm, *Population Modeling*, concerns a large or infinite number of voters; our goal is to estimate responses to new decision problems that are the *best* for the entire population. This scenario is similar to conducting a national poll: we have a population including thousands or millions of voters, but we can only sample a small number (say, hundreds) of votes. Thus, we aim to build a model that represents the entire population, using a small number of votes from a small number of voters. There are several ways to aggregate uncertain voter models (see for example Chapter 10 of Brandt et al. [64]); our approach is to estimate the next vote from a random voter in the population. Since we cannot observe all voters, our model should generalize not only a "known" voter's future behavior, but *all* voters' future behavior.

In the second paradigm, *Representative Decisions*, we have a small number of "representative" voters; our goal is to estimate best responses to new decision problems for this group of representatives. This scenario is similar to multi-stakeholder decisions including organ allocation or public policy design: these decisions are made by a small number of representatives (e.g., experts in medicine or policy), who often have very limited time to express their preferences. As in *Population Modeling* we aim to estimate the next vote from a random expert—however in this paradigm, all voters are "known", i.e., in the training data.

Both voting paradigms can be represented as a machine learning problem: observed votes are "data", with which we select a best-fit model from a hypothesis

class; these models make predictions about future votes.[10] Thus, we split all observed votes into a training set (for model fitting) and a test set (for evaluation). How we split votes into a training and test set is important: in *Representative Decisions* we aim to predict future decisions from a *known* pool of voters—so both the training and test set should contain votes from each voter. In *Population Modeling* we aim to predict future decisions from the entire voter population—so the training set should contain only some votes from some voters (i.e., "training" voters), while the test set should contain the remaining votes from training voters, and all responses from the non-training voters.

We propose several group indecision models, each of which is based on the models from Section 13.4; please see Appendix F.2 for more details.

`VMixture` **Model.** We first learn a best-fit indecision (sub)model for each training voter; the overall model generates responses by first selecting a training voter uniformly at random, and then responding according to their submodel.

*k*-`Mixture` **Model.** This model consists of $k$ submodels, each of which is an indecision model with its own utility vector $\boldsymbol{u}$ and threshold $\lambda$. The *type* of each submodel (`Min/Max-`$\delta$, `Min/Max-`$U$, `Dom`) is itself a categorical variable. Weight parameters $\boldsymbol{w} \in \mathbb{R}^k$ indicate the importance of each submodel. This model votes by selecting a submodel from the softmax distribution[11] on $\boldsymbol{w}$, and responds according to the chosen submodel.

*k*-`Min-`$\delta$ **Mixture.** This model is equivalent to *k*-`Mixture`, however all submodels are of type `Min-`$\delta$. We include this model since `Min-`$\delta$ is the most-common best-fit

[10] Several researchers have used techniques from machine learning for social choice [89, 117, 175, 330].

[11] With the softmax distribution, the probability of selecting $i$ is $e^{\boldsymbol{w}_i} / \sum_j e^{\boldsymbol{w}_j}$. We use this distribution for mathematical convenience, though it is straightforward to learn the distribution directly.

| Model Name | *Represenatitives* (20) | | *Population* (100) | |
|---|---|---|---|---|
| | *Indecisive* | *Strict* | *Indecisive* | *Strict* |
| 2-Min-$\delta$ | -0.90/-0.88 | -0.46/-0.47 | -0.87/-0.88 | -0.54/-0.52 |
| 2-Mixture | -0.87/**-0.86** | -0.45/-0.47 | -0.87/-0.88 | -0.53/-0.52 |
| VMixture | -0.92/-0.90 | -0.49/-0.51 | -0.93/-0.94 | -0.57/-0.56 |
| Min-$\delta$ | -0.92/-0.90 | -0.46/-0.48 | -0.87/-0.87 | -0.54/-0.53 |
| Max-$\delta$ | -0.95/-0.90 | -0.45/**-0.46** | -0.96/-0.95 | -0.54/-0.52 |
| Min-$U$ | -0.96/-0.95 | -0.52/-0.54 | -0.98/-0.99 | -0.58/-0.57 |
| Max-$U$ | -0.87/**-0.86** | -0.54/-0.54 | -0.94/-0.94 | -0.58/-0.57 |
| Dom | -1.08/-1.07 | -0.57/-0.58 | -1.05/-1.06 | -0.61/-0.60 |
| MLP | -0.40/-1.55 | -0.15/-0.85 | -0.71/**-0.77** | -0.42/**-0.51** |
| Logit | -0.91/-0.88 | -0.53/-0.54 | -0.93/-0.94 | -0.57/-0.56 |
| Rand | -1.03/-1.00 | N/A | -1.07/-1.07 | N/A |

TABLE 13.2: Average train-set and test-set LL per question (reported as "train/test") for *Representative Decisions* with 20 training voters, (left) and *Population Modeling* with 100 training voters (right), for both the *Indecisive* and *Strict* participant groups. The greatest test-set LL is highlighted for each column. For *Representatives*, the test set includes only votes from the representative voters; for *Population*, the test set includes all voters.

indecision model for individual participants (see § 13.5).

We simulate both the *Population Modeling* and *Representative Decisions* settings using various train/test splits of our survey data. For *Population Modeling* we randomly select 100 training voters; half of each training voter's responses are added to the test set, and half to the training set. All responses from non-training voters are added to the test set.[12]

For *Representative Decisions* we randomly select 20 training voters ("representatives"), and randomly select half of each voter's responses for testing; all other responses are used for training; all non-training voters are ignored.

For both of these settings we fit all mixture models (2-Mixture, 2-Min-$\delta$, and VMixture), each individual indecision model from Section 13.4, and each each baseline model. Table 13.2 shows the training-set and test-set LL for each method, for both voting paradigms. Most indecision models achieve similar test-set LL, with

[12]Each voter in our data answers different questions, so all questions in the test set are "new."

the exception of Dom. In the *Representatives* setting, both mixture models and (non-mixture) indecision models perform well (notably, better than MLP. This is somewhat expected, as the *Representatives* setting uses very little training data, and complex ML approaches such as MLP are prone to overfitting—this is certainly the case in our experiments. In the *Population* setting the mixture models outperform individual indecision models; this is expected, as these mixture models have a strictly larger hypothesis class than any individual model. Unsurprisingly, MLP achieves the greatest test-set LL in the *Population* setting—yet provides no insight as to how these decisions are made.

## 13.6 Discussion

In many cases it is natural to feel indecisive, for example when voting in an election or buying a new car; people are especially indecisive when their choices have moral consequences. Importantly, there are many possible *causes* for indecision, and each conveys different meaning: I may be indecisive when voting for a presidential candidate because I feel unqualified to vote; I may be indecisive when buying a car because all options seem too similar. Using a small study, in Section 13.2 we demonstrate that indecision cannot be interpreted as a "flipping a coin" to decide between alternatives. This violates a key assumption in the technical literature, and it complicates the task of selecting the *best* alternative for an individual or group. Indeed, defining the "best" alternative for indecisive agents depends on what indecision means.

These philosophical and psychological questions have become critical to computer science researchers, since we now use preference modeling and social choice to guide deployed AI systems. The indecision models we develop in Section 13.4

and test in Section 13.5 provide a framework for understanding why people are indecisive—and how indecision may influence expressed preferences when people are allowed to be indecisive (§ 13.4.1), and when they are required to express strict preferences (§ 13.4.1). The datasets collected in Study 1 (§ 13.2) and Study 2 (§ 13.5) provide some insight into the causes for indecision, and we believe other researchers will uncover more insights from this data in the future.

Several questions remain for future work. First, what are the causes for indecision, and what meaning do they convey? This question is well-studied in the philosophy and social science literature, and AI researchers would benefit from interdisciplinary collaboration. Methods for preference elicitation [53] and active learning [135] may be useful here.

Second, if indecision has meaning beyond the desire to "flip a coin", then what is the best outcome for an indecisive agent? ... for a group of indecisive agents? This might be seen as a problem of winner determination, from a perspective of social choice [245].

## 13.7 Authors and Publication

This chapter was written by Duncan C McElfresh, Lok Chan, Kenzie, Doyle, Walter Sinnott-Armstrong, Vincent Conitzer, Jana Schaich Borg, and John P Dickerson; it appeared at AAAI'21 [218].

# Chapter 14: Perceptions of Fairness

## 14.1 Introduction

In this chapter we focus on one aspect of algorithm design: algorithmic *fairness*. In particular, we study whether people *understand* common notions of algorithmic fairness used by AI and ML researchers. Unlike previous chapters, we focus almost entirely on (human) stakeholders, and very little on actual AI or ML systems.

Research into algorithmic fairness has grown in both importance and volume over the past few years, driven in part by the emergence of a grassroots Fairness, Accountability, Transparency, and Ethics (FATE) in Machine Learning (ML) community. Different metrics and approaches to algorithmic fairness have been proposed, many of which are based on prior legal and philosophical concepts, such as disparate impact and disparate treatment [47, 83, 129]. However, definitions of ML fairness do not always align with pre-existing legal and moral frameworks. The rapid expansion of this field makes it difficult for professionals to keep up, let alone the general public. Furthermore, misinformation about notions of fairness can have significant legal implications.[1]

Computer scientists have largely focused on developing mathematical notions of fairness and incorporating them into ML systems. A much smaller collection of studies have measured public perception of bias and (un)fairness in algorithmic decision making. However, as both the academic community and society in general

[1] https://www.cato.org/blog/misleading-veritas-accusation-google-bias-could-result-bad-law

continue to discuss issues of ML fairness, it remains unclear whether non-experts—who will be *impacted* by ML-guided decisions—understand various mathematical definitions of fairness sufficiently to provide opinions and critiques. We emphasize that these technologies are likely to have greater impact on marginalized populations, and those with lower levels of education, as in the case of hiring and criminal justice [37, 136]. For this reason, we focus on a non-expert audience and a context (hiring) that most people would find relatively familiar.

**Contributions.**

- We take a step toward addressing this issue by studying peoples' comprehension and perceptions of three definitions of ML fairness: *demographic parity*, *equal opportunity,* and *equalized odds* [157]. Specifically, we address the following research questions:

  **RQ1** When provided with an explanation intended for a non-technical audience, do non-experts comprehend each definition and its implications?

  **RQ2** What factors play a role in comprehension?

  **RQ3** How are comprehension and sentiment related?

  **RQ4** How do the different definitions compare in terms of comprehension?

- We developed two online surveys to address these research questions. We presented participants with a simplified decision making scenario and an accompanying *fairness rule* expressed in the scenario's context. We asked questions related to the participants' comprehension of and sentiment toward this rule. Tallying the number of correct responses to the comprehension questions gives us a *comprehension score* for each participant.

- In our first study (Study-1), we found that this comprehension score is a consistent and reliable indicator of understanding demographic parity. Further exploratory analysis suggested two additional hypotheses to examine in our second, main study (*Study-2*): that education level is an important predictor for comprehension, and that *negative* sentiment is associated with *greater* comprehension of demographic parity.

- In a second study (Study-2), we used a similar approach to compare comprehension among all three definitions of interest. We find that (1) education is a significant predictor of rule understanding, (2) the counter-intuitive definition of Equal Opportunity with False Negative Rate was significantly harder to understand than other definitions, and (3) participants with low comprehension scores tended to express less negative sentiment toward the fairness rule. This underlines the importance of considering stakeholders before deploying a "fair" ML system, because some stakeholders may not understand or agree with an ML-specific notion of fairness. Our goal is to help to designers and adopters of fairness approaches understand whether they are communicating with stakeholders effectively.

## 14.2 Related Work

In response to many instances of bias in fielded artificial intelligence (AI) and machine learning (ML) systems, ML fairness has received significant attention from the computer science community. Notable examples include gender bias in job-related ads [97], racial bias in evaluating names on resumes [70], and racial bias in predicting criminal recidivism [17]. To correct biased behavior, researchers have proposed several mathematical and algorithmic notions of fairness.

Most algorithmic fairness definitions found in literature are motivated by the philosophical notion of individual fairness (e.g., see [257]), and legal definitions of disparate impact/treatment (e.g., see [37]). Several ML-specific definitions of fairness have been proposed which claim to uphold these philosophical and legal concepts. These definitions of "ML fairness" fall loosely into two categories (for a review, see [84]). *Statistical Parity* posits that in a *fair* outcome, individuals from different protected groups have the same chance of receiving a positive (or negative) outcome. Similarly, *Predictive Parity* [157] asserts that the predictive accuracy should be similar across different protected groups–often measured by the false positive rate (FPR) or false negative rate (FNR) in binary classification settings. Several other definitions have been proposed, based on concepts such as calibration [248] and causality [187]. Of course, all of these definitions make limiting assumptions; no concept of fairness is perfect [157]. The question remains, *which* of these fairness definitions are appropriate, and in *what context?* There are two important components to answering this question: *communicating* these fairness definitions to a general audience, and *measuring their perception* of these definitions in context.

Communicating ML-related concepts is an active and growing research area. In particular, *interpretable ML* focuses on communicating the decision making process and results of ML-based decisions to a general audience [203]. Many tools have been developed to make ML models more interpretable, and many demonstrably improve understanding of ML-based decisions [170, 261]. These models often rely on concepts from probability and statistics—teaching these concepts has long been an active area of research. Batanero et al. [39] provide an overview of teaching probability and how students learn probability; our surveys use their method of communicating probability, which relies on proportions. We draw on several other concepts

from this literature for our study design; for example avoiding numerical and statistical representations [144, 145], which can be confusing to a general audience. Instead we provide relatable examples, accompanied by examples and graphics [167].

Effectively communicating ML concepts is necessary to achieve our second goal of understanding peoples' perceptions of these concepts. One particularly active research area focuses on how people perceive bias in algorithmic systems. For example, Woodruff et al. [321] investigated perceptions of algorithmic bias among marginalized populations, using a focus group-style workshop; Grgic-Hlaca et al. [151] study the underlying factors causing perceptions of bias, highlighting the importance of selecting appropriate features in algorithmic decision making; Plane et al. [246] look at perceptions of discrimination of online advertising; Harrison et al. [161] studies perceptions of fairness in stylized machine learning models; Srivastava et al. [295] note that perceived appropriateness of an ML notion of fairness may depend on the domain in which the decision making system is deployed, but suggest that simpler notions may best capture lay perceptions of fairness.

A related body of work studied how people perceive algorithmic decision makers. Lee [194] studies perceptions of fairness, trust, and emotional response of algorithmic decision makers — as compared to human decision makers. Similar work studies perception of fairness in the context of splitting goods or tasks, and in loan decisions [195, 196, 275]. Binns et al. [48] studies how different explanation styles impact perceptions of algorithmic decision makers.

This substantial body of prior research provided inspiration and guidance for our work. Prior work has studied both the effective communication of, and perceptions of, ML-related concepts. We hypothesize that these concepts are in fact related; to that end, we design experiments to simultaneously study peoples' *comprehension* of and *perceptions* of common ML fairness definitions.

## 14.3 Methods

To study perceptions of ML fairness, we conducted two online surveys where participants were presented with a hypothetical decision making scenario. Participants were then presented with a "rule" for enforcing fairness. We then asked each participant several questions on their comprehension and perceptions of this fairness rule. We first conducted Study-1 to validate our methodology; we then conducted the larger and broader Study-2 to address our main research questions. Both studies were approved by the University of Maryland Institutional Review Board (IRB).

### 14.3.1 Study-1

In Study-1 we tested three different decision making scenarios based on real-world decision problems: hiring, giving employee awards, and judging a student art project. However, we observed no difference in participant responses between these scenarios; for this reason, we focus exclusively on hiring in Study-2 (see 14.3.2). Please see Appendix G.4 for a description of the Study-1 scenarios, and Appendix G.2.5 for relevant survey results. In Study-1, we chose (what we believe is) the simplest definition of ML fairness, namely, demographic parity. In short, this rule requires that the fraction of one group who receives a *positive* outcome (e.g., an award or job offer) is equal for both groups.

#### 14.3.1.1 Survey Design

Here we provide a high-level discussion of the survey design; the full text of each survey can be found in Appendix G.4. The participant first receives a consent form (see Appendix G.5). If consent is obtained, the participant sees a short paragraph explaining the decision making scenario. To make demographic parity accessible to

a non-technical audience, and to avoid bias related to algorithmic decision making, we frame this notion of fairness as a *rule* that the decision maker must follow to be fair. In the hiring scenario, we framed this decision rule as follows: *The fraction of applicants who receive job offers that are female should equal the fraction of applicants that are female. Similarly, the fraction of applicants who receive job offers that are male should equal the fraction of applicants that are male.*

We then ask two questions concerning participant evaluation of the scenario, nine comprehension questions about the fairness rule, two self-report questions on participant understanding and use of the rule, and four free response questions on comprehension and sentiment. For example, one comprehension question is: *Is the following statement TRUE OR FALSE: This hiring rule always allows the hiring manager to send offers exclusively to the most qualified applicants*. Finally, we collect demographic information (age, gender, race/ethnicity, education level, and expertise in a number of relevant fields).

We conducted in-person cognitive interviews [160] to pilot our survey, leading to several improvements in the question design (see Appendix G.1. Most notably, because some cognitive interview participants appeared to use their own personal notions of fairness rather than our provided rule, we added questions to assess this compliance issue.

#### 14.3.1.2 Recruitment and Participants

We recruited participants using the online service Cint [87], which allowed us to loosely approximate the 2017 U.S. Census distributions [68] for ethnicity and education level, allowing for broad representation. We required that participants be 18 years of age or older, and fluent in English. Participants were compensated using Cint's rewards system; according to a Cint representative: "[Participants] can

choose to receive their rewards in cash sent to their bank accounts (e.g. via PayPal), online shopping opportunities with one of multiple online merchants, or donations to a charity."

Data was collected during August 2019. In total 147 participants were included in the Study-1 analysis, including 75 men (51.0%), 71 women (48.3%), and 1 (0.7%) preferring not to answer. The average age was 46 years (SD = 16). Ethnicity and educational attainment are summarized in Table 14.1. On average, participants completed the survey in 14 minutes.

Table 14.1 summarizes the ethnicity and education level of participants in both Study-1 and Study-2.

TABLE 14.1: Participant demographics across ethnicity and education level, compared to the 2017 U.S. Census. AI = American Indian, AN = Alaska Native, NH = Native Hawaiian, PI = Pacific Islander, AA = African American. Note that in Study-2, two participants did not report their education level.

| | Percent of Sample | | |
|---|---|---|---|
| | **Census** | **Study-1** | **Study-2** |
| **Ethnicity** | | | |
| AI or AN | 0.7 | 0.7 | 0.9 |
| Asian or NH or PI | 5.7 | 1.4 | 2.3 |
| Black or AA | 12.3 | 10.2 | 15.8 |
| Hispanic or Latinx | 18.1 | 12.2 | 7.7 |
| Other | 2.6 | 2.7 | 1.4 |
| White | 60.6 | 72.8 | 71.9 |
| **Education Level** | | | |
| Less than HS | 12.1 | 6.1 | 6.9 |
| HS or equivalent | 27.7 | 29.9 | 24.9 |
| Some post-secondary | 30.8 | 30.6 | 24.9 |
| Bachelor's and above | 29.4 | 33.3 | 42.7 |

### 14.3.2 Study-2

Study-2 follows a very similar structure to Study-1 with a few changes. First, we decided to use only the hiring (HR) decision scenario (See Appendix G.2.5 for more

in-depth discussion). Second, we expanded to three definitions of fairness: *demographic parity* (DP), *equal opportunity* (EP), and *equalized odds* (EO) [157]. Within EP, we tested both False Negative Rate (FNR) and False Positive Rate (FPR), resulting in a total of four conditions.

#### 14.3.2.1 Survey Design

Here we provide a high-level discussion of the differences between Study-2 and Study-1; the full text of each survey can be found in Appendix G.4. We used a between-subjects design with random assignment among the four conditions (DP, FNR, FPR, EO). Again, we frame each notion of fairness as a *hiring rule* that the decision maker must follow to be fair. For example, in FPR we define the award rule as follows: *The fraction of unqualified male candidates who receive job offers should equal the fraction of unqualified female candidates who receive job offers.*

For this version, we added graphical examples to further clarify our explanations (see Fig. 14.1 for an example). We used the all the same questions as in Study-1 but added two additional Likert-scale questions assessing participant sentiment: one asked whether they liked the rule, and the other asked whether they agreed with the rule. One free response question (asking how participants personally would go about the hiring process to ensure it was fair), which did not consistently provide useful responses in Study-1, was removed from the Study-2 survey in an effort to keep the expected completion time similar.

#### 14.3.2.2 Recruitment and Participants

We again used the Cint service to recruit participants. Compensation for participation was handled in the same manner as described in §14.3.1.2. Because our initial

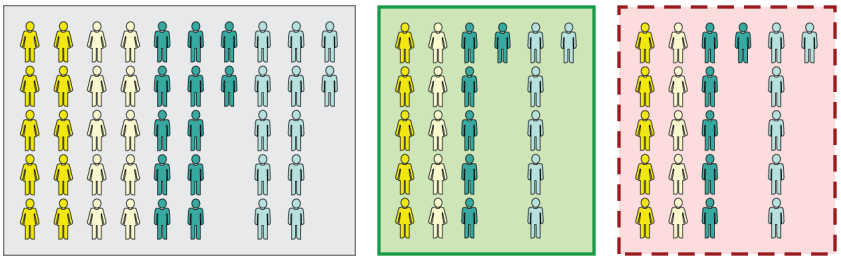


FIGURE 14.1: A graphical example to describe a fair hiring outcome for EO. Yellow people represent females while green people represent males. The darker colors represent qualified individuals while the lighter colors represent unqualified individuals. The gray box represents the original pool of applicants. The green box represent individuals that received job offers while the red box with a dashed border represents individuals that did *not* receive job offers.

sample (intended to target education, ethnicity, gender and age distributions approximating the U.S. census) skewed more highly educated than we had hoped, we added a second round of recruitment one week later primarily targeting participants without bachelor's degrees. Hereafter, we report on both samples together.

Data was collected during January and February 2020. In total 349 participants were included in the Study-2 analysis, including 142 men (40.7%), 203 women (58.2%), 1 other (0.3%), and 3 (0.9%) preferring not to answer. The average age was 45 years (SD=15). Ethnicity and educational attainment are summarized in Table 14.1. On average, participants completed the survey in 16 minutes.

### 14.3.3 Data Analysis

Free response questions were qualitatively coded for statistical testing. In Study-1, one question was coded by a single researcher for simple correctness (see Appendix G.2.1), and the other was independently coded by three researchers (resolved to 100%) to capture sentiment information (see Appendix G.2.3). In Study-2, both questions were independently coded by 2-3 researchers (resolved to 100%). Participants who provided nonsensical answers, answers not in English, or other non-responsive answers to free response questions were excluded from all analysis.

The following methods were used for all statistical analyses unless otherwise specified. Correlations with nonparametric ordinal data were assessed using Spearman's rho. Omnibus comparisons on nonparametric ordinal data were performed with a Kruskal–Wallis (K-W) test, and relevant post-hoc comparisons with Mann–Whitney U (M-WU) tests. Post-hoc $p$-values were adjusted for multiple comparisons using Bonferroni correction. $\chi^2$ tests were used for comparisons of nominal data. Box plots show median and first and third quartiles; whiskers extend to $1.5 * \text{IQR}$ (interquartile range), with outliers indicated by points. The full analysis script for both studies can be found on GitHub. [2]

### 14.3.4 Limitations

As with all surveys, our study has certain limitations. We recruited a demographically broad population, but web panels are generally more tech-savvy than the broader population [258]. We consider this acceptable for a first effort. Some participants may be "satisficing" rather than answering carefully. We mitigate this by disqualifying participants with off-topic or non-responsive free-text responses. Further, this limitation can be expected to be consistent across conditions, enabling reasonable comparison. Finally, better or clearer explanations of the fairness definitions we explored are certainly possible; we believe our explanations were sufficient to allow us to investigate our research questions, especially because they were designed to be consistent across conditions.

[2] https://github.com/saharaja/ICML2020-fairness

## 14.4 Results

In this section we first discuss the preliminary findings from Study-1 (see §14.4.1). These findings were used as hypotheses for further exploration and testing in Study-2; we discuss those results second (see §14.4.2).

### 14.4.1 Study-1

We analyze survey responses for Study-1 and make several observations. We first validate our comprehension score as a measure of participant understanding; we then generate hypotheses for further exploration in Study-2.

#### 14.4.1.1 Our Survey Effectively Captures Rule Comprehension

We find that we can measure comprehension of the fairness rule. The comprehension score was calculated as the total correct responses out of a possible 9. All questions were weighted equally. The relevant questions included 2 multiple choice, 4 true/false, and 3 yes/no questions. The average score was 6.2 (SD=2.3).

We validate our comprehension score using two methods: internal validity testing, and correlation against two self-report and one free response question included in our survey (see Appendix G.2.1 for further details).

**Internal Validity** Cronbach's $\alpha$ and item-total correlation were used to assess internal validity of the comprehension score. Both measures met established thresholds [124, 230]: Cronbach's $\alpha = 0.71$, and item-total correlation for 8 of the 9 items (all but Q5) $> 0.3$.

**Question Correlation** We find that self-reported rule understanding and use are reflected in comprehension score. First, we compared comprehension score to self-reported rule understanding (Q13): "I am confident I know how to apply the award rule described above," rated on a five-point Likert scale from strongly agree (1) to strongly disagree (5). The median response was "agree" (Q1 $= 1$, Q3 $= 3$). Higher comprehension scores tended to be associated with greater confidence in understanding (Spearman's $\rho = 0.39$, $p < 0.001$), supporting the notion that comprehension score is a valid measure of rule comprehension.

Next, we compared comprehension score to a self-report question about the participant's use of the rule (Q14), with the following options: (a) "I applied the provided award rule only," (b) "I used my own ideas of what the correct award decision should be rather than the provided award rule," or (c) "I used a combination of the provided award rule and my own ideas of what the correct award decision should be." We find that participants who claimed to use only the rule scored significantly higher (mean 7.09) than those who used their own notions (4.90) or a combination (4.68) (post-hoc M-WU, $p < 0.001$ for both tests; corrected $\alpha = 0.05/3 = 0.017$). This further corroborates our comprehension score.

Finally, we asked participants to explain the rule in their own words (Q12). Each response was then qualitatively coded as one of five categories—**Correct**: describes rule correctly; **Partially correct**: description has some errors or is somewhat vague; **Neither**: vague description of purpose of the rule rather than how it works, or pure opinion; **Incorrect**: incorrect or irrelevant; and **None**: no answer, or expresses confusion. Participants whose responses were either correct (mean comprehension score = 7.71) or partially correct (7.03) performed significantly better on our survey than those responding with neither (5.13) or incorrect (4.24) (post-hoc M-WU, $p < 0.001$

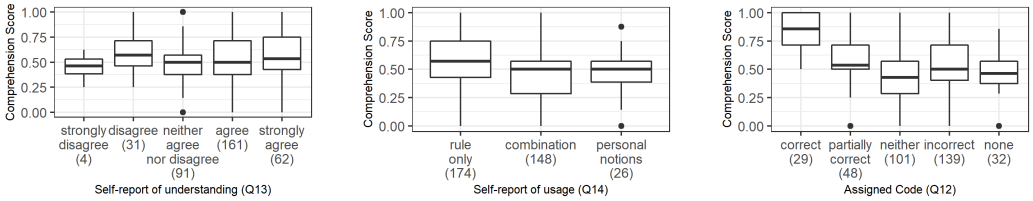
for these four comparisons, corrected $\alpha = 0.05/10 = 0.005$). These findings further validate our comprehension score. Additional details of these results and the associated statistical tests can be found in Appendix G.2.1.

#### 14.4.1.2 Hypotheses Generated

We analyzed the data from Study-1 in an exploratory fashion intended to generate hypotheses that could be tested in Study-2. We highlight here three key hypotheses that emerged from the data.

**Education Influences Comprehension** We used Poisson regression models to explore whether various demographic factors were associated with differences in comprehension. We found that a model including education as a regressor had greater explanatory power than a model without (see Appendix G.2.2 for further details).

**Disagreement with the Rule is Associated with Higher Comprehension Scores** We asked participants for their opinion on the presented rule in a free response question (Q15). These responses were qualitatively coded to capture participant sentiment toward the rule in one of five categories – **Agree**: generally positive sentiment towards rule; **Depends**: describes both pros and cons of the given rule; **Disagree**: generally negative sentiment towards rule; **Not understood**: expresses confusion about rule; **None**: no answer, or lacks opinion on appropriateness of the rule. Participants who expressed disagreement with the rule performed better (mean comprehension score = 7.02) than those who expressed agreement (5.50), did not understand the rule (4.44), or provided no response (5.09) to the question (post-hoc M-WU, $p < 0.005$ for these three comparisons; corrected $\alpha = 0.05/10 = 0.005$). Appendix G.2.3 provides further details.




(A) Grouped by response to Q13

(B) Grouped by response to Q14.

(C) Grouped by coded response to Q12.

FIGURE 14.2: Comprehension scores grouped by questions. In (a), self-reported understanding of the rule was not related to comprehension score. X-axis is reversed for figure and correlation test. In (b), rule compliance (leftmost on the x-axis) was associated with higher comprehension scores. One participant who did not provide a response was excluded from this figure and the relevant analysis. Finally, in (c), participants who provided either correct or partially correct responses tended to perform better.

**Non-Compliance is Associated with Lack of Understanding** We were interested in understanding why some participants failed to adhere to the rule, as measured by their self-report of rule usage in Q14. We labeled those who responded with either having used their own personal notions of fairness ($n = 29$) or some combination of their personal notions and the rule ($n = 28$) as “non-compliant" (NC), with the remaining $n = 89$ labeled as “compliant" (C). One participant who did not provide a response was excluded from this analysis, conducted using $\chi^2$ tests.

Non-compliant participants were less likely to self-report high understanding of the rule in Q13 (see Fig. G.7). Moreover, non-compliance also appears to be associated with a reduced ability to correctly explain the rule in Q12 (see Fig. G.8). This fits with the overall strong relationship we observed among comprehension scores, self-reported understanding, ability to explain the rule, and compliance.

Further, negative participant sentiment towards the rule (Q15) also appears to be associated with greater compliance (see Fig. G.9). Thus, non-compliant participants appear to behave this way because they do not *understand* the rule, rather than because they do not *like* it. Refer to Appendix G.2.4 for further details.

### 14.4.2 Study-2

We first confirm the validity of our comprehension score, then compare comprehension across definitions and examine the hypotheses generated in Study-1.

#### 14.4.2.1 Score Validation

We validated our metric using the same approach used in Study-1, i.e., assessing both internal validity and correlation with self-report and free-response questions. We report the results of this assessment here.

**Internal Validity** We again used Cronbach's $\alpha$ and item-total correlation to assess internal validity of the comprehension score. An initial assessment using all 349 responses yielded Cronbach's $\alpha = 0.38$, and item-total correlation $> 0.3$ for only four of the nine comprehension questions. Since both measures performed below established thresholds [124, 230], we investigated further and repeated these measurements individually for each fairness-definition condition (DP, FNR, FPR, EO). This procedure showed stark differences in Cronbach's $\alpha$ based on definition: DP=0.64, FNR=0.39, FPR=0.49, EO=0.62. Item-total correlations followed a similar pattern: best in DP, worst in FNR. Based on these differences, we iteratively removed problematic questions from the score on a per-definition basis until all remaining questions achieved an item-total correlation of $> 0.3$ [124]. By removing poorly performing questions, we increase our confidence that the measured comprehension scores are meaningful for further analysis. Table 14.2 specifies which questions were retained for analysis in each definition.

Because questions were dropped on a per-definition basis, the maximum of the resulting scores varied from 4-8 depending on the definition, rather than being a uniform 9. We normalized this treating comprehension score as a percentage of the

TABLE 14.2: Questions that were used for downstream analysis after iterative removal of questions with poor item-total correlation.

| | **Questions** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Q3 | Q4 | Q5 | Q6 | Q7 | Q8 | Q9 | Q10 | Q11 |
| DP | X | X | | | X | X | X | X | X |
| FNR | X | X | X | | | X | | | |
| FPR | X | X | X | X | | X | | X | X |
| EO | X | X | X | | X | X | X | X | X |

maximum for each condition rather than a raw score. We report this *adjusted score* in the remainder of §14.4.2. The average score was 0.53 (SD=0.22).

**Question Correlation** As in Study-1, we compare comprehension scores with responses to self-report and free response questions included in our survey.

First, we compared comprehension score to self-reported rule understanding (Q13), as described in §14.4.1.1. The median response was "agree" ($Q1 = 2, Q3 = 3$). We assess the correlation between these responses and comprehension score using Spearman's rho (appropriate for ordinal data). Unlike in Study-1, there was no relationship between self-reported understanding and comprehension score (Fig. 14.2a).

Next, we compared comprehension score to a self-report question about the participant's use of the rule (Q14), as described in §14.4.1.1. A K-W test revealed a relationship between self-reported rule usage and comprehension score ($p < 0.001$). We find that participants who claimed to use only the rule tended to score higher (mean comprehension score = 0.58) than those who used a combination of the rule and their own notions of fairness (0.47, $p < 0.01$). No other differences were found (post-hoc M-WU; corrected $\alpha = 0.05/3 = 0.017$). This suggests that participants are answering at least somewhat honestly: when they try to apply the rule, comprehension scores improve (see Fig. 14.2b).

Finally, we asked participants to explain the rule in their own words (Q12). Each

response was then qualitatively coded as one of five categories, as described in §14.4.1.1. These results can be seen in Fig. 14.2c. A K-W test revealed a relationship between comprehension score and coded responses to Q12 ($p < 0.001$). Correct (mean comprehension score = 0.83) responses were associated with higher comprehension scores than partially correct (0.58), neither (0.44), incorrect (0.52), and none (0.48) responses ($p < 0.001$ for all); partially correct responses were also associated with higher comprehension scores than neither responses ($p < 0.001$); and incorrect responses were associated with higher comprehension scores that neither responses ($p < 0.005$). No other differences were found (post-hoc M-WU; corrected $\alpha = 0.05/10 = 0.005$). These findings support our claim that our comprehension score is a valid measure of fairness-rule comprehension.

#### 14.4.2.2 Education and Definition are Related to Comprehension Score

One hypothesis generated by Study-1 was that comprehension score is positively correlated with education level. We investigated this hypothesis further in Study-2 using linear regression models followed by model selection. We believe this exploratory approach to be appropriate despite the previously formulated hypothesis, given the introduction of a new variable in Study-2, i.e., fairness definition.

Eleven models were tested, regressing different combinations of demographics (ethnicity, gender, education, and age) and condition (fairness definition). Models were compared using Akaike information criterion (AIC), a standard method of evaluating model quality and performing model selection [9]. Comparison by AIC revealed that the model using just education (edu) and fairness definition (def) as regressors was the model of best fit. In this model, having a Bachelor's degree or above resulted in a score increase of 0.14, and the FNR condition caused a score decrease of -0.11 ($p < 0.004$ for both; corrected $\alpha = 0.05/11 = 0.0045$). A regression

table of the best fit model can be found in Table 14.3.

TABLE 14.3: Regression table for the best fit model, with two covariates: education (baseline: no HS) and definition (baseline: DP). Est. = estimate, CI = confidence interval.

| **Covariate** | **Est.** | **95% CI** | $p$ |
|---|---|---|---|
| *Education* | | | |
| HS | 0.00 | [-0.10, 0.10] | 0.989 |
| Post-secondary, no BS | 0.09 | [-0.01, 0.18] | 0.078 |
| Bachelor's and above | 0.14 | [0.04, 0.23] | $< 0.004$ |
| *Definition* | | | |
| EO | -0.08 | [-0.14, 0.01] | 0.020 |
| FPR | -0.05 | [-0.11, 0.01] | 0.124 |
| FNR | -0.11 | [-0.18, -0.05] | $< 0.001$ |

AIC results of each of the eleven models, along with the relevant regressors, can be seen in Table G.1 in Appendix G.3.1. Comprehension score as a function of education and fairness definition can be seen in Figs. 14.3 and 14.4.

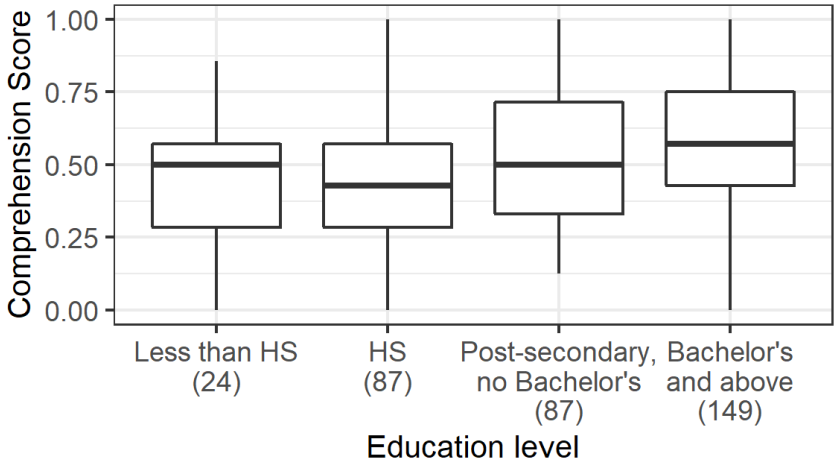



FIGURE 14.3: Comprehension score grouped by education level. Higher education was associated with higher comprehension scores. Note that two participants who did not report their education level were removed from this figure and the relevant analysis.

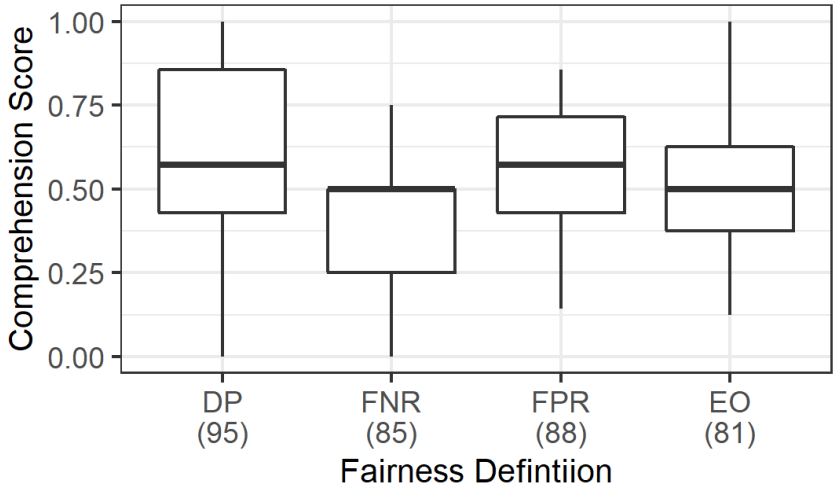



FIGURE 14.4: Comprehension score grouped by fairness definition. The FNR condition was associated with lower comprehension sore.

#### 14.4.2.3 Greater Negative Sentiment Toward the Rule is Associated with Higher Comprehension Scores

In Study-1, we found a relationship between participant sentiment towards the rule and comprehension score. To better interrogate this phenomenon, in Study-2 we added two more questions to the survey to directly address the issue of sentiment, rather than relying on a free-response question. One (Q15) asks, “To what extent do you agree with the following statement: I like the hiring rule?", and is evaluated on a five-point Likert scale from “strongly agree" (1) to “strongly disagree" (5). The other (Q16) asks, “To what extent do you agree with the following statement: I agree with the hiring rule?", and is also evaluated on a five-point Likert scale from “strongly agree" (1) to “strongly disagree" (5).

Using Spearman’s rho, we assessed the correlation between responses to these two questions and comprehension score. A minor correlation was found between liking the rule and comprehension score, i.e., those who disliked the rule were more likely to have higher comprehension scores ($\rho = -0.11, p < 0.05$; see Fig. 14.5).

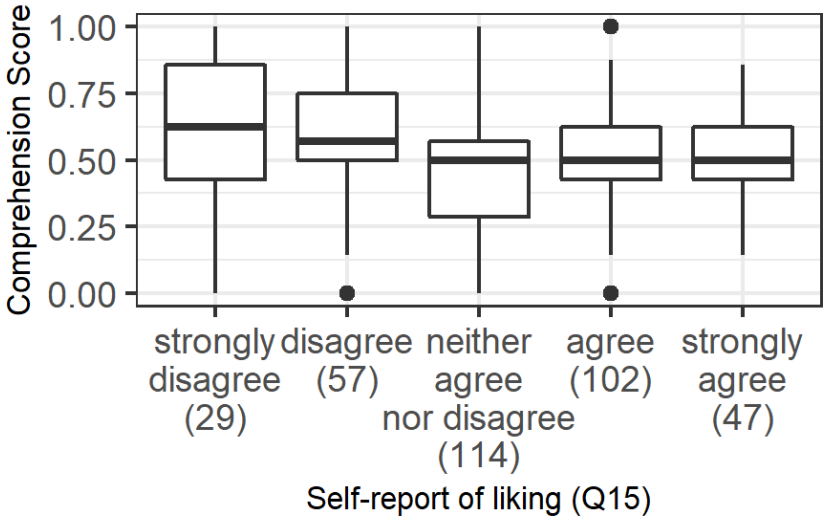



FIGURE 14.5: Comprehension score grouped by response to Q15. Dislike of the rule was associated with higher comprehension scores. X-axis is reversed for figure and correlation test.

A slight correlation was also found between agreeing with the rule and comprehension score, i.e., disagreement was associated with higher comprehension scores ($\rho = -0.11, p < 0.05$; see Fig. 14.6).

#### 14.4.2.4 Non-Compliance is Associated with Lack of Understanding

A final hypothesis generated in Study-1 involves non-compliance: i.e., why do participants who report *not* using the rule to answer the comprehension questions behave this way? In Study-1, we found that this was due to the fact that non-compliant participants were less able to *understand* the rule, rather than because they did not *like* it. We also observed this in our results from Study-2: compliant participants exhibited higher self-reported understanding of the rule ($p < 0.001$, Fig. G.11), were more likely to correctly explain the rule ($p < 0.001$, Fig. G.12), and were more likely to dislike the rule ($p < 0.05$, Fig. G.13). We observed no relationship between compliance and agreement with the rule (Fig. G.14). Refer to Appendix G.3.2 for more details.

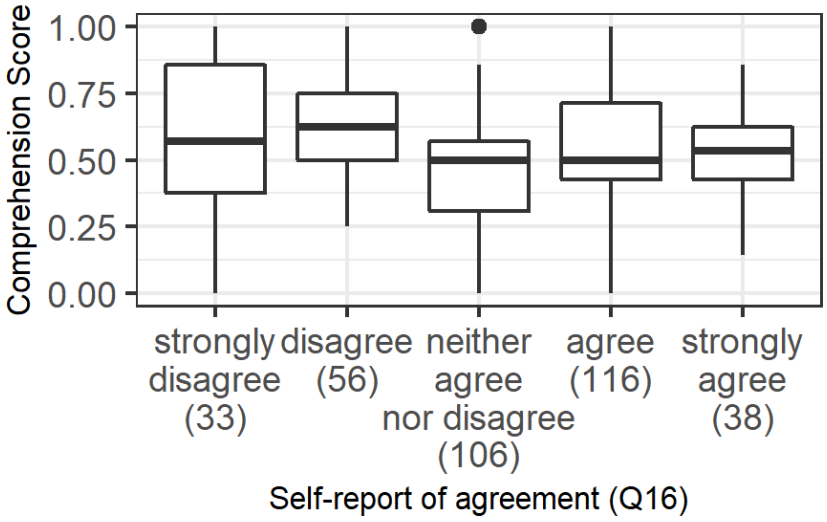



FIGURE 14.6: Comprehension score grouped by response to Q16. Disagreement with the rule was associated with higher comprehension score. X-axis is reversed for figure and correlation test.

## 14.5 Discussion

Bias in machine learning is a growing threat to justice; to date, ML bias has been documented in both commercial and government applications, in sectors such as medicine, criminal justice, and employment. In response, ML researchers have proposed various notions of *fairness* to correct these biases. Most ML fairness definitions are purely mathematical, and require some knowledge of machine learning. While they are intended to benefit the general public, it is unclear whether the general public agrees with — or even understands — these notions of ML fairness.

We take an initial step to bridge this gap by asking *do people understand the notions of fairness put forth by ML researchers?* To answer this question we develop a short questionnaire to assess understanding of three particular notions of ML fairness (demographic parity, equal opportunity, and equalized odds). We find that our comprehension score (with some adjustments for each definition) appears to be a consistent and reliable indicator of understanding the fairness metrics. The comprehension score demonstrated in this work lays a foundation for many future studies

exploring other fairness definitions.

We do find, however, that comprehension is lower for equal opportunity, false negative rate than other definitions. In general, comprehension scores for equal opportunity (both FNR and FPR) were less internally consistent than other fairness rules, suggesting participant responses were also more "noisy" for equal opportunity. This is somewhat intuitive: equal opportunity is difficult to understand, as it only involves one type of error (FNR or FPR) rather than both. Furthermore, FNR participants had the lowest comprehension scores *and* the lowest consistency of all conditions. We believe this finding also matches intuition: FNR is a strange notion in the context of hiring, as it concerns only those qualified applicants who were *not* hired or offered jobs. Indeed, in free-response questions several participants mentioned that they do not understand why qualified candidates are *not* hired. We believe many participants fixated on this strange setting, impacting their comprehension scores. This finding is potentially problematic, as equal opportunity definitions are increasingly used in practice. Indeed, major fairness tools such as Google What-If tool [318] and the IBM AI Fairness 360 [40] specifically focus on equal opportunity. Further work should be put into making descriptions of nuanced fairness metrics more accessible.

Our analysis also identified other issues that should be considered when thinking about mathematical notions of fairness. First, we find that education is a strong predictor of comprehension. This is especially troubling, as the negative impacts of biased ML are expected to disproportionately impact the most marginalized [37] and displace employment opportunities for those with the least education [136]. Lack of understanding may hamper these groups' ability to effectively advocate for themselves. Designing more accessible explanations of fairness should be a top research priority.

Second, we find that those with the weakest comprehension of fairness metrics also express the least negative sentiment toward them. When fairness is a concern, there are always trade-offs—between accuracy and equity, or between different stakeholders, and so on. Balancing these trade-offs is an uncomfortable dilemma often lacking an objectively correct solution. It is possible that those who comprehend this dilemma *also* recognize the precarious trade-off struck by any mathematical definition of fairness, and are therefore dissatisfied with it. It is also possible that participants with positive sentiment toward fairness definitions are genuinely appreciative of fairness in general—and since education is correlated with comprehension, these participants may be less well-off than participants with high comprehension scores. From another perspective, this finding is more insidious. If those with the weakest understanding of AI bias are also least likely to protest, then major problems in algorithmic fairness may remain uncorrected.

## 14.6 Authors and Publication

This chapter was written by Debjani Saha, Candice Schumann, Duncan C McElfresh, Michelle Mazurek, and Michael Tschantz; it appeared at ICML'20 [272].

# Part IV

# Conclusions & Future Research

# Chapter 15: Conclusion

Using algorithms for resource allocation and decision making requires a careful balance of priorities. On one hand, advanced algorithms can identify ways to use resources far more efficiently, and far quicker, than any human (or team of humans) could hope for. In Part I of this thesis I describe the algorithms behind one such application—kidney exchange—where algorithms have been used to great effect for more than a decade. Similar methods are used regularly in transportation, supply chain management, medical resource allocation, and finance. Algorithms are also widely used to help people make better decisions: perhaps the most common example of this is recommender systems, which are used widely in retail and marketing applications to suggest products that are relevant to consumers; similar methods are used in social media to recommend content to users. Decision support algorithms are used in medicine, for example to identify an appropriate diagnosis and treatment plan. In each of these examples, algorithms have demonstrated value—which is further confirmed by their increasing adoption.

Furthermore, algorithms lend stability and credibility to a decision process. Many fielded algorithms are deterministic—meaning that the same input will always produce the same output. This consistency is especially important in high-stakes scenarios, for example in kidney exchange, where people tend to avoid uncertainty and ambiguity [71]. In some cases algorithms are even seen as *more* trustworthy or objective than humans [194].

On the other hand, the use of algorithms can complicate things. Algorithms often assist or replace a human decision maker, and designing an algorithm requires that we explicitly state how decisions *ought* to be made. This requires a level of reflection that isn't always necessary when a decision is made only by humans. Furthermore, algorithms are often based on a variety of *assumptions* about the world—assumptions about human behavior, user priorities, data availability and accuracy, available computing power, and so on. Each of these *design considerations* raise important questions for algorithm designers, as well as stakeholders. Here I briefly outline some important algorithm design considerations, many of which are covered in this thesis.

**Uncertainty** When algorithms are deployed in unpredictable environments, we guide them with our beliefs about the future; when these beliefs are wrong, algorithms can perform very poorly. This *uncertainty* is a problem in deployed algorithms; Chapter 3 presents a "robust" (risk-averse) algorithm to deal with uncertainty in kidney exchange algorithms; Chapters 4 and 5 present data-driven approaches with adjustable levels of risk aversion. These optimization-based approaches perform well in computational simulations, however they are very sensitive to our characterization of uncertainty—such as the distribution or range of an uncertain parameter. Furthermore, optimization-based approaches tend to require modifying or replacing an existing algorithm; in many applied contexts, including kidney exchange, this is impractical. Deployed algorithms are often a small part of a much larger ecosystem governed by local law, organizational policies, and oversight boards. Changing the nature of a deployed algorithm can incur enormous costs; in kidney exchange this can require months or years of deliberation. However even if a deployed algorithm cannot be modified, we can still improve its performance

by changing its *use*. We take this approach in Chapter 8 to address uncertainty in kidney exchange by eliciting information from certain participants in the exchange. Certain kidney recipients can have an especially large impact on the behavior of a kidney exchange matching algorithm; if these patients *refuse* a certain donor, this can alter the entire structure of the exchange. We develop methods for selectively gathering information from these recipients; this information can substantially improve the exchange outcome, without modifying the deployed matching algorithm.

**Transparency** In high-stakes applications such as medicine or finance, it is usually important to justify how decisions are made: What assumptions were made? What evidence was used? What alternatives were considered? This justification becomes murky when an algorithm is involved. If an algorithm *makes* a decision, it is important to characterize the reasoning and historical data that resulted in the decision. If an algorithm *helps* a person make a decision, it is also important to characterize the algorithm's role in the decision process. These questions relate to the *transparency* of an algorithm and the decision process it participates in. Complex ML and AI algorithms can be especially difficult to understand, even for experts. The nascent field of eXplainable Artificial Intelligence (XAI) endeavors to build transparent algorithms—both by designing algorithms that are *inherently* easy to understand, and by *explaining* algorithm output using post-hoc analysis. I argue in Chapter 10 that existing XAI approaches are too technocentric completely address issues of algorithmic transparency, and that stakeholders should play a central role in the development of algorithmic systems. Chapter 12 describes one way to engage more directly with stakeholders, using their feedback to generate useful explanations of ML models.

**Social Bias** In recent years, algorithmic *bias* has been an important topic both inside and outside of academia. Deployed algorithms have been shown to mimic, and even amplify, dangerous social biases—most notably racism and sexism [18, 67, 67, 96, 231]. Algorithmic bias is closely related to algorithmic transparency—since it is more difficult to detect potential biases from complex (or "opaque") algorithms. Myriad "debiasing" methods have been proposed in the AI and ML literature, though there is currently no standard of practice. Furthermore, mitigating one type of algorithmic bias can perpetuate another type of bias [182], and certain debiasing measures can also impact privacy [256].

**User Behavior** No matter how well-designed an algorithm is, we cannot predict how it will be used "in the wild." How someone uses an algorithmic system depends on a variety of factors: the context in which the system is used, how much experience the person has, their prior experience with similar systems, their preconceived notions of AI and computers, and so on. It is essential for computer scientists to consider both the users of their products, and the context in which they are used. Part III of this thesis addresses several user-specific challenges to algorithm design: Chapter 11 studies how AI suggestions influence decision making, Chapter 13 highlights the importance of *indecision* in behavior models, and Chapter 14 measures whether people understand common notions of algorithmic fairness.

Designing an algorithm to make important decisions is a juggling act. Increasing an algorithm's performance can require making it less transparent, or more difficult to use; removing one type of social bias can provoke other biases, and can impact user privacy. Striking the *appropriate* balance of priorities is a matter of deliberation—which should center on the stakeholders and users of a system.

# Chapter 16: Future Work

## 16.1 Kidney Exchange

**RQ1: Online Kidney Exchange** Researchers often treat kidney exchange as a *static* problem: the patient-donor pool is considered to be constant (i.e., patient-donor pairs do not enter or exit the pool while a matching is being constructed), and matching policies typically consider only the *current* exchange pool, and not future pools. However kidney exchange is an inherently dynamic process: patient-donor pairs and NDDs are constantly entering and exiting the exchange, and today's matching inevitably impacts tomorrow's exchange pool. Research has demonstrated that matching policies designed for a dynamic environment can increase overall welfare [30, 104, 105, 307], yet there are both theoretical and practical challenges that must be overcome. First, it is theoretically challenging to design policies for an online environment, especially when there is substantial uncertainty about the future. An online policy that adapts its behavior based on the exchange pool might also give policymakers pause—since its behavior might not be intuitive. Future work should explore dynamic policies that are *transparent* to policymakers: there is both a legal and ethical obligation to develop transparent policies in the context of kidney exchange [239].

## 16.2 Preferences & Social Choice

**RQ2: Learning Preferences over Sets** Stakeholder preference play a major role in many modern algorithmic systems. A canonical example is recommender systems, which are commonplace in marketing, online commerce, and social media applications; similar systems have been proposed for high-stakes decisions such as kidney exchange [133] and self-driving cars [228]. In resource allocation settings such as kidney exchange, the "outcome" is usually a large or complex set; in kidney exchange, the outcome is a "matching", or a cycle packing on a directed graph, which represents a set of potential transplants in the exchange. To understand how stakeholders value different kidney exchange outcomes, we need to understand how they value different *sets* of transplants; this raises both mathematical and moral challenges. From a moral perspective, comparing kidney exchange matchings is similar to a slightly lower-stakes version of the trolley problem: during each match run, exchanges choose a set of patients to receive transplants immediately (or very soon after matching), and all other patients in the exchange must wait for the next match run—which means they face several more weeks of dialysis, at least. These dilemmas are not straightforward, especially when the size of a matching can vary. For example: when is it reasonable to choose a *smaller* matching, when a larger matching is possible? From a mathematical perspective, preferences over sets raise modeling challenges. Most preference models assume that people have preferences over individual "items"—for example, they prefer Product A to Product B, or they prefer that Patient A receives priority for a kidney transplant over Patient B [133]. Some preference models (and aggregation techniques) over sets have been studied, primarily in the context of social choice [36, 59, 62, 65]. There are still many open theoretical

and practical questions here: for example how to learn and aggregate[1] these preferences efficiently, and how to design a relatively transparent aggregation method for resource allocation (e.g., for kidney exchange).

**RQ3: Dealing with Indecision** Chapter 13 highlights the importance of *indecision* in human decision making, and we propose several preference models for indecisive agents. There are both theoretical and empirical questions raised in this work. The largest theoretical question is one of social choice: if some agents are indecisive, how do we define consensus? Does this depend on the cause or indecision? Some of these questions have been studied for incomparable items and incomplete preferences [122, 192, 244], but not indecision. From an empirical perspective, further studies are needed to understand the nature and causes of indecision. For example, can indecision be resolved through deliberation or information sharing? Do people wish to resolve their indecision? What scenarios are likely to evoke indecision?

## 16.3 Participatory Algorithm Design

**RQ4: Facilitating Collaboration between Stakeholders and Technicians** A fundamental challenge in Participatory Algorithm Design (PAD) (see Chapter 10) is the knowledge gap between stakeholders and technicians (such as computer scientists and engineers). As we find in Chapter 14, many people do not understand algorithmic concepts—even the most "basic" notions of fairness used by computer scientists. To facilitate PAD, we need to both *learn* what is important to both stakeholders and technicians, and to *convey* these concepts to both groups; this presents a communication problem. Here we might draw inspiration from the practice of *Shared Decision Making* [121] (SDM) in medicine—a process where physicians work closely with

[1]This question is related to the study of *committee elections* and *multiwinner elections* in social choice.

their patient to decide the most appropriate course of treatment. In SDM, patients and their physicians discuss treatment options, the patient's preferences, and the likelihood of different outcomes. The challenge of SDM is that patients know their own needs and values, but they're not always medical experts; on the other hand, medical staff know about treatment options, but they don't always know what their patients want. In this way SDM closely parallels PAD: patients in SDM are analogous to stakeholders (who know what they need, but are not algorithm experts), while medical staff are analogous to computer scientists. Many techniques developed for SDM are readily applicable to PAD, e.g. for measuring patient conflict and decision uncertainty [232], or for measuring mutual understanding between doctors and their patients [159].

# Appendix A: Appendix to Chapter 3

## A.1 Edge Weight Robust Formulation

We develop an edge weight robust formulation with uncertainty set $\mathcal{U}_\Gamma^I$, based on the position-indexed chain-edge formulation formulation (PICEF) introduced by Dickerson et al. [109]. In Section A.1.1 we review the PICEF formulation, and in Section A.1.2 we introduce our linear formulation for edge-weight robust kidney exchange.

Section A.1.3 and Section A.1.4 describe the solution methods for constant uncertainty budget $\Gamma$ and variable uncertainty budget $\gamma(|\mathbf{x}|)$ for decision variables $\mathbf{x}$, respectively.

For simplicity, we use the abbreviation $KEX(\mathcal{U})$ to refer to the robust kidney exchange problem, with uncertainty set $\mathcal{U}$.

### A.1.1 PICEF Formulation

The position-indexed chain-edge formulation (PICEF) is a compact formulation proposed by Dickerson et al. [109], with a polynomial (with regard to the compatibility graph size and exogenous cycle cap) count of both variables and constraints. This formulation 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 NDDs $N$

- $C$: a set of cycles on exchange graph $G$
- $L$: chain cap (maximum number of edges used in a chain)
- $w_e$: edge weights for each edge $e \in E$
- $w_c^C$: cycle weights for each cycle $c \in C$, defined as $w_c^C = \sum_{e \in c} w_e$

This formulation uses one decision variable for each cycle, and several decision variables for each edge to represent chains:

- $z_c \in \{0, 1\}$: 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

Note that edges between an NDD $n \in N$ and a patient-donor vertex $v \in P$ may only take position 1 in a chain, while edges between two patient-donor pairs may take any position $1, 2, \ldots, L$ 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.

$$\mathcal{K}(e) = \begin{cases} \{1\} & e \text{ begins in } n \in N \\ \{1, 2, \ldots, L\} & e \text{ begins in } v \in P \end{cases}$$

We also use the following notation for flow into and out of vertices:

- $\delta^-(s)$ and $\delta^-(S)$: the set of edges into vertex $s$ or set of vertices $S$
- $\delta^+(s)$ and $\delta^+(S)$: the set of edges out of vertex $s$ or set of vertices $S$

The PICEF formulation is given in Problem A.1.

$$\max \quad \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e y_{ek} + \sum_{c \in C} w_c z_c \tag{A.1a}$$

$$\text{s.t.} \quad \sum_{e \in \delta^-(i)} \sum_{k \in \mathcal{K}(e)} y_{ek} + \sum_{c \in C : i \in c} z_c \leq 1 \qquad i \in P \tag{A.1b}$$

$$\sum_{e \in \delta^+(i)} y_{e1} \leq 1 \qquad i \in N \tag{A.1c}$$

$$\sum_{\substack{e \in \delta^-(i) \wedge \\ k \in \mathcal{K}(e)}} y_{ek} \geq \sum_{e \in \delta^+(i)} y_{e,k+1} \qquad \begin{array}{l} i \in P \\ k \in \{1, \ldots, L-1\} \end{array} \tag{A.1d}$$

$$y_{ek} \in \{0, 1\} \qquad e \in E, k \in \mathcal{K}(e) \tag{A.1e}$$

$$z_c \in \{0, 1\} \qquad c \in C \tag{A.1f}$$

The Objective (A.1a) maximizes the total weight of a matching, defined by the cycle decision variables $z_c$ and edge variables $y_{ek}$. Feasible matchings may only use each edge once, and must contain valid chains. Capacity constraints ensure that each edge is used at most once:

The capacity constraints for each vertex are as follows:

- Constraint A.1b: each patient-donor vertex $i \in P$ may only participate in one cycle or one chain
- Constraint A.1c: each NDD $i \in N$ may only participate in one chain

Valid chains must begin in an NDD, and conserve flow through patient-donor pairs:

- Constraint A.1d: a 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 $k$

In the next section we present the mixed integer linear program formulation for $KEX(\mathcal{U}_\Gamma^I)$, based on PICEF.

### A.1.2 Our Robust Formulation

To simplify notation, let $\mathcal{M}^P$ be the set of all feasible matchings for the PICEF formulation. The edge weight robust kidney exchange problem $KEX(\mathcal{U}_\Gamma^I)$ is given in Equation A.2.

$$\max \min_{\mathbf{w} \in \mathcal{U}_\Gamma^I} \quad \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e y_{ek} + \sum_{c \in C} w_c z_c \tag{A.2a}$$

$$\text{s.t.} \quad (\mathbf{z}, \mathbf{y}) \in \mathcal{M}^P \tag{A.2b}$$

Proposition A.1 states that this problem is identical to the robust formulation with one-sided uncertainty set $\mathcal{U}_\Gamma^{I1}$—that is, $KEX(\mathcal{U}_\Gamma^I) = KEX(\mathcal{U}_\Gamma^{I1})$.

**Proposition A.1.** *The problems $KEX(\mathcal{U}_\Gamma^I)$ and $KEX(\mathcal{U}_\Gamma^{I1})$ are equivalent.*

*Proof.* In $KEX(\mathcal{U}_\Gamma^I)$ (Problem A.2), edge weights are minimized with respect to uncertainty set $\mathcal{U}_\Gamma^I$. The objective is minimized when up to $\Gamma$ edge weights are reduced by the maximum amount within $\mathcal{U}_\Gamma^I$ ($d_e$), and one edge weight is reduced by $(\Gamma - \lfloor \Gamma \rfloor) d_e$ . That is, $KEX(\mathcal{U}_\Gamma^I)$ only considers realized edge weights on the interval $\hat{w}_e \in [w_e - d_e, w_e]$. This is equivalent to restricting $\alpha_e$ to the interval $[-1, 0]$ in $\mathcal{U}_\Gamma^I$, which is equivalent to $\mathcal{U}_\Gamma^{I1}$. □

Thus we must solve Problem A.3, with uncertainty set $\mathcal{U}_\Gamma^{I1}$.

$$\max \min_{\mathbf{w} \in \mathcal{U}_\Gamma^{I1}} \quad \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e y_{ek} + \sum_{c \in C} w_c z_c \tag{A.3a}$$

$$\text{s.t.} \quad (\mathbf{z}, \mathbf{y}) \in \mathcal{M}^P \tag{A.3b}$$

Next we develop a MILP formulation for Problem A.3 by directly minimizing its Objective (A.3a). This minimum occurs when $\lfloor \Gamma \rfloor$ edge weights are reduced by $d_e$, and one edge weight is reduced by $(\Gamma - \lfloor \Gamma \rfloor)d_e$. For this reason we refer to $d_e$ as the *discount value* of edge $e$, and all edges that receive reduced weight in the robust matching are *discounted*.

For simplicity, we define a variable $\hat{y}_e$ for each edge $e \in E$ such that $\hat{y}_e$ is 1 if edge $e$ is used in the matching, and 0 otherwise. Note that edge $e$ is used in the matching if it is used in a chain (i.e., any $y_{ek} = 1$), or if it is used in a cycle (i.e., $z_c = 1$ for any cycle $c$ containing $e$). Thus we define variables $\hat{y}_e$ using the following constraint.

$$\sum_{k \in \mathcal{K}(e)} y_{ek} + \sum_{c \in C : e \in c} z_c = \hat{y}_e \quad , e \in E$$

$$\hat{y}_e \in \{0, 1\}, e \in E$$

Next we minimize the Objective (A.3a)w.r.t. $\mathcal{U}_\Gamma^{I1}$, by discounting up to $\Gamma$ edges. Note that if only $G < \Gamma$ edges are used in a matching, only $G$ edge weights may be discounted. Thus let $\Gamma' \equiv \min\{G, \Gamma\}$ be the number of discounted edges, with

$$G = \sum_{e \in E} \hat{y}_e,$$

the total number of edges used in the matching. To linearize the definition of $\Gamma'$ we introduce variable $h$, which is 1 if $G < \Gamma$ and 0 otherwise. The statement $\Gamma' = \min(G, \Gamma)$ can be linearized using the following constraints,

$$\Gamma - G \leq Wh$$

$$G - \Gamma \leq W(1 - h)$$

$$G - Wh \leq \Gamma'$$

$$\Gamma - W(1-h) \leq \Gamma'$$

$$h \in \{0,1\}$$

where $W$ is a large constant.

The objective of Problem A.3 is minimized the the $\Gamma'$ discounted edges are those with the *largest* discount value $d_e$. To select these edges we introduce variables $g_e \in \{0,1\}$ for each edge $e \in E$. Let $m$ be the smallest $d_e$ of any discounted edge—that is, $m$ is the $\lceil \Gamma' \rceil^{th}$ highest $d_e$ of any edge used in the matching. We define $g_e$ as follows

$$g_e = \begin{cases} 1 & \text{if } d_e \geq m \\ 0 & \text{otherwise} \end{cases}$$

That is, $g_e$ is 0 if $d_e$ is smaller than the $\lceil \Gamma' \rceil^{th}$ highest discount value of edges used in the matching, and 1 otherwise. We can define these variables using linear constraints in two steps. First, note that variables $g_e$ and $d_e$ must obey the same ordering relation. That is, $g_i \geq g_j \Leftrightarrow d_i \geq d_j$ must hold for all $i, j \in E, i \neq j$. Note that variables $d_e$ are constant, and can be sorted during preprocessing. Let $\geq_d$ indicate this ordering relation.

Next we ensure that only $\Gamma'$ edges are discounted. Note that $\hat{y}_e = 1$ implies that edge $e$ is used in the matching. Edge $e$ should be discounted if it is used in the matching, and if $d_e$ is above the minimum discount value (that is, $g_e = 1$). Thus, edge $e$ should be discounted if the following identity holds

$$g_e \hat{y}_e = 1$$

Using this observation, we can ensure that exactly $\Gamma'$ edges are discounted with the following constraint,

$$\sum_{e \in E} g_e \hat{y}_e = \Gamma'.$$

For any feasible matching $M = (\mathbf{y}, \mathbf{z})$, we can directly solve the minimization in Problem A.3 by discounting the $\Gamma'$ edges used in $M$ with the largest discount values. This is accomplished using variables $g_e$; Equation A.4 gives the solution of this minimization when $\Gamma$ is integer, which is expressed as a function $Z(\mathbf{y}, \mathbf{z})$; the next section extends this formulation to accommodate non-integer $\Gamma$.

$$Z(\mathbf{y}, \mathbf{z}) = \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e y_{ek} + \sum_{c \in C} w_c z_c - \sum_{e \in E} g_e d_e \hat{y}_e \tag{A.4a}$$

$$\text{s.t.} \quad \sum_{k \in \mathcal{K}(e)} y_{ek} + \sum_{c \in C : e \in c} z_c = \hat{y}_e, e \in E \tag{A.4b}$$

$$\sum_{e \in E} \hat{y}_e = G \tag{A.4c}$$

$$\Gamma - G \leq Wh \tag{A.4d}$$

$$G - \Gamma \leq W(1 - h) \tag{A.4e}$$

$$G - Wh \leq \Gamma' \tag{A.4f}$$

$$\Gamma - W(1 - h) \leq \Gamma' \tag{A.4g}$$

$$\sum_{e \in E} g_e \hat{y}_e = \Gamma' \tag{A.4h}$$

$$g_e, \hat{y}_e \in \{0, 1\}, e \in E \tag{A.4i}$$

$$g_a \geq_d g_b, a, b \in E, a \neq b \tag{A.4j}$$

$$h \in \{0, 1\} \tag{A.4k}$$

Note that this formulation contains two sets of quadratic terms: $g_e y_{ek}$ for $k \in$

$\mathcal{K}(e)$ for $e \in E$, and $g_e z_c$ for $c \in C$ and $e \in E$. We linearize these terms in the following section, after considering non-integer $\Gamma$.

**Non-Integer $\Gamma$** The number of discounted edges $\Gamma'$ may be integer or non-integer valued. When $\Gamma'$ is not integer valued, up to $\lfloor \Gamma' \rfloor$ edges are fully discounted by value $d_e$, and the edge with the smallest discount value is discounted by $(\Gamma - \lfloor \Gamma \rfloor) d_e$. We include this fractional discount by using two sets of indicator variables $g_e^f$ and $g_e^p$ for all $e \in E$, and then discount each edge $e$ as follows:

- $e$ is fully discounted if $g_e^p = g_e^f = 1$.
- $e$ is discounted by fractional amount $(\Gamma - \lfloor \Gamma \rfloor)$ if $g_e^f = 0$ and $g_e^p = 1$
- $e$ is not discounted if $g_e^f = g_e^p = 0$.

Thus if $\Gamma'$ is integer, $g_e^f = g_e^p$ for all $e \in E$; if $\Gamma'$ is not integer, then $\lceil \Gamma' \rceil$ matching edges should be at least partially discounted ($g_e^p = 1$), and $\lfloor \Gamma' \rfloor$ matching edges should be fully discounted ($g_e^p = g_e^f = 1$). These indicator variables are defined in the same way as $g_e$ in Equation A.4: $g_e^f, g_e^p \in \{0, 1\}$, and they obey the same ordering relation as $d_e$. However, the number of matching edges with $g_e^f = 1$ can be different than the number of edges with $g_e^p = 1$.

First note that $\lceil \Gamma' \rceil$ matching edges must have $g_e^p = 1$. Recall that $G$ is the number of matching edges, and $\Gamma' = \min(\Gamma, G)$; if $\Gamma < G$, then $\lceil \Gamma' \rceil = \lceil \Gamma \rceil$, and otherwise $\lceil \Gamma' \rceil = G$. The variable $h$ is defined to be 1 if $G < \Gamma$ and 0 otherwise. Thus, we use the following constraint to require that $\lceil \Gamma' \rceil$ matching edges have $g_e^p = 1$:

$$\sum_{e \in E} g_e^p \hat{y}_e = hG + (1 - h) \lceil \Gamma \rceil.$$

Similarly, we can require that $\lfloor \Gamma' \rfloor$ edges have $g_e^f = 1$ with the following constraint

$$\sum_{e \in E} g_e^f \hat{y}_e = hG + (1-h)\lfloor \Gamma \rfloor.$$

Thus if $G < \Gamma$, then all $G$ matching edges have $g_e^f = g_e^p = 1$; otherwise, there are $\lceil \Gamma \rceil$ matching edges with $g_e^p = 1$, and $\lfloor \Gamma \rfloor$ matching edges with $g_e^f = 1$, where the matching edge with the smallest discount has $g_e^f = 0$ and $g_e^p = 1$.

Using these indicator variables, the new objective of the robust formulation is

$$\max \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e y_{ek} + \sum_{c \in C} w_c z_c - (1 - \Gamma + \lfloor \Gamma \rfloor) \sum_{e \in E} g_e^f d_e \hat{y}_e \\ - (\Gamma - \lfloor \Gamma \rfloor) \sum_{e \in E} g_e^p d_e \hat{y}_e$$

which discounts an edge $e$ by weight $d_e$ if $g_e^f = g_e^p = 1$, and by weight $d_e\,(\Gamma - \lfloor \Gamma \rfloor)$ if $g_e^f = 0$ and $g_e^p = 1$. Note that there are two sets of quadratic terms in this problem: $g_e^f \hat{y}_e$ and $g_e^p \hat{y}_e$ for all $e \in E$. To linearize these terms we introduce the variables $\hat{g}_e^f \equiv g_e^f \hat{y}_e$ and $\hat{g}_e^p \equiv g_e^p \hat{y}_e$, which we define using the following constraints.

$$\begin{aligned} \hat{g}_e^f &\leq g_e^f \\ \hat{g}_e^f &\leq \hat{y}_e \qquad\qquad , e \in E \\ \hat{g}_e^f &\geq g_e^f + \hat{y}_e - 1 \\ \hat{g}_e^f &\in \{0,1\}, e \in E \end{aligned}$$

$$\begin{aligned} \hat{g}_e^p &\leq g_e^p \\ \hat{g}_e^p &\leq \hat{y}_e \qquad\qquad , e \in E \\ \hat{g}_e^p &\geq g_e^p + \hat{y}_e - 1 \\ \hat{g}_e^p &\in \{0,1\}, e \in E \end{aligned}$$

To linearize the term $hG$, we introduce variable $\hat{g} \equiv hG$, which is defined using the following constraints. As before, $W$ is a large constant.

$$\hat{h} \leq hW$$

$$\hat{h} \leq G$$

$$\hat{h} \geq G - (1-h)W$$

$$\hat{h} \geq 0$$

Finally, for any feasible matching $M = (\mathbf{y}, \mathbf{z})$, we can directly solve the minimization in problem A.3 by discounting the $\Gamma'$ edges used in $M$ with the largest discount values. This is accomplished using variables $g_e^f$ and $g_e^p$; Equation A.5 gives the solution of this minimization for general $\Gamma > 0$.

$$Z(\mathbf{y}, \mathbf{z}) = \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e y_{ek} + \sum_{c \in C} w_c z_c - (1 - \Gamma + \lfloor \Gamma \rfloor) \sum_{e \in E} \hat{g}_e^f d_e \tag{A.5a}$$

$$- (\Gamma - \lfloor \Gamma \rfloor) \sum_{e \in E} \hat{g}_e^p d_e \tag{A.5b}$$

$$\text{s.t.} \quad \sum_{k \in \mathcal{K}(e)} y_{ek} + \sum_{c \in C : e \in c} z_c = \hat{y}_e, e \in E \tag{A.5c}$$

$$\sum_{e \in E} \hat{y}_e = G \tag{A.5d}$$

$$\Gamma - G \leq Wh \tag{A.5e}$$

$$G - \Gamma \leq W(1-h) \tag{A.5f}$$

$$G - Wh \leq \Gamma' \tag{A.5g}$$

$$\Gamma - W(1-h) \leq \Gamma' \tag{A.5h}$$

$$\sum_{e \in E} \hat{g}_e^p = \hat{h} + (1-h)\lceil \Gamma \rceil \tag{A.5i}$$

$$\sum_{e \in E} \hat{g}_e^f = \hat{h} + (1-h)\lfloor \Gamma \rfloor \tag{A.5j}$$

$$\begin{aligned} \hat{g}_e^f &\leq g_e^f \\ \hat{g}_e^f &\leq \hat{y}_e \qquad\qquad , e \in E \\ \hat{g}_e^f &\geq g_e^f + \hat{y}_e - 1 \end{aligned} \tag{A.5k}$$

$$\begin{aligned} \hat{g}_e^p &\leq g_e^p \\ \hat{g}_e^p &\leq \hat{y}_e \qquad\qquad , e \in E \\ \hat{g}_e^p &\geq g_e^p + \hat{y}_e - 1 \end{aligned} \tag{A.5l}$$

$$\hat{h} \leq hW \tag{A.5m}$$

$$\hat{h} \leq G \tag{A.5n}$$

$$\hat{h} \geq G - (1-h)W \tag{A.5o}$$

$$g_e^p, g_e^f, \hat{y}_e \in \{0,1\}, e \in E \tag{A.5p}$$

$$g_a^f \geq_d g_b^f, a, b \in E, a \neq b \tag{A.5q}$$

$$g_a^p \geq_d g_b^p, a, b \in E, a \neq b \tag{A.5r}$$

$$\hat{g}_e^p, \hat{g}_e^f \in \{0,1\}, e \in E \tag{A.5s}$$

$$h \in \{0,1\} \tag{A.5t}$$

$$\hat{h} \geq 0 \tag{A.5u}$$

Equation A.5 is the direct minimization of the Objective of $KEX(\mathcal{U}_\Gamma^{I1})$ (A.3a). Thus we directly apply this minimization solution to the original Problem A.3, to obtain the final linear formulation in Equation A.6.

$$\max \sum_{e \in E} \sum_{k \in \mathcal{K}(e)} w_e y_{ek} + \sum_{c \in C} w_c z_c - (1 - \Gamma + \lfloor \Gamma \rfloor) \sum_{e \in E} \hat{g}_e^f d_e \tag{A.6a}$$

$$- (\Gamma - \lfloor \Gamma \rfloor) \sum_{e \in E} \hat{g}_e^p d_e \tag{A.6b}$$

$$\text{s.t.} \quad \sum_{e \in \delta^-(i)} \sum_{k \in \mathcal{K}(e)} y_{ek} + \sum_{c \in C : i \in c} z_c \leq 1 \qquad i \in P \tag{A.6c}$$

$$\sum_{e \in \delta^+(i)} y_{e1} \leq 1 \qquad i \in N \tag{A.6d}$$

$$\sum_{\substack{e \in \delta^-(i) \wedge \\ k \in \mathcal{K}(e)}} y_{ek} \geq \sum_{e \in \delta^+(i)} y_{e,k+1} \qquad \begin{array}{l} i \in P \\ k \in \{1, \ldots, L-1\} \end{array} \tag{A.6e}$$

$$\sum_{k \in \mathcal{K}(e)} y_{ek} + \sum_{c \in C : e \in c} z_c = \hat{y}_e \qquad e \in E \tag{A.6f}$$

$$\sum_{e \in E} \hat{y}_e = G \tag{A.6g}$$

$$\Gamma - G \leq Wh \tag{A.6h}$$

$$G - \Gamma \leq W(1-h) \tag{A.6i}$$

$$G - Wh \leq \Gamma' \tag{A.6j}$$

$$\Gamma - W(1-h) \leq \Gamma' \tag{A.6k}$$

$$\sum_{e \in E} \hat{g}_e^p = \hat{h} + (1-h)\lceil \Gamma \rceil \tag{A.6l}$$

$$\sum_{e \in E} \hat{g}_e^f = \hat{h} + (1-h)\lfloor \Gamma \rfloor \tag{A.6m}$$

$$\begin{array}{rll} \hat{g}_e^f & \leq g_e^f & \\ \hat{g}_e^f & \leq \hat{y}_e & e \in E \\ \hat{g}_e^f & \geq g_e^f + \hat{y}_e - 1 & \end{array} \tag{A.6n}$$

$$\begin{array}{rll} \hat{g}_e^p & \leq g_e^p & \\ \hat{g}_e^p & \leq \hat{y}_e & e \in E \\ \hat{g}_e^p & \geq g_e^p + \hat{y}_e - 1 & \end{array} \tag{A.6o}$$

$$\hat{h} \leq hW \tag{A.6p}$$

$$\hat{h} \leq G \tag{A.6q}$$

$$\hat{h} \geq G - (1 - h)W \tag{A.6r}$$

$$g_a^f \geq_d g_b^f \qquad a, b \in E, a \neq b \tag{A.6s}$$

$$g_a^p \geq_d g_b^p \qquad a, b \in E, a \neq b \tag{A.6t}$$

$$y_{ek} \in \{0, 1\} \qquad e \in E, k \in \mathcal{K}(e) \tag{A.6u}$$

$$z_c \in \{0, 1\} \qquad c \in C \tag{A.6v}$$

$$g_e^p, g_e^f, \hat{y}_e \in \{0, 1\} \qquad e \in E \tag{A.6w}$$

$$\hat{g}_e^p, \hat{g}_e^f \in \{0, 1\} \qquad e \in E \tag{A.6x}$$

$$h \in \{0, 1\} \tag{A.6y}$$

$$\hat{h} \geq 0 \tag{A.6z}$$

### A.1.3 Solution Method for Constant Uncertainty Budget

This section describes the algorithm for solving the edge-weight robust formulation in Section A.1.2, when it is unreasonable to find all cycles in the exchange graph during preprocessing. We build on the cycle pricing method in Dickerson et al. [109], which in turn built on corrected versions of methods presented by Glorie et al. [148] and Plaut et al. [247].

This method begins by solving the LP relaxation of Problem A.6 on a *reduced model* (using a small number of cycles), and then identifying *positive-price cycles*—which may improve the solution—and adding these to the model. If no positive-price cycles exist, then the solution is optimal on the reduced LP relaxation. This process is known as the *pricing problem*.

After optimizing the reduced LP relaxation, we proceed in one of two ways

1. If the solution is fractional, then we fix one of the fractional variables and branch, as in a standard branch-and-bound tree,

2. If the solution is integral, then it is the optimal solution to Problem A.6.

This combination of cycle pricing and branch-and-bound is known as *branch-and-price*.

Algorithm 6 is the branch-and-price method for solving Problem A.6. There are only two inputs to this algorithm: the kidney exchange graph $G$, and the set of fixed decision variables $\mathbf{X}_F$. At each branch in the search tree, a new decision variable is fixed to either 0 or 1 and added to $\mathbf{X}_F$. When both 1) no positive price cycles exist for reduced model $\mathbf{M}$ and solution $\mathbf{X}$, and 2) the solution $\mathbf{X}$ is integral, then $\mathbf{X}$ is returned.

**Algorithm 6** BranchAndPrice

**Require:** $G, \mathbf{X}_F$
Generate subset of cycles $C'$, in $G$
Create reduced model $\mathbf{M}$, with cycles $C'$
$\mathbf{X} \leftarrow$ Solve LP relaxation of $\mathbf{M}$
$C^+ \leftarrow \text{CyclePrice}(G, \mathbf{X})$ ▷ Find positive-price cycles
**while** $C^+ \neq \varnothing$ **do**
    Add cycles $C^+$ to $\mathbf{M}$
    $\mathbf{X} \leftarrow$ solve LP relaxation of $\mathbf{M}$
    $C^+ \leftarrow \text{CyclePrice}(G, \mathbf{X})$
**if** $\mathbf{X}$ is fractional **then**
    Find fractional binary variable $X_i \in \mathbf{X}$ closest to 0.5
    $\text{BranchAndPrice}(G, \mathbf{X}_F \cup (X_i = 0))$
    $\text{BranchAndPrice}(G, \mathbf{X}_F \cup (X_i = 1))$
**elsereturn** $\mathbf{X}$

The branch-and-price method in Algorithm 6 requires a cycle-pricing algorithm GetCycles. This algorithm either returns positive-price cycles—using the reduced model $\mathbf{M}$ and the current solution to the LP relaxation, $\mathbf{X}$—or determines that none exist. We adapt the cycle-pricing algorithm used by Dickerson et al. [109] to solve the PICEF formulation, which is based on [148] and [247]. These algorithms calculate the price $p_c$ of cycle $c$ as

$$p_c = \sum_{e \in c} (w_e - \delta_v)$$

where $w_e$ is the weight of edge $e$ in cycle $c$, and $\delta_e$ is the dual value of the vertex where $e$ ends. In the edge-weight robust problem, each edge $e$ may receive its nominal weight $w_e$ or its discounted weight $(w_e - d_e)$. It is not obvious whether the nominal or discounted weights should be used during cycle pricing.

To illustrate this problem, assume we know the optimal solution $\mathbf{X}$ to Problem A.6, and the set of cycles $C$ used in $\mathbf{X}$. We consider two methods for cycle pricing.

1. Calculate cycle prices using discounted edge weights $(w_e - d_e)$.

   Assume that, for some cycle $c \in C$, none of the edges in $c$ are discounted in $\mathbf{X}$. During branch-and-price, it may occur that—before adding $c$ to the reduced model—the following inequalities hold

   $$\sum_{e \in c} (w_e - d_e - \delta_v) \leq 0$$

   $$\sum_{e \in c} (w_e - \delta_v) > 0$$

   If discounted edge weights are used during pricing, $c$ appears to have negative price—and will not be added to the reduced model. In this case, the calculated price is incorrectly negative, branch-and-price may return a sub-optimal solution.

2. Calculate cycle prices using nominal edge weights $w_e$.

Assume that, for some cycle $c' \notin C$, *all* of the edges in $c'$ are discounted when it is added to the reduced model. It may occur that the following inequalities hold:

$$\sum_{e \in c'} (w_e - d_e - \delta_v) \leq 0$$

$$\sum_{e \in c'} (w_e - \delta_v) > 0$$

In this case, using nominal edge weights for cycle pricing will incorrectly determine that $c'$ has a positive price, and will add $c'$ to the reduced model.

Neither of these methods is ideal—using discounted weights can result in a sub-optimal solution, while using nominal weights adds cycles to the reduced model. Instead, we calculate cycle prices using discounted edge weights *only* for edges that will be discounted in *any* matching, and nominal edge weights for all other edges. As discussed in Section A.1.2, up to $\Gamma$ edges are discounted in every solution to Problem A.6; these are the edges with the largest discount values $d_e$. For any exchange graph with $|E|$ edges, the $\min(\Gamma, |M|)$ edges with the largest discount values are *always* discounted if they are used in a solution to Problem A.6. Algorithm 7 describes this method, which uses the cycle pricing method of [148] as a subroutine. Proposition A.2 states that this method never incorrectly determines that a cycle has negative price—and therefore never results in a sub-optimal solution.

**Proposition A.2.** *Algorithm 7 never determines that a positive-price cycle has a negative price.*

**Algorithm 7** CyclePrice

**Require:** $G = (V, E), \mathbf{X}$

1: $d^* \leftarrow \Gamma^{th}$ highest discount value $d_e$ in $E$

2: $w_e^* \leftarrow \begin{cases} w_e - d_e & \text{if } d_e \geq d^* \\ w_e & \text{otherwise} \end{cases}$ **return** PositivePriceCycles$(G, L, \mathbf{X}, w_e^*)$, the cycle pricer from [148]

### A.1.4 Solution Method for Variable Uncertainty Budget

In this section we describe a method for solving the edge-weight robust kidney exchange problem with variable budget, $KEX(\mathcal{U}_\gamma^{I1})$. Theorem A.1 is a direct adaptation of Theorem 4 of [249] to the edge-weight uncertain kidney exchange problem, which states that the solution of $KEX(\mathcal{U}_\gamma^{I1})$ can be found by solving several cardinality-restricted instances of $KEX(\mathcal{U}_\Gamma^{I1})$.

**Theorem A.1.** *Let $\mathcal{M}$ be the set of feasible matchings, with edge decision variables $\mathbf{x} \in \mathcal{M} \subset \{0,1\}^{|E|}$. The solution to $KEX(\mathcal{U}_\gamma^{I1})$ can be found by solving $|E|$ cardinality-restricted instances of $KEX(\mathcal{U}_\Gamma^{I1})$,*

$$\begin{aligned} \max \min_{\hat{\mathbf{w}} \in \mathcal{U}_\Gamma^I} \quad & \mathbf{x} \cdot \hat{\mathbf{w}} \\ s.t. \quad & \mathbf{x} \in \mathcal{M} \\ & \|\mathbf{x}\| \leq k \\ & \Gamma = \gamma(k) \end{aligned}$$

*with $k = 1, \ldots, |E|$, and taking the maximum-weight solution.*

The proof of this theorem is identical to the proof of Theorem 4 in Poss [249], and is omitted here. In practice, feasible matchings use far fewer than $|E|$ edges, and thus many fewer than $|E|$ instances of $KEX(\mathcal{U}_\Gamma^{I1})$ must be solved. Algorithm 8 describes our method for solving $KEX(\mathcal{U}_\gamma^{I1})$, which first finds the maximum cardinality

matching, and then solves each cardinality-restricted problem $KEX(\mathcal{U}_\Gamma^{I1})$.

**Algorithm 8** EdgeWeightRobust-$\gamma$

**Require:** Function $\gamma$, exchange graph $G$
1: Find the maximum cardinality matching $\mathbf{x}_C$
2: **for** $k \leftarrow 1$ to $\|\mathbf{x}_C\|$ **do**
3: $\quad \Gamma \leftarrow \gamma(k)$
4: $\quad \mathbf{x}_k^* \leftarrow$ solution to $KEX(\mathcal{U}_\Gamma^{I1})$, restricting cardinality to $k$
**return** The maximum-weight matching in $\{\mathbf{x}_k^*\}$

## A.2 Edge Existence Robust Formulation

In this section we develop an edge existence robust formulation for kidney exchange, using uncertainty set $\mathcal{U}_\Gamma^w$. Our approach is based on a formulation introduced by Anderson et al. [16], which adapts a formulation of the prize-collecting traveling salesman problem (PC-TSP). For simplicity, we use the abbreviation $KEX(\mathcal{U})$ to refer to the robust kidney exchange problem, with uncertainty set $\mathcal{U}$.

### A.2.1 PC-TSP Formulation

We begin with an overview of the PC-TSP method proposed by Anderson et al. [16]; it is based on a method for solving the prize-collecting traveling salesman problem (PC-TSP) introduced by Balas [34]. We use a version of the PC-TSP formulation with a finite chain cap; the uncapped formulation is much more compact. (Due to high failure rates, most fielded exchanges incorporate a finite maximum length of chains. That cap can be quite high, e.g., 20 or more, but is typically not allowed to float freely with parts of the input size, e.g., $|V|$.) This formulation is especially useful because it allows us to define decision variables equal to each chain weight used in the matching, without explicitly enumerating all possible chains.

This formulation uses all of the same parameters as PICEF:

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

PC-TSP uses one decision variable for each cycle ($z_c$) and each edge ($y_e$), and several auxiliary decision variables that help define the constraints:

- $z_c \in \{0, 1\}$: 1 if cycle $c$ is used in the matching, and 0 otherwise.
- $y_e \in \{0, 1\}$: 1 if edge $e$ is used in a chain, and 0 otherwise.
- $y_e^n \in \{0, 1\}$: 1 if edge $e$ is used in a chain starting with NDD $n$, and 0 otherwise.
- $w_n^N$ (auxiliary): total weight of the chain starting with NDD $n$.
- $f_v^i$ and $f_v^o$ (auxiliary): chain flow into and out of vertex $v \in P$, respectively.
- $f_v^{i,n}$ and $f_v^{i,n}$ (auxiliary): chain flow into and out of vertex $v \in P$, respectively, from a chain beginning with NDD $n \in N$.

The PC-TSP formulation with chain cap $L$ is given in Problem A.8. As before, we use the notation $\delta^-(v)$ for the set of edges into vertex $v$ and $\delta^+(v)$ for the set of edges out of $v$.

$$\max \quad \sum_{n \in N} w_n^N + \sum_{c \in C} w_c^C z_c \tag{A.8a}$$

$$\text{s.t.} \quad \sum_{e \in E} w_e y_e^n = w_n^N \qquad n \in N \tag{A.8b}$$

$$\sum_{n \in N} y_e^n = y_e \qquad e \in E \tag{A.8c}$$

$$\sum_{e \in \delta^-(v)} y_e = f_v^i \qquad v \in V \tag{A.8d}$$

$$\sum_{e \in \delta^+(v)} y_e = f_v^o \qquad v \in V \tag{A.8e}$$

$$\sum_{e \in \delta^-(v)} y_e^n = f_v^{i,n} \qquad v \in V, n \in N \tag{A.8f}$$

$$\sum_{e \in \delta^+(v)} y_e^n = f_v^{o,n} \qquad v \in V, n \in N \tag{A.8g}$$

$$f_v^o + \sum_{c \in C : v \in c} z_c \leq \quad f_v^i + \sum_{c \in C : v \in c} z_c \leq 1 \qquad v \in P \tag{A.8h}$$

$$f_v^o \leq 1 \qquad v \in N \tag{A.8i}$$

$$\sum_{e \in \delta^-(S)} y_e \geq f_v^i \qquad S \subseteq P, v \in S \tag{A.8j}$$

$$\sum_{e \in E} y_e^n \leq L \qquad n \in N \tag{A.8k}$$

$$f_v^{o,n} \leq f^{i,v} \leq 1 \qquad v \in V, n \in N \tag{A.8l}$$

$$y_e \in \{0,1\} \qquad e \in E \tag{A.8m}$$

$$z_c \in \{0,1\} \qquad c \in C \tag{A.8n}$$

$$y_e^n \in \{0,1\} \qquad e \in E, n \in N \tag{A.8o}$$

The objective A.8a maximizes the total weight of a matching, defined by the cycle decision variables $z_c$ and edge decision variables $y^e$. The auxiliary variables are defined using the following constraints:

- Constraint A.8b: defines $w_n^N$.
- Constraint A.8c: defines $y_e$, using $y_e^n$.
- Constraints A.8d and A.8e: define auxiliary variables $f_v^i$ and $f_v^o$.
- Constraints A.8f and A.8g: define auxiliary variables $f_v^{i,n}$ and $f_v^{o,n}$.

There is only one capacity constraint for each patient-donor vertex and each NDD:

- Constraint A.8h: each patient-donor vertex $v$ may only be used in one cycle $c$; or, if $v$ is used in a chain, chain flow out of $v$ can only be nonzero if there is chain flow out of $v$.
- Constraint A.8i: each NDD $n$ may only start one chain.

The follow constraints ensure that chain flow is conserved, and enforce the chain cap $L$:

- Constraint A.8k: chains can use no more than $L$ edges.
- Constraint A.8l: chain flow out of $v$ can only be nonzero if there is chain flow out of $v$. This constraint is equivalent to A.8h, but for variables $f_v^{o,n}$.

The final constraints ensure that each chain includes an NDD. These are very similar to the generalized subtour elimination constraints in the TSP literature.

- Constraint A.8j: for every subset $S$ of the donor-patient vertices, each vertex in $S$ can only participate in a chain if there is chain flow into $S$.

The number of constraints in A.8j grows exponentially with the number of patient-donor vertices, so it is necessary to use constraint generation with the PC-TSP formulation. We avoid constraint generation by developing a new formulation, which draws on concepts of both PC-TSP and PICEF; this formulation is introduced in the following section.

### A.2.2 Our PI-TSP Formulation

In this section we present the new position-indexed PC-TSP formulation (PI-TSP), which combines concepts from both the PC-TSP formulation and the PICEF formulation. The main advantage of our approach is in the formulation of chains. PC-TSP uses a fixed number of decision variables to allow long (or uncapped) chains, but requires constraint generation. PICEF does not require constraint generation, but the number of decision variables grows polynomially with the chain cap.

Our approach achieves the best of both worlds: PI-TSP uses a *fixed* number of decision variables for any chain cap, and does not require constraint generation. To our knowledge, ours is the first formulation to exhibit this behavior.

PI-TSP uses the same parameters as PICEF and PC-TSP:

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

PI-TSP also uses the same decision variables (and auxiliary variables) as PC-TSP. Two additional variables are added to the formulation: $p_e, p_v \geq 1$ for each edge $e \in E$ and patient-donor vertex $v \in P$, to represent $e$ and $v$'s position in a chain.

- $p_e \geq 1$: the position of edge $e$ in any chain.
- $p_v \geq 1$: the position of patient-donor vertex $v$ in any chain (equal to the position of any incoming edge).

- $\hat{p}_e \geq 0$: equal to $p_e$ if $e$ is used in a chain, and 0 otherwise. (i.e., $\hat{p}_e = p_e \cdot y_e$)
- $z_c \in \{0,1\}$: 1 if cycle $c$ is used in the matching, and 0 otherwise.
- $y_e \in \{0,1\}$: 1 if edge $e$ is used in a chain, and 0 otherwise.
- $y_e^n \in \{0,1\}$: 1 if edge $e$ is used in a chain starting with NDD $n$, and 0 otherwise.
- $w_n^N$ (auxiliary): total weight of the chain starting with NDD $n$.
- $f_v^i$ and $f_v^o$ (auxiliary): chain flow into and out of vertex $v \in P$, respectively.
- $f_v^{i,n}$ and $f_v^{i,n}$ (auxiliary): chain flow into and out of vertex $v \in P$, respectively, from a chain beginning with NDD $n \in N$.

The PI-TSP formulation with chain cap $L$ is given in Problem A.9. As before, we use the notation $\delta^-(v)$ for the set of edges into vertex $v$ and $\delta^+(v)$ for the set of edges out of $v$.

$$\max \quad \sum_{n \in N} w_n^N + \sum_{c \in C} w_c^C z_c \tag{A.9a}$$

$$\text{s.t.} \quad \sum_{e \in E} w_e y_e^n = w_n^N \qquad n \in N \tag{A.9b}$$

$$\sum_{n \in N} y_e^n = y_e \qquad e \in E \tag{A.9c}$$

$$\sum_{e \in \delta^-(v)} y_e = f_v^i \qquad v \in V \tag{A.9d}$$

$$\sum_{e \in \delta^+(v)} y_e = f_v^o \qquad v \in V \tag{A.9e}$$

$$\sum_{e \in \delta^-(v)} y_e^n = f_v^{i,n} \qquad v \in V, n \in N \tag{A.9f}$$

$$\sum_{e \in \delta^+(v)} y_e^n = f_v^{o,n} \qquad v \in V, n \in N \tag{A.9g}$$

$$f_v^o + \sum_{c \in C : v \in c} z_c \leq \quad f_v^i + \sum_{c \in C : v \in c} z_c \leq 1 \qquad v \in P \tag{A.9h}$$

$$f_v^o \leq 1 \qquad v \in N \tag{A.9i}$$

$$p_e = 1 \qquad e \in \delta^+(N) \tag{A.9j}$$

$$\hat{p}_e = p_e y_e \qquad e \in E \tag{A.9k}$$

$$p_v = \sum_{e \in \delta^-(v)} \hat{p}_e \qquad v \in P \tag{A.9l}$$

$$p_e = p_v + 1 \qquad v \in P, e \in \delta^+(v) \tag{A.9m}$$

$$\sum_{e \in E} y_e^n \leq L \qquad n \in N \tag{A.9n}$$

$$f_v^{o,n} \leq f^{i,v} \leq 1 \qquad v \in V, n \in N \tag{A.9o}$$

$$y_e \in \{0,1\} \qquad e \in E \tag{A.9p}$$

$$z_c \in \{0,1\} \qquad c \in C \tag{A.9q}$$

$$y_e^n \in \{0,1\} \qquad e \in E, n \in N \tag{A.9r}$$

All constraints are identical to those of PC-TSP, but without the subtour elimination constraints A.8j, and with the addition of the following constraints:

- Constraints A.9j: sets $p_e = 1$ for all edges out of NDD vertices.
- Constraints A.9k: defines $\hat{p}_e$.
- Constraints A.9l: for all vertices $v$, sets $p_v$ equal to the variable $p_e$ of *any* incoming edge.
- Constraints A.9m: for all outgoing edges of all vertices $v$, sets $p_e = p_v + 1$.

Two adjustments may be made to this formulation: first, the variables $p_v$ are not necessary, but are useful for illustration. We can remove these variables by combining Constraints A.9l and A.9m as follows:

$$p_{\bar{e}} = 1 + \sum_{e \in \delta^-(v)} \hat{p}_e \qquad v \in P, \bar{e} \in \delta^+(v)$$

Second, Constraints A.9k are nonlinear; we linearize these by replacing A.9k with the following constraints for each $e \in E$:

$$\hat{p}_e \leq y_e M$$

$$\hat{p}_e \leq p_e$$

$$p_e - (1 - y_e)M \leq \hat{p}_e$$

#### A.2.2.1 Experiments: Minimum Chain Length

We demonstrate the utility of the PI-TSP formulation by finding optimal matchings with a *minimum* chain length ($L_{min}$). We set a *maximum* chain length of $L_{max} = 3$, and vary the $L_{min}$ from 0 to 3. For some exchange graph, let $|M_{OPT}|$ be the score of the *optimal* matching (i.e., with no minimum chain length, and maximum chain length 3); we calculate the fractional optimality gap for the matching $M_l$ (with score $|M_l|$), which has minimum chain length $L_{min} = l$. We define $\Delta OPT(M_l)$ as

$$\Delta OPT(M_l) = \frac{|M_l| - |M_{OPT}|}{|M_{OPT}|}$$

We calculate optimal matchings for $L_{min} = 0, 1, 2, 3$, for each of the UNOS exchange graphs used in Section 6.5. Only 154 of the roughly 300 UNOS graphs contain chains; the remaining graphs may have no NDDs, or the NDDs may have no feasible donors. Focusing on these 154 graphs, we calculate $\Delta OPT$ and the chain lengths of each optimal matching, for each $L_{min}$. Figure A.1 shows histograms of $\Delta OPT$ and the chain lengths for all optimal matchings, for each $L_{min} = 0, 1, 2, 3$. Note that $\Delta OPT$ is zero for $L_{min} = 0$, by definition.

For some of these exchanges, a minimum chain length of 2 or 3 was infeasible (58 for $L_{min} = 2$, and 77 for $L_{min} = 3$, out of 154 total exchanges); we do not consider

these cases.

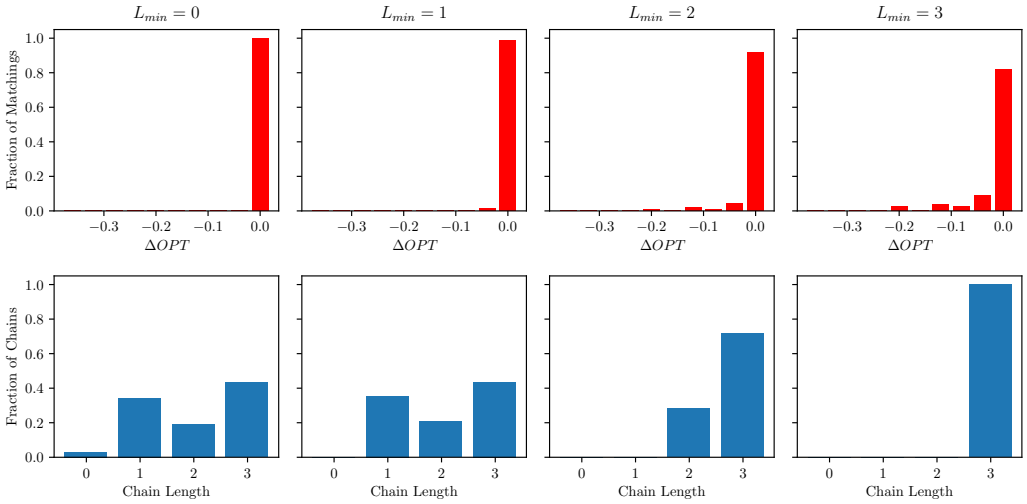



FIGURE A.1: $\Delta OPT$ (top row) and chain lengths (bottom row) for the optimal matchings with minimum chain length $L_{min}$, and maximum chain length of 3.

As expected, enforcing $L_{min} > 0$ results in longer chains – such that when $L_{min} = L_{max} = 3$, *all* chains have length 3. Surprisingly, enforcing a minimum chain length does not impact the overall matching score. Indeed, even when $L_{min} = 3$, 70% of all matchings have a *zero optimality gap*. However these experiments did not consider edge failures. As discussed in Section 3.4, edge failures impact long cycles and chains more than short cycles and chains; in practice, when edges have a nonzero failure probability, setting a high $L_{min}$ makes the matching more susceptible to failure (i.e., less robust).

### A.2.3 Edge Existence Robust Formulation

In this section we develop a mixed integer linear program formulation for the edge existence robust kidney exchange ($KEX(\mathcal{U}_\Gamma^w)$). This problem maximizes the matching score while minimizing the objective with respect to realized cycle and chain weights $\hat{\mathbf{w}}_\mathbf{c}^\mathbf{M}$ for the current matching $M$. We develop an edge-existence robust formulation by directly minimizing the PI-TSP Objective (A.9a) over all cycle and chain

weight realizations in $\mathcal{U}_\Gamma^w$. For brevity, let $\mathcal{M}^P$ be the set of all possible feasible solutions to the PI-TSP formulation; we represent these feasible solutions as $(\mathbf{y}, \mathbf{z}) \in \mathcal{M}$, where $\mathbf{y}$ are the edge decision variables for chains, and $\mathbf{z}$ are the cycle decision variables.

For any feasible solution $(\mathbf{y}, \mathbf{z}) \in \mathcal{M}^P$, we find the minimum objective value for any realized cycle and chain weights in $\mathcal{U}_\Gamma^w$ With some abuse of notation, this minimum is represented by the function $Z(\mathbf{y}, \mathbf{z})$. In this section we separate the realized weights $\hat{\mathbf{w}}_\mathbf{c}$ into the realized cycle weights $\hat{\mathbf{w}}_\mathbf{c}^\mathbf{C}$ and the realized chain weights $\hat{\mathbf{w}}_\mathbf{c}^\mathbf{N}$.

$$Z(\mathbf{y}, \mathbf{z}) = \min_{(\hat{\mathbf{w}}_\mathbf{c}^\mathbf{C}, \hat{\mathbf{w}}_\mathbf{c}^\mathbf{N}) \in \mathcal{U}_\Gamma^w} \sum_{n \in N} \hat{w}_n^N + \sum_{c \in C} \hat{w}_c^C z_c$$

Note that maximizing $Z(\mathbf{y}, \mathbf{z})$ is equivalent to solving $KEX(\mathcal{U}_\Gamma^w)$ – the robust kidney exchange problem with uncertainty set $KEX(\mathcal{U}_\Gamma^w)$. The following lemma states that this is equivalent to solving the constant-budget edge existence robust kidney exchange problem $KEX(\mathcal{U}_\Gamma^E)$.

**Lemma A.2.** *$KEX(\mathcal{U}_\Gamma^E)$ is equivalent to $KEX(\mathcal{U}_\Gamma^w)$*

*Proof.* Consider a feasible matching $M = (z_c, y_e)$. The only difference between $KEX(\mathcal{U}_\Gamma^E)$ and $KEX(\mathcal{U}_\Gamma^w)$ is the minimization of the objective over uncertainty sets $\mathcal{U}_\Gamma^E$ and $\mathcal{U}_\Gamma^w$ respectively.

Problem $KEX(\mathcal{U}_\Gamma^E)$ minimizes the matching weight over edge subsets $\hat{E} \subseteq E$, where $R = E \setminus \hat{E}$ contains up to $\Gamma$ edges:

- If $\Gamma = 1$, the largest decrease in matching weight occurs if the *highest weight* cycle or chain is discounted – that is, if $R$ contains the first edge in the highest weight chain, or any edge in the highest weight cycle.

- Similarly if $\Gamma = 2$, the largest decrease in matching weight occurs when the two highest-weight cycles and chains are discounted.

Thus, for any positive $\Gamma$ and any feasible matching $M$, the minimum objective in $KEX(\mathcal{U}_\Gamma^E)$ occurs when the $\Gamma$ highest-weight cycles and chains in $M$ are discounted.

In $KEX(\mathcal{U}_\Gamma^w)$, for any $\Gamma$ and any feasible matching $M$, the minimum occurs (trivially) when the $\Gamma$ highest-weight cycles or chains are discounted in $\mathcal{U}_\Gamma^w$.

For any matching $M$, minimizing the $KEX$ objective over $\mathcal{U}^E$ and $\mathcal{U}_\Gamma^w$ produce the same outcome – the $\Gamma$ highest-weight cycles and chains are discounted. Thus, the minimization in $KEX(\mathcal{U}^E)$ and $KEX(\mathcal{U}_\Gamma^w)$ is equivalent. □

Thus, to solve the constant-budget edge existence robust kidney exchange problem, we can solve Problem A.10 – which maximizes $Z(\mathbf{y}, \mathbf{z})$ over all feasible matchings $(\mathbf{y}, \mathbf{z}) \in \mathcal{M}^P$.

$$\max \quad Z(\mathbf{y}, \mathbf{z}) \tag{A.10a}$$

$$(\mathbf{y}, \mathbf{z}) \in \mathcal{M}^P \tag{A.10b}$$

We proceed by solving Problem A.10, which is equivalent to $KEX(\mathcal{U}_\Gamma^w)$. To solve this problem we first develop a linear formulation for $Z$ using the PC-TSP decision variables, and then we maximize this linear expression.

### A.2.4 Linear Formulation for $Z$

In this section we minimize the function $Z$ for any matching $(\mathbf{y}, \mathbf{z}) \in \mathcal{M}^P$, within uncertainty set $\mathcal{U}_\Gamma^w$. Within this uncertainty set, up to $\lfloor \Gamma \rfloor$ cycles and chains can have zero realized weight (i.e., $\hat{w}_c = 0$), and if $\Gamma$ is not integer, then one cycle or chain will have realized weight equal to the fraction $(\Gamma - \lfloor \Gamma \rfloor)$ of its total nominal weight (i.e., $\hat{w}_c = (\Gamma - \lfloor \Gamma \rfloor) w_c$. We say that any cycle or chain $c$ with $\hat{w}_c < w_c$ is *discounted*.

First note that if a matching uses $G$ cycles and chains, and $G < \Gamma$, only $G$ objects are discounted. Thus let $\Gamma' = \min\{G, \Gamma\}$ be the number of discounted cycles and chains, i.e.,

$$G = \sum_{c \in C} z_c + \sum_{n \in N} \sum_{e \in \delta^+(n)} y_e.$$

To linearize the definition of $\Gamma'$, we introduce variable $h$, which is 1 if $G < \Gamma$ and 0 otherwise. The statement $\Gamma' = \min\{G, \Gamma\}$ is linearized using the following constraints:

$$\Gamma - G \leq Wh$$

$$G - \Gamma \leq W(1-h)$$

$$G - Wh \leq \Gamma'$$

$$\Gamma - W(1-h) \leq \Gamma'$$

$$h \in \{0, 1\}$$

where $W$ is a large constant.

The function $Z$ is minimized w.r.t. the realized weights, when the $\Gamma'$ discounted cycles and chains are those with the *largest* weight. To select these objects we introduce variables $g_c^C, g_n^N \in \{0, 1\}$ for each cycle $c \in C$ and each chain's NDD $n \in N$. For any matching, let $m$ be the smallest weight of any discounted cycle or chain – that is, $m$ is the $\lceil \Gamma' \rceil^{th}$ highest weight of any cycle or chain used in the matching. We define $g_c^C$ and $g_n^N$ as follows

$$g_c^C = \begin{cases} 1 & \text{if } w_c^C \geq m \\ 0 & \text{otherwise} \end{cases} \qquad g_n^N = \begin{cases} 1 & \text{if } w_n^N \geq m \\ 0 & \text{otherwise} \end{cases}$$

Thus $g_c^C = 1$ or $g_n^N = 1$ implies that cycle $c$ or chain $n$ should be discounted if used in the matching. We define these variables using linear constraints, in two steps. First, we require that $g_j^{\{C,N\}} = 1$ only if $g_k^{\{C,N\}} = 1$ for all cycles and chains $k$ with weight larger than $w_j^{\{C,N\}}$. That is, we require that variables $g^{\{C,N\}}$ obey the same ordering as $w^{\{C,N\}}$. This ordering requirement can be defined using the following correspondences

$$g_i^C \geq g_j^C \Leftrightarrow w_i^C \geq w_j^C, \quad i,j \in C \tag{A.11}$$

$$g_c^C > g_n^N \Leftrightarrow w_c^C > w_n^N, \quad c \in C, n \in N \tag{A.12}$$

$$g_c^C \leq g_n^N \Leftrightarrow w_c^C \leq w_n^N, \quad c \in C, n \in N \tag{A.13}$$

$$g_i^N \geq g_j^N \Leftrightarrow w_i^N \geq w_j^N, \quad i,j \in N \tag{A.14}$$

Note that cycle weights are fixed but chain weights depend on the decision variables. Thus we determine ordering relation A.11 by sorting all cycle weights during preprocessing, and enforcing this ordering over $g_i^C$ using the relation $\geq_C$, defined as

$$\geq_C = \left\{ (g_a^C, g_b^C) \in \mathbf{g}^C \times \mathbf{g}^C \mid w_a^C \geq w_b^C \right\}.$$

Using this notation, the ordering relation $\geq_C$ contains all pairs of cycles $(a,b)$ such that $w_a^C \geq w_b^C$. For simplicity, I will denote this ordering relation as

$$a \geq_C b.$$

This ordering relation is enforced on variables $g_i^C$ using $(|C| - 1)$ constraints. The ordering required by correspondence A.12, A.13, and A.14 depend on the chain

weights, which in turn depend on decision variables. We can linearize these correspondences using the following inequalities

$$g_c^C + q_{cn} \geq g_n^N$$
$$W(1 - q_{cn}) \geq w_c^C - w_n^N$$

$$g_n^N + (1 - q_{cn}) \geq g_c^C$$
$$W q_{cn} \geq w_n^N - w_c^C$$

$$q_{cn} \in \{0,1\}, c \in C, n \in N,$$

Where $W$ is a large constant. When $w_c^C > w_n^N$, this forces $q_{cn}$ to be 0; as a result, the inequality $g_c^C \geq g_n^N$ must hold. Otherwise, if $w_c^C < w_n^N$, this forces $q_{cn}$ to be 1, which forces the inequality $g_n^N \geq g_c^C$ to hold.

Similarly, the following constraints enforce the ordering in correspondence A.14 over variables $g_n^N$

$$g_i^N + q_{ij}^N \geq g_j^N$$
$$W(1 - q_{ij}^N) \geq w_i^N - w_j^N$$

$$g_j^N + (1 - q_{ij}^N) \geq g_i^N$$
$$W q_{ij}^N \geq w_j^N - w_i^N$$

$$q_{ij}^N \in \{0,1\}, i, j \in N, i \neq j$$

Next we require that only $\Gamma'$ objects are discounted. Note that if cycle $c$ is discounted if $g_c^C w_c^C = 1$, and chain $n$ is discounted if $g_n^N \sum_{e \in E} y_e^n = 1$. Thus, the following identity requires that exactly $\Gamma$ objects are discounted:

$$\sum_{c \in C} z_c g_c^C + \sum_{n \in N} g_n^N \sum_{e \in \delta^+(n)} y_e = \Gamma'$$

We use these variables to directly minimize $Z(\mathbf{y}, \mathbf{z})$ w.r.t. $\mathcal{U}_\Gamma^w$, and the result is given in Equation A.15.

$$Z(\mathbf{y}, \mathbf{z}) = \sum_{n \in N} w_n^N + \sum_{c \in C} w_c z_c - \sum_{n \in N} g_n^N w_n^N - \sum_{c \in C} g_c^C w_c z_c \tag{A.15a}$$

$$\text{s.t.} \quad \Gamma - G \leq Wh \tag{A.15b}$$

$$G - \Gamma \leq W(1-h) \tag{A.15c}$$

$$G - Wh \leq \Gamma' \tag{A.15d}$$

$$\Gamma - W(1-h) \leq \Gamma' \tag{A.15e}$$

$$\begin{aligned} g_c^C + q_{cn} &\geq g_n^N \\ W(1 - q_{cn}) &\geq w_c^C - w_n^N \\ g_n^N + (1 - q_{cn}) &\geq g_c^C \\ W q_{cn} &\geq w_n^N - w_c^C \end{aligned} \quad , c \in C, n \in N \tag{A.15f}$$

$$\begin{aligned} g_i^N + q_{ij}^N &\geq g_j^N \\ W(1 - q_{ij}^N) &\geq w_i^N - w_j^N \end{aligned} \quad , q_{ij}^N \in \{0,1\}, i, j \in N, i \neq j \tag{A.15g}$$

$$\sum_{c \in C} z_c g_c^C + \sum_{n \in N} g_n^N \sum_{e \in \delta^+(n)} y_e = \Gamma' \tag{A.15h}$$

$$g_i^C \geq_C g_j^C \quad , i, j \in C, i \neq j \tag{A.15i}$$

$$q_{cn} \in \{0,1\}, \qquad c \in C, n \in N, \tag{A.15j}$$

$$h \in \{0,1\} \tag{A.15k}$$

Note that there are two sets of quadratic expressions in this formulation: $g_c^C z_c$, and $w_n^N g_n^N$. These are linearized in the next section, which addresses non-integer $\Gamma$.

### A.2.5 Non-Integer Uncertainty Budget

When $\Gamma$ is not integer, the actual number of discounted cycles and chains ($\Gamma'$) may be integer or non-integer valued. When $\Gamma'$ is not integer valued, up to $\lfloor \Gamma' \rfloor$ cycles and chains are fully discounted (i.e., $\hat{w}_c = 0$), and the smallest-weight cycle or chain is discounted by fraction $(\Gamma - \lfloor \Gamma \rfloor)$. We include this fractional discount by using two sets of indicator variables $f_i^{\{C,N\}}$ and $p_i^{\{C,N\}}$ for all cycles and chains $i \in C \cup N$, and then discount each $i$ as follows:

- $i$ is fully discounted if $p_i^{\{C,N\}} = f_i^{\{C,N\}} = 1$.
- $i$ is partially discounted fraction $(\Gamma - \lfloor \Gamma \rfloor)$ if $f_i^{\{C,N\}} = 0$ and $p_i^{\{C,N\}} = 1$
- $i$ is not discounted if $f_i^{\{C,N\}} = p_i^{\{C,N\}} = 0$.

Thus if $\Gamma'$ is integer, $f_i^{\{C,N\}} = p_i^{\{C,N\}}$ for all cycles and chains $i$; if $\Gamma'$ is not integer, then $\lceil \Gamma' \rceil$ cycles and chains are least partially discounted ($p_e^{\{C,N\}} = 1$), and $\lfloor \Gamma' \rfloor$ cycles and chains are fully discounted ($p_i^{\{C,N\}} = g_i^f = 1$). These indicator variables are defined in the same way as $g_i^{\{C,N\}}$ in Equation A.15: $p_i^{\{C,N\}}, f_i^{\{C,N\}} \in \{0,1\}$, and they obey the same ordering relation as the cycle and chain weights. However, the number of cycles and chains with $f_i^{\{C,N\}} = 1$ can be different than the number of cycles and chains with with $p_i^{\{C,N\}} = 1$. Thus we add new constraints for each of these variables.

**Setting the number of discounted cycles and chains.** First we require $\lceil \Gamma' \rceil$ cycles and chains have $p_i^{\{C,N\}} = 1$. Recall that $G$ is the number of matching edges, and $\Gamma' = \min(\Gamma, G)$; if $\Gamma < G$, then $\lceil \Gamma' \rceil = \lceil \Gamma \rceil$, and $\lceil \Gamma' \rceil = G$ otherwise. The variable $h$ is defined to be 1 if $G < \Gamma$ and 0 otherwise. Thus, the following constraint requires that $\lceil \Gamma' \rceil$ cycles and chains have $p_i^{\{C,N\}} = 1$:

$$\sum_{n \in N} p_n^N \sum_{e \in \delta^+(n)} y_e + \sum_{c \in C} p_c^C z_c = hG + (1-h)\lceil \Gamma \rceil.$$

Similarly, the following constraint requires that $\lfloor \Gamma' \rfloor$ cycles and chains have $f_i^{\{C,N\}} = 1$:

$$\sum_{n \in N} f_n^N \sum_{e \in \delta^+(n)} y_e + \sum_{c \in C} f_c^C z_c = hG + (1-h)\lfloor \Gamma \rfloor.$$

Thus if $G < \Gamma$, then all $G$ cycles and chains will have $f_i^{\{C,N\}} = p_i^{\{C,N\}} = 1$; otherwise, there are $\lceil \Gamma \rceil$ cycles and chains with $p_i^{\{C,N\}} = 1$, and $\lfloor \Gamma \rfloor$ cycles and chains with $f_i^{\{C,N\}} = 1$, where the partially-discounted cycle or chain has $f_i^{\{C,N\}} = 0$ and $p_i^{\{C,N\}} = 1$.

**Ordering relation over indicator variables.** To enforce the ordering relation over indicator variables $f_n^N$, $p_n^N$, $f_c^C$, and $p_c^C$, we use constraints similar to those used in the edge weight robust formulation. The auxiliary variables $q_{cn}$ and $q_{ij}^N$ are defined the same way here: $q_{cn}$ is 0 when $w_n^C > w_n^N$ and 1 otherwise; $q_{ij}^N$ is 0 if

$$
\begin{aligned}
f_c^C + q_{cn} &\geq f_n^N \\
p_c^C + q_{cn} &\geq p_n^N \\
f_n^N + (1 - q_{cn}) &\geq f_c^C \\
p_n^N + (1 - q_{cn}) &\geq p_c^C \\
W(1 - q_{cn}) &\geq w_c^C - w_n^N \\
W q_{cn} &\geq w_n^N - w_c^C
\end{aligned}
\qquad , c \in C, n \in N,
$$

$$
q_{cn} \in \{0, 1\}, c \in C, n \in N,
$$

Where $W$ is a large constant. When $w_c^C > w_n^N$, this forces $q_{cn}$ to be 0; as a result, the inequality $f_c^C \geq f_n^N$ and $p_c^C \geq p_n^N$ must hold. Otherwise, if $w_c^C < w_n^N$, this forces $q_{cn}$ to be 1, which forces the inequality $f_n^N \geq f_c^C$ and $p_n^N \geq p_c^C$ to hold.

Similarly, the following constraints enforce the ordering in correspondence A.14 over variables $f_n^N$ and $p_n^N$

$$
\begin{aligned}
f_i^N + q_{ij}^N &\geq f_j^N \\
p_i^N + q_{ij}^N &\geq p_j^N \\
f_j^N + (1 - q_{ij}^N) &\geq f_i^N \\
p_j^N + (1 - q_{ij}^N) &\geq p_i^N \\
W q_{ij}^N &\geq w_j^N - w_i^N \\
W(1 - q_{ij}^N) &\geq w_i^N - w_j^N
\end{aligned}
\qquad , i, j \in N, i \neq j
$$

$$
q_{ij}^N \in \{0, 1\}, i, j \in N, i \neq j
$$

As before, correspondence A.11, the ordering between cycle indicator variables, is enforced using the pre-determined ordering $\geq_C$.

$$\begin{matrix} f_a^C & \geq_C & f_b^C \\ p_a^C & \geq_C & p_b^C \end{matrix}, a, b \in C, a \neq b$$

**Objective for non-integer $\Gamma$.** Using these indicator variables, the new objective of the robust formulation is

$$\max \sum_{n \in N} w_n^N + \sum_{c \in C} w_c z_c - (1 - \Gamma + \lfloor \Gamma \rfloor) \left( \sum_{n \in N} w_n^N f_n^N + \sum_{c \in C} f_c^C w_c z_c \right) - (\Gamma - \lfloor \Gamma \rfloor) \left( \sum_{n \in N} w_n^N p_n^N + \sum_{c \in C} p_c^C w_c z_c \right)$$

which discounts cycle or chain $i$ by its full weight if $f_i^{\{C,N\}} = p_i^{\{C,N\}} = 1$, and by fraction $(\Gamma - \lfloor \Gamma \rfloor)$ of its weight if $f_i^{\{C,N\}} = 0$ and $p_i^{\{C,N\}} = 1$.

**Non-linear terms.** There are now 7 types of nonlinear terms in this formulation:

- $hG$,
- $w_n^N f_n^N$,
- $w_n^N p_n^N$,
- $z_c f_c^C$,
- $z_c p_c^C$,
- $f_n^N y_e$, and
- $p_n^N y_e$.

First we linearize the chain-related quadratic terms by introducing the variables $\hat{f}_n^N \equiv w_n^N f_n^N$ and $\hat{p}_n^N \equiv w_n^N p_n^N$. The following constraints define these new variables, using a large constant $W$.

$$\begin{aligned} \hat{f}_n^N &\leq f_n^N W \\ \hat{f}_n^N &\leq w_n^N \qquad\qquad\qquad , n \in N \\ \hat{f}_n^N &\geq w_n^N - (1 - f_n^N)W \\ & \qquad\qquad\qquad \hat{f}_n^N \geq 0, n \in N \end{aligned}$$

$$\begin{aligned} \hat{p}_n^N &\leq p_n^N W \\ \hat{p}_n^N &\leq w_n^N \qquad\qquad\qquad , n \in N \\ \hat{p}_n^N &\geq w_n^N - (1 - p_n^N)W \\ & \qquad\qquad\qquad \hat{p}_n^N \geq 0, n \in N \end{aligned}$$

Next we define variables $\hat{f}_c^C \equiv z_c f_c^C$ and $\hat{p}_c^C \equiv z_c p_c^C$ using the following constraints.

$$\begin{aligned} \hat{f}_c^C &\leq f_c^C \\ \hat{f}_c^C &\leq z_c \qquad\qquad , c \in C \\ \hat{f}_c^C &\geq f_c^C + z_c - 1 \\ & \qquad\qquad \hat{f}_c^C \in \{0, 1\}, c \in C \end{aligned}$$

$$\begin{aligned}
\hat{p}_c^C &\leq p_c^C \\
\hat{p}_c^C &\leq z_c \qquad\qquad , c \in C \\
\hat{p}_c^C &\geq p_c^C + z_c - 1 \\
\hat{p}_c^C &\in \{0,1\}, c \in C
\end{aligned}$$

To linearize the term $hG$, we introduce variable $\hat{g} \equiv hG$, which is defined using the following constraints. As before, $W$ is a large constant.

$$\begin{aligned}
\hat{h} &\leq hW \\
\hat{h} &\leq G \\
\hat{h} &\geq G - (1-h)W \\
\hat{h} &\geq 0
\end{aligned}$$

Finally, we introduce the variables $F_n \equiv f_n^N \sum_{e \in \delta^+(n)} y_e$ and $P_n \equiv p_n^N \sum_{e \in \delta^+(n)} y_e$, defined with the following constraints. Note that for each NDD $n \in N$ the sum of all $y_e$ variables is either zero (if $n$ does not initiate a chain) or 1 (if $n$ initiates a chain). Thus $F_n$ and $P_n$ are products of binary variables, which we define using the following constraints.

$$\begin{aligned}
F_n &\leq f_n^N \\
F_n &\leq \sum_{e \in \delta^+(n)} y_e \qquad\qquad , c \in C \\
F_n &\geq \sum_{e \in \delta^+(n)} y_e + f_n^N - 1 \\
F_n &\in \{0,1\}, n \in N
\end{aligned}$$

$$
\begin{aligned}
P_n &\leq p_n^N \\
P_n &\leq \sum_{e \in \delta^+(n)} y_e \qquad\qquad , n \in N \\
P_n &\geq \sum_{e \in \delta^+(n)} y_e + p_n^N - 1 \\
& P_n \in \{0,1\}, n \in N
\end{aligned}
$$

**Linear formulation.** Finally, for any feasible matching we directly minimize $Z$ by discounting the $\Gamma'$ largest-weight cycles and chains. This is accomplished using the variables $\hat{f}_n^N$, $\hat{p}_n^N$, $\hat{f}_c^C$, $\hat{p}_c^C$. Equation A.16 gives the minimization of $Z$ for any matching $(\mathbf{y}, \mathbf{z})$ , using only linear constraints.

$$
\begin{aligned}
Z(\mathbf{y}, \mathbf{z}) = \quad & \sum_{n \in N} w_n^N + \sum_{c \in C} w_c z_c - (1 - \Gamma + \lfloor \Gamma \rfloor) \left( \sum_{n \in N} \hat{f}_n^N + \sum_{c \in C} \hat{f}_c^C w_c \right) \\
& - (\Gamma - \lfloor \Gamma \rfloor) \left( \sum_{n \in N} \hat{p}_n^N + \sum_{c \in C} \hat{p}_c^C w_c \right)
\end{aligned} \tag{A.16a}
$$

s.t.

$$
\begin{aligned}
\Gamma - G &\leq Wh \\
G - \Gamma &\leq W(1-h) \\
G - Wh &\leq \Gamma' \\
\Gamma - W(1-h) &\leq \Gamma'
\end{aligned} \tag{A.16b}
$$

$$
\sum_{n \in N} P_n + \sum_{c \in C} \hat{p}_c^C = \hat{h} + (1-h)\lceil \Gamma \rceil \tag{A.16c}
$$

$$
\sum_{n \in N} F_n + \sum_{c \in C} \hat{f}_c^C = \hat{h} + (1-h)\lfloor \Gamma \rfloor \tag{A.16d}
$$

$$\begin{aligned} f_c^C + q_{cn} &\geq f_n^N \\ p_c^C + q_{cn} &\geq p_n^N \\ f_n^N + (1 - q_{cn}) &\geq f_c^C \\ p_n^N + (1 - q_{cn}) &\geq p_c^C \\ W(1 - q_{cn}) &\geq w_c^C - w_n^N \\ W q_{cn} &\geq w_n^N - w_c^C \end{aligned} \quad , \quad c \in C, n \in N$$

(A.16e)

$$\begin{aligned} f_i^N + q_{ij}^N &\geq f_j^N \\ p_i^N + q_{ij}^N &\geq p_j^N \\ f_j^N + (1 - q_{ij}^N) &\geq f_i^N \\ p_j^N + (1 - q_{ij}^N) &\geq p_i^N \\ W q_{ij}^N &\geq w_j^N - w_i^N \\ W(1 - q_{ij}^N) &\geq w_i^N - w_j^N \end{aligned} \quad , \quad i, j \in N, i \neq j$$

(A.16f)

$$\begin{aligned} f_a^C &\geq_C f_b^C \\ p_a^C &\geq_C p_b^C \end{aligned} \quad , \quad a, b \in C, a \neq b$$

(A.16g)

$$\begin{aligned} \hat{f}_n^N &\leq f_n^N W \\ \hat{f}_n^N &\leq w_n^N \\ \hat{f}_n^N &\geq w_n^N - (1 - f_n^N) W \end{aligned} \quad , \quad n \in N$$

(A.16h)

$$\begin{aligned} \hat{p}_n^N &\leq p_n^N W \\ \hat{p}_n^N &\leq w_n^N \qquad , \quad n \in N \\ \hat{p}_n^N &\geq w_n^N - (1 - p_n^N) W \end{aligned} \tag{A.16i}$$

$$\begin{aligned} \hat{f}_c^C &\leq f_c^C \\ \hat{f}_c^C &\leq z_c \qquad , \quad c \in C \\ \hat{f}_c^C &\geq f_c^C + z_c - 1 \end{aligned} \tag{A.16j}$$

$$\begin{aligned} \hat{p}_c^C &\leq p_c^C \\ \hat{p}_c^C &\leq z_c \qquad , \quad c \in C \\ \hat{p}_c^C &\geq p_c^C + z_c - 1 \end{aligned} \tag{A.16k}$$

$$\begin{aligned} \hat{h} &\leq hW \\ \hat{h} &\leq G \\ \hat{h} &\geq G - (1 - h) W \\ \hat{h} &\geq 0 \end{aligned} \tag{A.16l}$$

$$\begin{aligned} F_n &\leq f_n^N \\ F_n &\leq \sum_{e \in \delta^+(n)} y_e \qquad , c \in C \\ F_n &\geq \sum_{e \in \delta^+(n)} y_e + f_n^N - 1 \end{aligned} \tag{A.16m}$$

$$\begin{aligned} P_n &\leq p_n^N \\ P_n &\leq \sum_{e \in \delta^+(n)} y_e \qquad , n \in N \\ P_n &\geq \sum_{e \in \delta^+(n)} y_e + p_n^N - 1 \end{aligned} \tag{A.16n}$$

$$\hat{f}_c^C, \hat{p}_c^C \in \{0,1\}, \quad c \in C \tag{A.16o}$$

$$\hat{f}_n^N, \hat{p}_n^N \geq 0, \quad n \in N \tag{A.16p}$$

$$F_n \in \{0,1\}, n \in N \tag{A.16q}$$

$$P_n \in \{0,1\}, n \in N \tag{A.16r}$$

$$q_{ij}^N \in \{0,1\}, \quad i,j \in N, i \neq j \tag{A.16s}$$

$$q_{cn} \in \{0,1\}, \quad c \in C, n \in N \tag{A.16t}$$

$$h \in \{0,1\} \tag{A.16u}$$

The linear formulation for $KEX(\mathcal{U}_\Gamma^w)$ is obtained by adding the PI-TSP constraints to Problem A.16, and mazimizing the objective A.16a.

This linear formulation can be solved by any standard solver; our experiments use Gurobi [155].

## A.3 Robustness as Fairness

In this section we use the framework of edge weight uncertainty to address the problem of fairness in kidney exchange. Though seemingly unrelated, fairness and uncertainty share some key characteristics. The concept of *budgeted uncertainty* balances the nominal objective value with the worst case. A similar trade-off exists between

fairness and efficiency in kidney exchange: allocating kidneys to hard-to-match patients is *fair*, but often reduces the number of possible transplants.

### A.3.1 The Price of Fairness

In kidney exchange, fairness often pertains to *highly-sensitized* patients, who are very unlikely to find a compatible donor. Highly-sensitized patients face longer waiting times than lowly-sensitized patients[1]. In part this is because highly sensitized patients are hard to match; for this reason most kidney exchange optimization algorithms – which maximize matching size or weight – marginalize highly-sensitized patients.

A patient's sensitization level is measured by her Calculated Panel Reactive Antibody (CPRA) score, which ranges from 0 to 100. Patient-donor pair vertices in the exchange graph are highly-sensitized if the pair's patient has a CPRA score above some threshold $\tau$, which is set by policymakers ($\tau = 80$ is common). Let $V_H$ ($V_L$) be the set of highly-sensitized (lowly-sensitized) vertices in $P$, and let $E_H$ ($E_L$) be the set of all edges that end in $V_H$ ($V_L$).

Fairness for a matching $M$ is often quantified using the *utility* assigned to $V_H$ and $V_L$ – i.e., the sum of edge weights into each vertex set,

$$U_H(M) = \sum_{e \in E_H} x_e w_e, \quad U_L(M) = \sum_{e \in E_L} x_e w_e.$$

The *utilitarian* utility function is defined as $u(M) = U_H(M) + U_L(M)$ (i.e., the total edge weight of matching $M$). We might define a *fair* utility function $u_f : \mathcal{M} \to \mathbb{R}$, such that the matching $M_f^*$ that maximizes $u_f$ is considered fair:

$$M_f^* = \arg\max_{M \in \mathcal{M}} u_f(M)$$

[1] https://optn.transplant.hrsa.gov/data/

Fairness is quantified using the *fraction of the fair score* $\%F : M, \mathcal{M} \rightarrow [0,1]$ – i.e., the fraction of the maximum possible utility awarded to highly sensitized patients

$$\%F(M, \mathcal{M}) = U_H(M) / \max_{M' \in \mathcal{M}} U_H(M').$$

Bertsimas et al. [45] defines the *price of fairness* as the "relative system efficiency loss under a fair allocation assuming that a fully efficient allocation is one that maximizes the sum of [participant] utilities." Thus the price of fairness is defined using the set of matchings $\mathcal{M}$, the fair utility function $u_f$, and the utilitarian utility function $u$:

$$\mathsf{POF}((, \mathcal{M}), u_f) = \frac{u\left(M^*\right) - u\left(M_f^*\right)}{u\left(M^*\right)} \tag{A.17}$$

$\mathsf{POF}((, \mathcal{M}), u_f)$ is the relative loss in (utilitarian) efficiency caused by choosing the fair outcome $M_f^*$ rather than the most efficient outcome.

Balancing $\%F$ and $\mathsf{POF}($, i)s a key problem in kidney exchange. Achieving a high degree of fairness (high $\%F$) often incurs a high $\mathsf{POF}($, ;) on the other hand, requiring a low $\mathsf{POF}($, o)ften results in low $\%F$. Dickerson et al. [108] propose two algorithms for enforcing fairness in kidney exchange, and demonstrate that without chains, the price of fairness is low in theory. McElfresh and Dickerson [215] extended this result, finding that adding chains lowers the theoretical price of fairness – eventually to zero; they also propose a fair algorithm that limits the price of fairness.

In the next section we generalize one of the fair algorithm proposed by Dickerson et al. [108] using the framework of budgeted robust optimization, and demonstrate its versatility in balancing fairness and efficiency.

### A.3.2 Fairness Through Robustness

In this section we adapt the concept of budgeted uncertainty to apply budgeted *prioritization* to highly sensitized patients in kidney exchange. To prioritize certain patients over others, we assign each edge $e \in E$ a *priority weight* $\hat{w}_e \in [0, \infty)$, equal to the nominal weight multiplied by a factor $(1 + \alpha_e)$, with $\alpha_e \in [-1, \infty)$. There are many ways to prioritize highly sensitized vertices using priority weights: we may set $\alpha > 0$ for all edges in $E_H$, or we may set $\alpha = -1$ for edges in $E_L$, and so on.

To balance fairness with efficiency it reasonable to *limit* the degree of prioritization. To limit prioritization, we define a *budgeted prioritization set* $\mathcal{P}$, which bounds the sum of absolute differences between each $w_e$ and $\hat{w}_e$; this prioritization set is given in Equation A.18.

$$\mathcal{P}_\Gamma = \left\{ \hat{\mathbf{w}} \mid \hat{w}_e = w_e(1 + \alpha_e), \alpha_e \in [-1, \infty], \sum_{e \in E} \alpha_e w_e \leq \Gamma \right\} \tag{A.18}$$

To prioritize $V_H$, we define $\alpha_e$ differently for each edge $e$. In one type of approach, we prioritize $V_H$ by setting $\alpha_e$ to a constant ($\alpha$) for all $e \in E_H$. This approach is given by $\mathcal{P}^+_\Gamma$, in Equation A.19

$$\mathcal{P}^+_\Gamma = \left\{ \hat{\mathbf{w}} \mid \hat{w}_e = \begin{cases} w_e(1+\alpha) & \text{if } e \in E_H \\ w_e & \text{otherwise} \end{cases}, \alpha \geq 0, \alpha \sum_{e \in E} w_e \leq \Gamma \right\} \tag{A.19}$$

A different type of approach prioritizes $V_H$ by reducing all edges into $E_L$; this approach is given by $\mathcal{P}^-_\Gamma$, in Equation A.20.

$$\mathcal{P}_\Gamma^- = \left\{ \hat{\mathbf{w}} \mid \hat{w}_e = \begin{cases} w_e(1-\alpha) & \text{if } e \in E_L \\ w_e & \text{otherwise} \end{cases}, \alpha \in [0,1], \alpha \sum_{e \in E} w_e \leq \Gamma \right\} \tag{A.20}$$

To apply prioritization to kidney exchange, we either minimize or maximize the kidney exchange objective with respect to $\mathcal{P}$. By choosing $\alpha_e$ and prioritization budget $\Gamma$, this general framework can implement a wide variety of prioritization requirements. Next we show how budgeted prioritization generalizes a previous fair algorithm.

### A.3.3 Weighted Fairness

Weighted fairness was proposed by Dickerson et al. [108] to prioritize highly sensitized patients in kidney exchange. This fair algorithm maximizes the total matching weight, after multiplying all edge weights into highly sensitized patients by a factor $(1+\gamma)$, where parameter $\gamma$ is set by policymakers. Weighted fairness is equivalent to *maximizing* the kidney exchange objective over the budgeted prioritization set $\mathcal{P}^w$, given below. This prioritization set is equivalent to $\mathcal{P}_\Gamma^+$, with prioritization budget $\Gamma$ equal to $\gamma$ times the total weight received by highly sensitized patients.

$$\mathcal{P}_\gamma^w = \left\{ \hat{\mathbf{w}} \mid \hat{w}_e = \begin{cases} w_e(1+\alpha) & \text{if } e \in E_H \\ w_e & \text{otherwise} \end{cases}, \alpha \geq 0, \alpha \sum_{e \in E_H} w_e \leq \gamma \sum_{e \in E_H} w_e \right\} \tag{A.21}$$

Note that the uncertainty budget does not depend on edge weights, and can be written succinctly as Equation A.22.

$$\mathcal{P}_\gamma^w = \left\{ \hat{\mathbf{w}} \mid \hat{w}_e = \begin{cases} w_e(1+\alpha) & \text{if } e \in E_H \\ w_e & \text{otherwise} \end{cases}, 0 \leq \alpha \leq \gamma \right\} \tag{A.22}$$

Weighted fairness is implemented by *maximizing* over priority set $\mathcal{P}_\gamma^w$, as in Problem A.23

$$\max \max_{\hat{w} \in \mathcal{P}_\gamma^w} \hat{w} \cdot x_e \tag{A.23a}$$

$$\mathbf{x} \in \mathcal{M} \tag{A.23b}$$

**Proposition A.3.** *$\gamma$-weighted fairness is equivalent to maximizing the kidney exchange objective over $\mathcal{P}_\gamma^w$.*

As demonstrated in Equation A.21, weighted fairness uses the prioritization budget $\Gamma = \gamma \sum_{e \in E_H} w_e$, which is proportional to the weight received by highly sensitized patients. Thus, we may derive an upper bound on the POF(, f)or $\gamma$-weighted fairness.

**Proposition A.4.** *For $\gamma$-weighted fairness, and some matching $M$ the price of fairness for choosing matching $M$ is bounded above by*

$$POF((,u)_\gamma^w, M) \leq \frac{\gamma}{1+\gamma+U_L(M)/U_H(M)}.$$

*Proof.* Suppose that $\gamma$-weighted fairness chooses matching $M$ over a higher-weight matching $E$. In the worst case, both $F$ and $E$ receive nearly the same *priority weight* under $\gamma$-weighted fairness (within a small perturbation $\epsilon$). Let the utility awarded

by each outcome to highly- and lowly-sensitized patients be given by

$$\begin{aligned} U_H(M) &= A \quad U_L(M) &= B \\ U_H(E) &= 0 \quad\;\; U_L(E) &= A(1+\gamma) + B - \epsilon \end{aligned}$$

with $0 < \epsilon \ll 1$. Both $M$ and $E$ receive nearly the same priority weight from $\gamma$-weighted fairness, but $E$ receives $\gamma A$ more weight than $M$:

$$u^w_\gamma(M) = A(1+\gamma) + B$$

$$u^w_\gamma(E) = A(1+\gamma) + B - \epsilon$$

And thus $\gamma$-weighted fairness selects $M$ over $E$. Taking the limit as $\epsilon \to 0$, the price of fairness for choosing $M$ is

$$\mathsf{POF}((,u)^w_\gamma, M) = \frac{A(1+\gamma) + B - \epsilon - A - B}{A(1+\gamma) + B - \epsilon} = \frac{\gamma}{1+\gamma+B/A},$$

note that $A = U_H(M)$ and $B = U_L(M)$, and thus

$$\mathsf{POF}((,u)^w_\gamma, M) = \frac{\gamma}{1+\gamma+U_L(M)/U_H(M)}.$$

Note that this is the worst-case POF(, f)or choosing $M$, and thus

$$\mathsf{POF}((,u)^w_\gamma, M) \leq \frac{\gamma}{1+\gamma+U_L(M)/U_H(M)}.$$

□

It follows that this POF(, i)s maximized when $U_L(M) = 0$, which is the worst case POF(, f)or $\gamma$-weighted fairness.

**Corollary A.2.1.** *For $\gamma$-weighted fairness, the price of fairness is bounded above by*

$$POF((,u)^{w}_{\gamma}) \leq \frac{\gamma}{1+\gamma}$$

**Proposition A.5.** *Let $U^*_H$ be the maximum possible utility for highly-sensitized patients. For $\gamma$-weighted fairness, and some matching $M$ the fraction of the fair score $\%F$ for matching $M$ is bounded below by*

$$\%F(M, \mathcal{M}) \geq 1 - \frac{U_L(M)}{U^*_H}\frac{1}{1+\gamma}.$$

*Proof.* Let $M \in mM$ be a feasible matching, and let $U^*_H$ be the maximum possible utility for highly-sensitized patients. Consider the worst case scenario for $\gamma$-weighted fairness: two outcomes receive nearly equal utility from $\gamma$-weighted fairness, but the outcome chosen is far less fair. Let the *fair* outcome $F$ assign the maximum possible utility to highly sensitized patients, and zero utility to lowly sensitized patients:

$$u_H(F) = U^*_H, \quad u_L(F) = 0.$$

Let $M$ be the outcome selected by $\gamma$-weighted fairness, which assigns utility $\beta U_H(M)$ to highly sensitized patients, with $0 < \beta < 1$, and some utility $A + \epsilon$ to lowly sensitized patients, with $0 < \epsilon \ll 1$:

$$u_H(M) = \beta U^*_H, \quad u_L(M) = A + \epsilon,$$

and note that $\beta$ is $\%F$, the fraction of the fair score, for outcome $M$.

Letting $\epsilon \to 0$, both $F$ and $M$ receive the same utility under $\gamma$-weighted fairness; that is,

$$U_H^*(1+\gamma) = \beta U_H^*(1+\gamma) + U_L(M).$$

Rearranging, we have

$$\%F(M, \mathcal{M}) = \beta = 1 - \frac{U_L(M)}{U_H^*}\frac{1}{1+\gamma}.$$

This is the worst-case outcome for $\%F$, and thus

$$\%F(M, \mathcal{M}) \geq \beta = 1 - \frac{U_L(M)}{U_H^*}\frac{1}{1+\gamma}.$$

□

It follows that the worst-case $\%F$ occurs when $U_L$ is maximal, and $M = M^*$.

**Corollary A.2.2.** *Let $U_H^*$ and $U_L^*$ be the maximum possible utility for highly- and lowly-sensitized patients, respectively. Under $\gamma$-weighted fairness, the fraction of the fair score $\%F$ is bounded below by*

$$\%F(*, \mathcal{M}) \geq 1 - \frac{U_L^*}{U_H^*}\frac{1}{1+\gamma}.$$

These results may be used to balance $\%F$ and $\mathsf{POF}(,)$ subject to policymaker requirements. For example, suppose policymakers require that $\%F \geq f$, and $\mathsf{POF}(\leq, p)$, for some constants $f$ and $p$. If we know the maximum utility for highly- and lowly- sensitized patients, we can bound $\gamma$ using the worst-case bounds from Corollary A.2.1 and A.2.2. Inverting the bounds from these Corollaries with $p = \mathsf{POF}(,$ a)nd $f = \%f$, we have

$$\gamma \leq \frac{p}{1-p}, \quad \gamma \geq \frac{U_L^*}{U_H^*}\frac{1}{1-f} - 1.$$

Combining these restrictions, we arrive at the bounded prioritization set $\mathcal{P}_p^f$, given in Equation A.24.

$$\mathcal{P}_p^f = \left\{ \hat{\mathbf{w}} \mid \hat{w}_e = \begin{cases} w_e(1+\gamma) & \text{if } e \in E_H \\ w_e & \text{otherwise} \end{cases}, \frac{U_L^*}{U_H^*}\frac{1}{1-f} - 1 \leq \gamma \leq \frac{p}{1-p} \right\} \tag{A.24}$$

There are two important observations about this prioritization set. First, not all choices of $f$ and $p$ are valid, and this depends on $U_L^*/U_H^*$; that is, choosing either $f$ or $p$ necessarily bounds the other. Second, there are many ways to use $\mathcal{P}_p^f$ in practice: we might *minimize* or *maximize* $\hat{\mathbf{w}}$ before maximizing the kidney exchange objective (i.e., setting $\gamma$ to its maximum or minimum value; this is equivalent to the weighted fairness proposed by Dickerson et al. [108].

Alternatively, we might allow $\gamma$ to vary within the range of set by $\mathcal{P}_p^f$. This approach allows the optimization algorithm to *choose* the value of $\gamma$, such that priority weight is maximized. Note that this is not equivalent to weighted fairness (Problem A.23), which maximizes priority weight *before* maximizing the objective. This variable-$\gamma$ approach is given in Problem A.25.

$$\max_{\hat{w} \in \mathcal{P}_p^f} \sum_{e \in E} \hat{w}_e \cdot x_e$$

$$\mathbf{x} \in \mathcal{M}$$

By directly applying the definition of $\hat{w}$ to this problem, we arrive at Problem A.26.

$$\max \quad (1+\gamma) \sum_{e \in E_H} w_e \cdot x_e + \sum_{e \in E_L} w_e \cdot x_e \tag{A.26a}$$

$$\frac{U_L^*}{U_H^*}\frac{1}{1-f} - 1 \leq \gamma \leq \frac{p}{1-p} \tag{A.26b}$$

$$\mathbf{x} \in \mathcal{M} \tag{A.26c}$$

In the next section we tighten this the bound on $\%F$ for $\gamma$-weighted fairness, by relaxing the bounds on $\gamma$.

#### A.3.3.1 Variable Weighted Fairness

The bounds in Corollary A.2.1 and A.2.2 are for the *worst-case* bounds on $\gamma$; however, the worst-case scenarios that produce these bounds may never occur. Instead, we use Proposition A.4 and A.5 to bound $\gamma$ for some feasible matching $M$.

As before, suppose that policymakers require that $\%F \geq f$, and $\mathsf{POF}(\leq, p)$, for some constants $f$ and $p$. If we know the maximum utility for highly-sensitized patients, we can bound $\gamma$ (for some matching $M$) using the worst-case bounds from Proposition A.4 and A.5. Inverting these bounds with $p = \mathsf{POF}(, \text{a})\text{nd } f = \%f$, we have

$$\gamma \leq \frac{p}{1-p}\left(1 + \frac{U_L(M)}{U_H(M)}\right), \quad \gamma \geq \frac{U_L(M)}{U_H^*}\frac{1}{1-f} - 1.$$

Applying these bounds on $\gamma$ results in the following prioritization set $\mathcal{P}_p^f$, given in Equation A.27.

$$\mathcal{P}_p^f = \left\{ \hat{\mathbf{w}} \mid \hat{w}_e = \begin{cases} w_e(1+\gamma) & \text{if } e \in E_H \\ w_e & \text{otherwise} \end{cases}, \frac{U_L(M)}{U_H^*}\frac{1}{1-f} - 1 \leq \gamma \leq \frac{p}{1-p}\left(1 + \frac{U_L(M)}{U_H(M)}\right) \right\} \tag{A.27}$$

As before, we might maximize or minimize the prioritization weight over $\mathcal{P}_f^p$ (i.e., weighted fairness), or allow $\gamma$ to vary within the range of $\mathcal{P}_f^p$. Note that allowing $\gamma$ to vary adds variable inequalities, which depends on the decision variables of $M$.

# Appendix B: Appendix to Chapter 6

## B.1 Price of Fairness in the Random Graph Model

Ashlagi and Roth [24] characterize efficient matchings in a random graph model without chains, and Dickerson et al. [108] build on this to show that the price of fairness without chains is bounded above by 2/33. Dickerson et al. [106] extend the efficient matching of Ashlagi and Roth [24] to include chains, but do not calculate the price of fairness. We close the remaining theory gap regarding the price of fairness with chains. Appendix B.1.1 describes the random graph model, and Appendix B.1.2 presents the theoretical price of fairness with chains.

### B.1.1 Random Graph Model

Let all patient-donor pairs $P$ be partitioned into subsets $V^{X\text{-}Y}$ for each patient blood type $X$ and donor blood type $Y$. These subsets will be further partitioned into lowly- and highly sensitized pairs $V_L^{X\text{-}Y}$ and $V_H^{X\text{-}Y}$. Let $\mu_X$ be the fraction of both patients and donors of each blood type $X$.

Let $N^X$ be the set of NDDs of blood type $X$. Let $\beta|P|$ be the total number of NDDs, with the same blood type distribution as patients. That is, $|N^X| = \beta\mu_X|P|$, with $X \in \{A, B, AB, O\}$.

Patient-donor vertices may be blood-type compatible, but will not be connected

by a directed edge due to tissue-type incompatibility. Let $\bar{p}$ be the fraction of patient-donor pairs that are blood-type-compatible, but tissue-type-incompatible. We refer to certain blood-type vertex subsets of as follows:

1. $V^{\text{A-B}}$ and $V^{\text{B-A}}$: reciprocal pairs
2. $V^{\text{X-X}}$: self-demanded pairs
3. $V^{\text{AB-B}}, V^{\text{AB-A}}, V^{\text{AB-O}}, V^{\text{A-O}}, V^{\text{B-O}}$: over-demanded pairs
4. $V^{\text{A-AB}}, V^{\text{B-AB}}, V^{\text{O-A}}, V^{\text{O-B}}, V^{\text{O-AB}}$: under-demanded pairs

To reflect real-world exchanges, assume $\bar{p} > 1 - \lambda$, $\mu_{\text{O}} > \mu_{\text{A}} > \mu_{\text{B}} > \mu_{\text{AB}}$, and $\bar{p} < 2/5$. WLOG, let $|V^{\text{A-B}}| > |V^{\text{B-A}}|$, and assume that the absolute difference between these pools grows sublinearly with the size of the exchange, that is $|V^{\text{A-B}}| - |V^{\text{B-A}}| = o(n)$.

### B.1.2 The Price of Fairness With Chains

We calculate the price of fairness in this model by exploring all of the possible ways that the efficient matching can proceed, which depends on $\beta$. We state without proof that there are only four possible matchings with nonzero price of fairness, and several matchings with zero price of fairness. It is tedious, but straightforward, to confirm this statement, using the assumptions made while constructing these matchings. Figure B.1 shows each possible matching on this model, and some of the impossible matchings.

Propositions B.2, B.3, B.4, and B.5 give the price of fairness for each of the four matchings with nonzero price of fairness; for each of these cases, $\beta < \mu_{\text{AB}}(1 - \bar{p})$. Proposition B.1 states that the price of fairness is zero when $\beta > \mu_{\text{AB}}(1 - \bar{p})$.

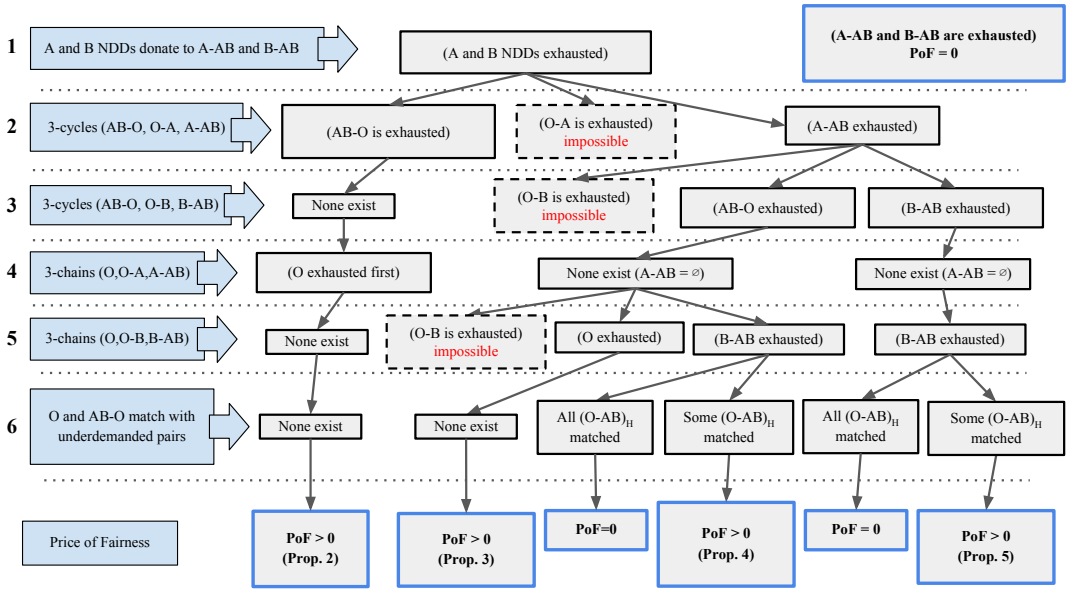



FIGURE B.1: All possible matchings on the random graph model. Boxes with blue borders represent the matching outcomes, and boxes with black borders represent intermediate steps in each matching. Some of the impossible matchings are shown as boxes with dashed black borders.

In all of these matchings, the price of fairness is bounded above by the price of fairness without NDDs, found by Dickerson et al. [108]; Theorem 6.1 states this finding, which uses by Lemmas B.2 and B.3.

Theorem 6.2 states that the price of fairness is zero when $\beta > 1/8$, and Lemmas B.4, B.5, B.6, and B.7 give bounds on $\beta$ for each matching with nonzero price of fairness.

We start with the efficient matching proposed in [106] using cycles and chains up to length 3. This matching may proceed in many different ways, depending on $\beta$. However, most outcomes are impossible based on the canonical assumptions for the random graph model. Figure B.1 shows all possible ways that the matching can proceed.

Lemma B.1 states that even without chains, all highly-sensitized patients except for those in $V^{\text{O-AB}}$ are matched in the efficient matching, only using cycles; this Lemma will be used in all following propositions.

**Lemma B.1.** *Denote by $\mathcal{M}$ the set of matchings in $G(n)$ using cycles and chains up to length 3. As $n \to \infty$, a.s. all highly sensitized pairs can be matched with no efficiency loss under the lexicographic fair algorithm, except for those of type O-AB.*

*(This Lemma uses the same efficient matching introduced by Dickerson [106].)*

*sketch.* Begin with the efficient matching $M^*$ using only cycles up to length 3, proposed by Dickerson in [108]. $M^*$ matches all over-demanded and self-demanded vertices with high probability, but leaves some under-demanded vertices unmatched. We proceed through the initial steps of matching $M^*$ to show that *all* vertices in $V_H^{\text{O-A}}$, $V_H^{\text{O-B}}$, $V_H^{\text{A-AB}}$, and $V_H^{\text{B-AB}}$ are matched.

1. Match all vertices in $V^{\text{B-A}}$ in 2-cycles with $V^{\text{A-B}}$, exhausting $V^{\text{B-A}}$ and leaving $|V^{\text{A-B}}| \propto o(n)$.

2. Match all remaining vertices in $V^{\text{A-B}}$ in 3-cycles with $V^{\text{B-O}}$ and $V^{\text{O-A}}$. There are only $|V^{\text{A-B}}| \propto o(n)$ of these cycles, which will become negligible to the price of fairness as $n \to \infty$.

3. Match all remaining vertices in $V^{\text{A-O}}$ in 2-cycles with $V^{\text{O-A}}$. Note that $|V^{\text{A-O}}| \propto \bar{p}\mu_A\mu_O$ and $|V^{\text{O-A}}| \propto \mu_A\mu_O$. The $V^{\text{A-O}}$ vertices are exhausted first if $|V^{\text{A-O}}| < |V^{\text{O-A}}|$, which holds almost surely because $\bar{p}\mu_A\mu_O < \mu_A\mu_O$ due to the assumption $\bar{p} < 2/5$. All highly sensitized vertices $V_H^{\text{O-A}}$ are matched because $(1-\lambda)\mu_A\mu_O < \bar{p}\mu_A\mu_O$ holds under the assumption $1-\lambda < \bar{p}$. Thus both $V^{\text{A-O}}$ and $V_H^{\text{O-A}}$ are exhausted, and $|V^{\text{O-A}}| \propto \mu_A\mu_O(1-\bar{p})$.

4. Match all remaining vertices in $V^{\text{B-O}}$ in 2-cycles with $V^{\text{O-B}}$. Note that $|V^{\text{B-O}}| \propto \bar{p}\mu_B\mu_O$ and $|V^{\text{O-B}}| \propto \mu_B\mu_O$. As before, the a.s. $|V^{\text{O-B}}| > |V^{\text{B-O}}|$. All highly sensitized vertices $V_H^{\text{O-B}}$ are matched a.s., because $\bar{p}\mu_B\mu_O > (1-\lambda)\mu_B\mu_O$ holds

under the assumption $\bar{p} > 1 - \lambda$. Thus both $V^{\text{B-O}}$ and $V_H^{\text{O-B}}$ are exhausted, and $|V^{\text{O-B}}| \propto \mu_B \mu_O (1 - \bar{p})$.

5. Match all vertices in $V^{\text{AB-A}}$ in 2-cycles with $V^{\text{A-AB}}$. Note that, $|V^{\text{AB-A}}| \propto \bar{p}\mu_A \mu_{\text{AB}}$ and $|V^{\text{A-AB}}| \propto \mu_A \mu_{\text{AB}}$. As before, a.s. $|V^{\text{A-AB}}| > |V^{\text{A-AB}}|$. All highly sensitized vertices $V_H^{\text{A-AB}}$ are matched, because $\bar{p}\mu_A \mu_{\text{AB}} > (1 - \lambda)\mu_A \mu_{\text{AB}}$ under the assumption $\bar{p} > 1 - \lambda$. Thus both $V^{\text{AB-A}}$ and $V_H^{\text{A-AB}}$ are exhausted, and $|V^{\text{A-AB}}| \propto \mu_A \mu_O (1 - \bar{p})$.

6. Match all vertices in $V^{\text{AB-B}}$ in 2-cycles with $V^{\text{B-AB}}$. Note that, $|V^{\text{AB-B}}| \propto \bar{p}\mu_B \mu_{\text{AB}}$ and $|V^{\text{B-AB}}| \propto \mu_B \mu_{\text{AB}}$, and a.s. $|V^{\text{B-AB}}| > |V^{\text{AB-B}}|$. All highly sensitized vertices $V_H^{\text{B-AB}}$ are matched, because $\bar{p}\mu_B \mu_{\text{AB}} > (1 - \lambda)\mu_B \mu_{\text{AB}}$ under the assumption $\bar{p} > 1 - \lambda$. Thus both $V^{\text{AB-B}}$ and $V_H^{\text{B-AB}}$ are exhausted, and $|V^{\text{B-AB}}| \propto \mu_B \mu_O (1 - \bar{p})$.

Thus, these initial steps of matching $M^*$ exhaust all highly sensitized pairs in $V_H^{\text{O-A}}$, $V_H^{\text{O-B}}$, $V_H^{\text{A-AB}}$, and $V_H^{\text{B-AB}}$.

□

With uniform edge weights, lexicographic fairness requires that we match the maximum possible number of highly sensitized vertices. Lemma B.1 states that the efficient matching $M^*$ includes all highly sensitized patients, except for those in $V^{\text{O-AB}}$. Therefore all efficiency loss–and price of fairness–is caused by matching vertices in $V_H^{\text{O-AB}}$.

Using both chains and cycles increases overall efficiency. In the dense graph model used in this Appendix, adding chains can only decrease the price of fairness.

Proposition 1 in [108] states that with only cycles up to length 3, and assuming $\bar{p} > 1 - \lambda$, and $\mu_{\text{O}} < 3\mu_{\text{A}}/2$, and $\mu_{\text{O}} > \mu_{\text{A}} > \mu_{\text{B}} > \mu_{\text{AB}}$, the price of fairness is

at most $\frac{2}{33}$. In the dense graph model used here, adding chains tightens this upper bound.

The following propositions tighten the upper bound on the price of fairness, for every possible value of $\beta$.

**Proposition B.1.** *Assume*

***1*** $\beta > (1-\bar{p})\mu_{AB}$.

*Denote by $\mathcal{M}$ the set of matchings in $G(n)$ using cycles and chains up to length 3. As $n \to \infty$, almost surely $\mathit{POF}(\mathcal{M}, u_{LEX}) = 0$.*

*sketch.* We begin by executing the initial steps of matching $M^*$ as done in the proof of Lemma B.1, matching all highly sensitized vertices except for those in $V_H^{\text{O-AB}}$. The following steps continue the matching $M^*$ from Lemma B.1.

7. A- and B-type NDDs donate to $V^{\text{A-AB}}$ and $V^{\text{B-AB}}$, respectively. Note that $|N^A| \propto \beta\mu_A$ and $|V^{\text{A-AB}}| \propto (1-\bar{p})\mu_A\mu_{\text{AB}}$. Assuming $\beta > (1-\bar{p})\mu_{\text{AB}}$, the inequality $\beta\mu_{\text{A}} > \mu_A\mu_{\text{AB}}(1-\bar{p})$ holds and a.s. $|N^{\text{A}}| > |V^{\text{A-AB}}|$. By the same argument, a.s. $|N^{\text{B}}| > |V^{\text{B-AB}}|$. Thus, both $V^{\text{A-AB}}$ and $V^{\text{B-AB}}$ are exhausted, and $|N^{\text{B}}| \propto \mu_{\text{B}}\left(\beta - (1-\bar{p})\mu_{\text{AB}}\right)$ and $|N^{\text{A}}| \propto \mu_{\text{A}}\left(\beta - (1-\bar{p})\mu_{\text{AB}}\right)$.

8. Create cycles of the form (AB-O, O-X, X-AB), with X $\in \{\text{A}, \text{B}\}$. None of these cycles occur because both $V^{\text{A-AB}}$ and $V^{\text{B-AB}}$ have been exhausted in previous steps.

9. Create chains of the form (O,O-X,X-AB), with X $\in \{\text{A}, \text{B}\}$. None of these cycles occur, because both $V^{\text{A-AB}}$ and $V^{\text{B-AB}}$ have been exhausted in previous steps.

10. Remaining O-type NDDs donate to remaining under-demanded vertices, beginning with $V^{\text{O-AB}}$. Note that no O-type NDDs have been used in previous steps, so $|N^{\text{O}}| \propto \beta\mu_{\text{O}}$.

11. 2-cycles are created with $V^{\text{AB-O}}$ and remaining under-demanded vertices, beginning with $V^{\text{O-AB}}$. Note that no vertices in $V^{\text{AB-O}}$ have been used in previous steps, so $|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{O}}\mu_{\text{AB}}$.

The final two steps match up to $|V^{\text{AB-O}}| + |N^{\text{O}}| \propto \beta\mu_{\text{O}} + \bar{p}\mu_{\text{O}}\mu_{\text{AB}}$ vertices in $V^{\text{O-AB}}$. The only remaining highly-sensitized vertices are in $V_H^{|}OAB \propto (1-\lambda)\mu_{\text{O}}\mu_{\text{AB}}$. Assuming that $\bar{p} > 1-\lambda$, the inequality $\beta\mu_{\text{O}} + \bar{p}\mu_{\text{O}}\mu_{\text{AB}} > \bar{p}\mu_{\text{O}}\mu_{\text{AB}} > (1-\lambda)\mu_{\text{O}}\mu_{\text{AB}}$ holds, and a.s. $|V^{\text{AB-O}}| + |N^{\text{O}}| > |V_H^{|}OAB|$. This exhausts all vertices in $|V_H^{\text{O-AB}}|$. All other highly-sensitized vertices were matched in steps 1-6 of, as in Lemma B.1. Thus, all highly sensitized vertices can be matched with no efficiency loss, and the price of fairness is zero.

□

PropositionB.1 assumes that $\beta$ is extremely large, specifically $\beta > 1/4 > (1-\bar{p})\mu_{\text{AB}}$. In practice, $\beta < 0.01$ – that is, the number of NDDs in an exchange is often less than 1% of the size of the exchange. The following Propositions address the price of fairness when $\beta < (1-\bar{p})\mu_{\text{AB}} < 1/4$.

**Proposition B.2.** *Assume*

***A.1*** $\beta < \mu_A(1-\bar{p}) - \bar{p}\mu_{AB}$

***A.2*** $\beta < \mu_{AB}(1-\bar{p}) - \bar{p}\mu_{AB}\mu_O/\mu_A$

***A.3*** $\beta < \mu_{AB}\left(\frac{\mu_A}{\mu_A+\mu_O} - \bar{p}\right)$

*These constraints imply $\beta \in [0, 1/8]$.Denote by $\mathcal{M}$ the set of matchings in $G(n)$ using cycles and chains up to length 3. Almost surely as $n \to \infty$, the price of fairness is*

$$POF(\mathcal{M}, u_{LEX}) = \frac{(1-\lambda)\mu_O\mu_{AB}}{u_E}$$

*with*

$$u_E = \bar{p}\Big[2\mu_{AB}\mu_B + 2\mu_{AB}\mu_A + 3\mu_{AB}\mu_O$$
$$+2\mu_A\mu_O + 2\mu_B\mu_O + \mu_O^2 + \mu_A^2 + \mu_B^2 + \mu_{AB}^2\Big]$$
$$+2\mu_A\mu_B + \beta\left(\mu_A + \mu_B + 2\mu_O\right)$$

*sketch.* We begin with matching $M^*$ as done in the proof of Lemma B.1, matching all highly sensitized vertices except for those in $V_H^{\text{AB-O}}$. We now complete the efficient matching using both 3-cycles and 3-chains as in [106].

7. A- and B-type NDDs donate to $V^{\text{A-AB}}$ and $V^{\text{B-AB}}$, respectively. Note that $|N^A| \propto \beta\mu_A$ and $|V^{\text{A-AB}}| \propto (1-\bar{p})\mu_A\mu_{\text{AB}}$. The inequality $\beta\mu_{\text{A}} < \mu_A\mu_{\text{AB}}(1-\bar{p})$ holds due to assumption **A.2**, and a.s. $|N^{\text{A}}| < |V^{\text{A-AB}}|$. By the same argument, a.s. $|N^{\text{B}}| < |V^{\text{B-AB}}|$. Thus, both $N^{\text{A}}$ and $N^{\text{B}}$ are exhausted, and $|V^{\text{A-AB}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{A}}$ and $|V^{\text{B-AB}}| \propto \mu_{\text{B}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{B}}$.

8. Create cycles of the form (AB-O, O-A, A-AB). The current size of each vertex group is

   (1) $|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$

   (2) $|V^{\text{O-A}}| \propto (1-\bar{p})\mu_{\text{A}}\mu_{\text{O}}$

   (3) $|V^{\text{A-AB}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{A}}$

   The inequality $(1) < (2)$ holds due to the model assumptions, so a.s. $|V^{\text{AB-O}}| < |V^{\text{O-A}}|$. Note that the inequality $(1) < (3)$ can be written as

$$\beta < \mu_{\text{AB}}(1-\bar{p}) - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}/\mu_{\text{A}}$$

which holds by assumptions **A.2**, and a.s. $|V^{\text{AB-O}}| < |V^{\text{A-AB}}|$. Executing these cycles exhausts $V^{\text{AB-O}}$ and leaves the following vertices remaining

(1) $|V^{\text{O-A}}| \propto (1-\bar{p})\mu_{\text{A}}\mu_{\text{O}} - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$.

(2) $|V^{\text{A-AB}}| \propto (1-\bar{p})\mu_{\text{A}}\mu_{\text{AB}} - \bar{p}\mu_{\text{AB}}\mu_{\text{O}} - \beta\mu_{\text{A}}$

9. Create cycles of the form (AB-O, O-B, B-AB). The previous step exhausted $V^{\text{AB-O}}$, so none of these cycles occur.

10. Create chains of the form (O,O-A,A-AB). The current size of each vertex group is

(1) $|N^{\text{O}}| \propto \beta\mu_{\text{O}}$

(2) $|V^{\text{O-A}}| \propto (1-\bar{p})\mu_{\text{A}}\mu_{\text{O}} - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$

(3) $|V^{\text{A-AB}}| \propto (1-\bar{p})\mu_{\text{A}}\mu_{\text{AB}} - \bar{p}\mu_{\text{AB}}\mu_{\text{O}} - \beta\mu_{\text{A}}$

The inequality $(1) < (2)$ holds due to assumption **A.1**, so a.s. $|N^{\text{O}}| < |V^{\text{O-A}}|$. Note that inequality $(1) < (3)$ can be written as

$$\beta < \mu_{\text{AB}}\left(\frac{\mu_{\text{A}}}{\mu_{\text{A}}+\mu_{\text{O}}} - \bar{p}\right)$$

which holds due to **A.3**. Thus a.s. $|N^{\text{O}}| < |V^{\text{A-AB}}|$, and $|N^{\text{O}}|$ is exhausted. The vertices unmatched by these chains are

(1) $|V^{\text{O-A}}| \propto (1-\bar{p})\mu_{\text{A}}\mu_{\text{O}} - \bar{p}\mu_{\text{AB}}\mu_{\text{O}} - \beta\mu_{\text{O}}$

(2) $|V^{\text{A-AB}}| \propto (1-\bar{p})\mu_{\text{A}}\mu_{\text{AB}} - \bar{p}\mu_{\text{AB}}\mu_{\text{O}} - \beta\left(\mu_{\text{A}}+\mu_{\text{O}}\right)$

11. Remaining O-type NDDs donate to remaining under-demanded vertices. The previous step exhausted $N^{\text{O}}$, so none of these donations occur.

12. 2-cycles are created with $V^{\text{AB-O}}$ and remaining under-demanded vertices. The previous step exhausted $V^{\text{AB-O}}$, so none of these cycles occur.

In the efficient matching described above, the number of *matched* pairs in each under-demanded group is

$$|V^{\text{O-A}}| \propto \mu_{\text{O}} \left(\beta + \bar{p}\left(\mu_{\text{A}} + \mu_{\text{AB}}\right)\right)$$

$$|V^{\text{O-B}}| \propto \bar{p}\mu_{\text{B}}\mu_{\text{O}}$$

$$|V^{\text{A-AB}}| \propto (\beta + \bar{p}\mu_{\text{AB}})(\mu_{\text{A}} + \mu_{\text{O}})$$

$$|V^{\text{B-AB}}| \propto (\beta + \bar{p}\mu_{\text{AB}})\mu_{\text{B}}$$

$$|V^{\text{O-AB}}| = 0$$

Combining these with the over-demanded and self-demanded vertices, the total size of the efficient matching is

$$\begin{aligned} u_E = \bar{p}\Big[&2\mu_{\text{AB}}\mu_{\text{B}} + 2\mu_{\text{AB}}\mu_{\text{A}} + 3\mu_{\text{AB}}\mu_{\text{O}} \\ &+2\mu_{\text{A}}\mu_{\text{O}} + 2\mu_{\text{B}}\mu_{\text{O}} + \mu_{\text{O}}^2 + \mu_{\text{A}}^2 + \mu_{\text{B}}^2 + \mu_{\text{AB}}^2\Big] \\ &+2\mu_{\text{A}}\mu_{\text{B}} + \beta\left(\mu_{\text{A}} + \mu_{\text{B}} + 2\mu_{\text{O}}\right) \end{aligned}$$

This efficient matching includes all highly sensitized vertices except for those in $V_H^{\text{O-AB}}$. To calculate the price of fairness we now find the size of the fair matching. We match each vertex in $V_H^{\text{O-AB}}$ by removing a 3-cycle of the form (AB-O, O-A, A-AB) and creating a 2-cycle (AB-O, O-AB). This matching used $|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{O}}\mu_{\text{AB}}$ 3-cycles of this form, while $|V_H^{\text{O-AB}}| \propto (1-\lambda)\mu_{\text{O}}\mu_{\text{AB}}$. The model assumption $\bar{p} > 1 - \lambda$ ensures that $|V^{\text{AB-O}}| > |V_H^{\text{O-AB}}|$, and all vertices in $V_H^{\text{O-AB}}$ can be matched in this way.

To match each vertex in $V_H^{\text{O-AB}}$, we remove from the matching one vertex from both $V^{\text{O-A}}$ and $V^{\text{A-AB}}$. Thus the total efficiency loss is $|V_H^{\text{O-AB}}| \propto (1-\lambda)\mu_{\text{O}}\mu_{\text{AB}}$. The price of fairness is

$$\mathsf{POF}(\mathcal{M}, u_{LEX}) = \frac{(1-\lambda)\mu_{\text{O}}\mu_{\text{AB}}}{u_E}$$

With $u_E$ defined previously. □

**Proposition B.3.** *Assume*

***1*** $\beta < \mu_{AB}(1-\bar{p}) - \mu_{AB}\mu_O\bar{p}/(\mu_A+\mu_B)$

***2*** $\beta < \frac{\mu_A\mu_{AB}(1-\bar{p})+\mu_B\mu_O(1-\bar{p})-\bar{p}\mu_O\mu_{AB}}{\mu_A+\mu_O}$

***3*** $\beta > \mu_{AB}(1-\bar{p}) - \mu_{AB}\mu_O\bar{p}/\mu_A$

***4*** $\beta < \mu_{AB}(1-\bar{p}) - \bar{p}\mu_{AB}\mu_O/\mu_A + (1-\bar{p})\mu_B\mu_O/\mu_A$

***5*** $\beta < \mu_{AB}(1-\bar{p}) - \mu_O\mu_{AB}/(1-\mu_{AB})$

*Note that as written, constraint **4** is a looser bound than **5**, and can be removed. However it is convenient to leave **4** for clarity. These constraints imply $\beta \in [0, 1/12]$. Denote by $\mathcal{M}$ the set of matchings in $G(n)$ using cycles and chains up to length 3. Almost surely as $n \to \infty$, the price of fairness is*

$$\mathit{POF}(\mathcal{M}, u_{LEX}) = \frac{(1-\lambda)\mu_O\mu_{AB}}{u_E}$$

*with*

$$u_E = \bar{p}\Big[2\mu_{AB}\mu_B + 2\mu_{AB}\mu_A + 3\mu_{AB}\mu_O$$
$$+2\mu_A\mu_O + 2\mu_B\mu_O + \mu_O^2 + \mu_A^2 + \mu_B^2 + \mu_{AB}^2\Big]$$
$$+2\mu_A\mu_B + \beta\left(\mu_A + \mu_B + 2\mu_O\right)$$

*sketch.* We begin with matching $M^*$ as done in the proof of Lemma B.1, matching all highly sensitized vertices except for those in $V_H^{\text{AB-O}}$. We now complete the efficient matching using both 3-cycles and 3-chains as in [106].

7. A- and B-type NDDs donate to $V^{\text{A-AB}}$ and $V^{\text{B-AB}}$, respectively. Note that $|N^A| \propto \beta\mu_A$ and $|V^{\text{A-AB}}| \propto (1-\bar{p})\mu_A\mu_{\text{AB}}$. The inequality $\beta\mu_{\text{A}} < \mu_A\mu_{\text{AB}}(1-\bar{p})$ holds due to assumption **A.1**, and a.s. $|N^{\text{A}}| < |V^{\text{A-AB}}|$. By the same argument, a.s. $|N^{\text{B}}| < |V^{\text{B-AB}}|$. Thus, both $N^{\text{A}}$ and $N^{\text{B}}$ are exhausted, and $|V^{\text{A-AB}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{A}}$ and $|V^{\text{B-AB}}| \propto \mu_{\text{B}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{B}}$.

8. Create cycles of the form (AB-O, O-A, A-AB). The current size of each vertex group is

   (1) $|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$

   (2) $|V^{\text{O-A}}| \propto (1-\bar{p})\mu_{\text{A}}\mu_{\text{O}}$

   (3) $|V^{\text{A-AB}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{A}}$

   Note that the inequality $(3) < (1)$ can be written as

   $$\beta > \mu_{\text{AB}}(1-\bar{p}) - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}/\mu_{\text{A}}$$

   which holds by assumption **3**, and a.s. $|V^{\text{A-AB}}| < |V^{\text{AB-O}}|$. The inequality $(3) < (2)$ can be written as

$$\beta > (1-\bar{p})(\mu_{\text{AB}} - \mu_{\text{O}})$$

which holds by model assumptions, and a.s. $|V^{\text{A-AB}}| < |V^{\text{O-A}}|$. Executing these cycles exhausts $V^{\text{A-AB}}$ and leaves the following vertices remaining

$$|V^{\text{O-A}}| \propto (1-\bar{p})\mu_{\text{A}}(\mu_{\text{O}} - \mu_{\text{AB}}) + \mu_{\text{A}}\beta$$

$$|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{AB}}\mu_{\text{O}} - \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + \mu_{\text{A}}\beta$$

9. Create cycles of the form (AB-O, O-B, B-AB). The current size of each vertex group is

   (1) $|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{AB}}\mu_{\text{O}} - \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + \mu_{\text{A}}\beta$

   (2) $|V^{\text{O-B}}| \propto (1-\bar{p})\mu_{\text{B}}\mu_{\text{O}}$

   (3) $|V^{\text{B-AB}}| \propto \mu_{\text{B}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{B}}$

   Inequality $(1) < (2)$ can be written as

$$\beta < \mu_{\text{AB}}(1-\bar{p}) - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}/\mu_{\text{A}} + (1-\bar{p})\mu_{\text{B}}\mu_{\text{O}}/\mu_{A}$$

   which holds by assumption **4**. Inequality $(1) < (3)$ can be written as

$$\beta < \mu_{\text{AB}}(1-\bar{p}) - \mu_{\text{AB}}\mu_{\text{O}}\bar{p}/(\mu_{\text{A}} + \mu_{\text{B}})$$

   which holds by assumption **1**.

   Executing these cycles exhausts $V^{\text{AB-O}}$ and leaves the following vertices remaining

$$|V^{\text{O-B}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + (\mu_{\text{B}} - \bar{p}(\mu_{\text{AB}} + \mu_{\text{B}}))\mu_{\text{O}} - \beta\mu_{\text{A}}$$

$$|V^{\text{B-AB}}| \propto ((1-\bar{p})\mu_{\text{AB}} - \beta)\,(\mu_{\text{A}} + \mu_{\text{B}}) - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$$

10. Create chains of the form (O,O-A,A-AB). Previous steps exhausted $V^{\text{A-AB}}$ so none of these chains occur.

11. Create chains of the form (O,O-B,B-AB). The current size of each vertex group is

    (1) $|N^{\text{O}}| \propto \beta\mu_{\text{O}}$

    (2) $|V^{\text{O-B}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + (\mu_{\text{B}} - \bar{p}(\mu_{\text{AB}} + \mu_{\text{B}}))\mu_{\text{O}} - \beta\mu_{\text{A}}$

    (3) $|V^{\text{B-AB}}| \propto ((1-\bar{p})\mu_{\text{AB}} - \beta)\,(\mu_{\text{A}} + \mu_{\text{B}}) - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$

    The inequality $(1) < (2)$ can be written as

$$\beta < \frac{\mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + \mu_{\text{B}}\mu_{\text{O}}(1-\bar{p}) - \bar{p}\mu_{\text{O}}\mu_{\text{AB}}}{\mu_{\text{A}} + \mu_{\text{O}}}$$

    which holds by assumption 2. The inequality $(1) < (3)$ can be written as

$$\beta < \mu_{\text{AB}}(1-\bar{p}) - \mu_{\text{O}}\mu_{\text{AB}}/(1-\mu_{\text{AB}})$$

    which holds by assumption 5. Executing these chains exhausts $N^{\text{O}}$ and leaves the following vertices remaining

$$|V^{\text{O-B}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + (\mu_{\text{B}} - \bar{p}(\mu_{\text{AB}} + \mu_{\text{B}}))\mu_{\text{O}} - \beta(\mu_{\text{A}} + \mu_{\text{O}})$$

$$|V^{\text{B-AB}}| \propto (1-\bar{p})\mu_{\text{AB}} - \beta)(\mu_{\text{A}} + \mu_{\text{B}}) - (\beta + \bar{p}\mu_{\text{AB}})\mu_{\text{O}}$$

12. Remaining O-type NDDs donate to remaining under-demanded vertices. The previous step exhausted $N^{\text{O}}$, so none of these donations occur.

13. 2-cycles are created with $V^{\text{AB-O}}$ and remaining under-demanded vertices. The previous steps exhausted $V^{\text{AB-O}}$, so none of these cycles occur.

In the efficient matching described above, the number of *matched* pairs in each under-demanded group is

$$|V^{\text{O-A}}| \propto (1-\bar{p})\mu_{\text{A}}(\mu_{\text{O}} - \mu_{\text{AB}}) + \mu_{\text{A}}\beta$$

$$\begin{aligned}|V^{\text{O-B}}| \propto{}& \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + (\mu_{\text{B}} - \bar{p}(\mu_{\text{AB}} + \mu_{\text{B}}))\mu_{\text{O}} \\ &- \beta(\mu_{\text{A}} + \mu_{\text{O}})\end{aligned}$$

$$|V^{\text{A-AB}}| = 0$$

$$|V^{\text{B-AB}}| \propto (1-\bar{p})\mu_{\text{AB}} - \beta)(\mu_{\text{A}} + \mu_{\text{B}}) - (\beta + \bar{p}\mu_{\text{AB}})\mu_{\text{O}}$$

$$|V^{\text{O-AB}}| = 0$$

Combining these with the over-demanded and self-demanded vertices, the total size of the efficient matching is

$$\begin{aligned}u_E = \bar{p}\Big[&2\mu_{\text{AB}}\mu_{\text{B}} + 2\mu_{\text{AB}}\mu_{\text{A}} + 3\mu_{\text{AB}}\mu_{\text{O}} \\ &+2\mu_{\text{A}}\mu_{\text{O}} + 2\mu_{\text{B}}\mu_{\text{O}} + \mu_{\text{O}}^2 + \mu_{\text{A}}^2 + \mu_{\text{B}}^2 + \mu_{\text{AB}}^2\Big] \\ &+2\mu_{\text{A}}\mu_{\text{B}} + \beta\left(\mu_{\text{A}} + \mu_{\text{B}} + 2\mu_{\text{O}}\right)\end{aligned}$$

This efficient matching includes all highly sensitized vertices except for those in $V_H^{\text{O-AB}}$. To calculate the price of fairness we now find the size of the fair matching. We match each vertex in $V_H^{\text{O-AB}}$ by removing a 3-cycle of the form (AB-O, O-A, A-AB) and creating a 2-cycle (AB-O, O-AB). This matching used $|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{O}}\mu_{\text{AB}}$ 3-cycles of this form, while $|V_H^{\text{O-AB}}| \propto (1-\lambda)\mu_{\text{O}}\mu_{\text{AB}}$. The model assumptions ensure that $|V^{\text{AB-O}}| > |V_H^{\text{O-AB}}|$, and all vertices in $V_H^{\text{O-AB}}$ can be matched in this way.

To match each vertex in $V_H^{\text{O-AB}}$, we remove from the matching one vertex from both $V^{\text{O-A}}$ and $V^{\text{A-AB}}$. Thus the total efficiency loss is $|V_H^{\text{O-AB}}| \propto (1-\lambda)\mu_{\text{O}}\mu_{\text{AB}}$. The price of fairness is

$$\mathsf{POF}(\mathcal{M}, u_{LEX}) = \frac{(1-\lambda)\mu_{\text{O}}\mu_{\text{AB}}}{u_E}$$

With $u_E$ defined previously. □

**Proposition B.4.** *Assume*

***1*** $\beta > \mu_{AB}(1-\bar{p}) - \mu_{AB}\mu_O\bar{p}/\mu_A$

***2*** $\beta < \mu_{AB}(1-\bar{p}) - \mu_{AB}\mu_O\bar{p}/(\mu_A+\mu_B)$

***3*** $\beta < \mu_{AB}(1-\bar{p}) - \bar{p}\mu_{AB}\mu_O/\mu_A + (1-\bar{p})\mu_B\mu_O/\mu_A$

***4*** $\beta > \mu_{AB}\left((1-\bar{p}) - \frac{\mu_O}{1-\mu_{AB}}\right)$

***5*** $\beta < \mu_{AB}(1-\bar{p}) - \lambda\mu_O\frac{\mu_{AB}}{1-\mu_{AB}}$

*These constraints imply $\beta \in [0, 1/8]$. Denote by $\mathcal{M}$ the set of matchings in $G(n)$ using cycles and chains up to length 3. Almost surely as $n \to \infty$, the price of fairness is*

$$POF(\mathcal{M}, u_{LEX}) = \frac{(1-\mu_{AB})((1-\bar{p})\mu_{AB} - \beta) - \lambda\mu_{AB}\mu_O}{u_E}$$

*with*

$$\begin{aligned} u_E = {} & \mu_{AB}\mu_B + \mu_A\left(\mu_{AB} + 2\mu_B\right) + \beta\mu_O \\ & + \bar{p}\left[\mu_A^2 + \mu_A\mu_{AB} + \mu_{AB}^2 + \mu_{AB}\mu_B + \mu_B^2\right. \\ & \left. + 2(\mu_A + \mu_{AB} + \mu_B)\mu_O + \mu_O^2\right] \end{aligned}$$

*sketch.* We begin with matching $M^*$ as done in the proof of Lemma B.1, matching all highly sensitized vertices except for those in $V_H^{\text{AB-O}}$. We now complete the efficient matching using both 3-cycles and 3-chains as in [106].

7. A- and B-type NDDs donate to $V^{\text{A-AB}}$ and $V^{\text{B-AB}}$, respectively. Note that $|N^A| \propto \beta\mu_A$ and $|V^{\text{A-AB}}| \propto (1-\bar{p})\mu_A\mu_{\text{AB}}$. The inequality $\beta\mu_{\text{A}} < \mu_A\mu_{\text{AB}}(1-\bar{p})$ holds due to assumption 2, and a.s. $|N^{\text{A}}| < |V^{\text{A-AB}}|$. By the same argument, a.s. $|N^{\text{B}}| < |V^{\text{B-AB}}|$. Thus, both $N^{\text{A}}$ and $N^{\text{B}}$ are exhausted, and $|V^{\text{A-AB}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{A}}$ and $|V^{\text{B-AB}}| \propto \mu_{\text{B}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{B}}$.

8. Create cycles of the form (AB-O, O-A, A-AB). The current size of each vertex group is

   (1) $|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$

   (2) $|V^{\text{O-A}}| \propto (1-\bar{p})\mu_{\text{A}}\mu_{\text{O}}$

   (3) $|V^{\text{A-AB}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{A}}$

   Note that the inequality $(3) < (1)$ can be written as

$$\beta > \mu_{\text{AB}}(1-\bar{p}) - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}/\mu_{\text{A}}$$

which holds by assumption **1**, and a.s. $|V^{\text{A-AB}}| < |V^{\text{AB-O}}|$. The inequality $(3) < (2)$ can be written as

$$\beta > (1-\bar{p})(\mu_{\text{AB}} - \mu_{\text{O}})$$

which holds by model assumptions, and a.s. $|V^{\text{A-AB}}| < |V^{\text{O-A}}|$. Executing these cycles exhausts $V^{\text{A-AB}}$ and leaves the following vertices remaining

$$|V^{\text{O-A}}| \propto (1-\bar{p})\mu_{\text{A}}(\mu_{\text{O}} - \mu_{\text{AB}}) + \mu_{\text{A}}\beta$$

$$|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{AB}}\mu_{\text{O}} - \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + \mu_{\text{A}}\beta$$

9. Create cycles of the form (AB-O, O-B, B-AB). The current size of each vertex group is

   (1) $|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{AB}}\mu_{\text{O}} - \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + \mu_{\text{A}}\beta$

   (2) $|V^{\text{O-B}}| \propto (1-\bar{p})\mu_{\text{B}}\mu_{\text{O}}$

   (3) $|V^{\text{B-AB}}| \propto \mu_{\text{B}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{B}}$

   Inequality $(1) < (2)$ can be written as

$$\beta < \mu_{\text{AB}}(1-\bar{p}) - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}/\mu_{\text{A}} + (1-\bar{p})\mu_{\text{B}}\mu_{\text{O}}/\mu_{A}$$

   which holds by assumption **3**. Inequality $(1) < (3)$ can be written as

$$\beta < \mu_{\text{AB}}(1-\bar{p}) - \mu_{\text{AB}}\mu_{\text{O}}\bar{p}/(\mu_{\text{A}} + \mu_{\text{B}})$$

which holds by assumption 2.

Executing these cycles exhausts $V^{\text{AB-O}}$ and leaves the following vertices remaining

$$|V^{\text{O-B}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + (\mu_{\text{B}} - \bar{p}(\mu_{\text{AB}} + \mu_{\text{B}}))\mu_{\text{O}} - \beta\mu_{\text{A}}$$

$$|V^{\text{B-AB}}| \propto ((1-\bar{p})\mu_{\text{AB}} - \beta)\,(\mu_{\text{A}} + \mu_{\text{B}}) - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$$

10. Create chains of the form (O,O-A,A-AB). Previous steps exhausted $V^{\text{A-AB}}$ so none of these chains occur.

11. Create chains of the form (O,O-B,B-AB). The current size of each vertex group is

    (1) $|N^{\text{O}}| \propto \beta\mu_{\text{O}}$

    (2) $|V^{\text{O-B}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + (\mu_{\text{B}} - \bar{p}(\mu_{\text{AB}} + \mu_{\text{B}}))\mu_{\text{O}} - \beta\mu_{\text{A}}$

    (3) $|V^{\text{B-AB}}| \propto ((1-\bar{p})\mu_{\text{AB}} - \beta)\,(\mu_{\text{A}} + \mu_{\text{B}}) - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$

    The inequality $(3) < (1)$ can be written as

$$\beta > \mu_{\text{AB}}\left((1-\bar{p}) - \frac{\mu_{\text{O}}}{1-\mu_{\text{AB}}}\right)$$

    which holds by assumption 4. The inequality $(3) < (2)$ can be written as

$$\beta > (1-\bar{p})(\mu_{\text{AB}} - \mu_{\text{O}})$$

    which holds by the model assumptions. Executing these chains exhausts $V^{\text{B-AB}}$ and leaves the following vertices remaining

$$|N^{\text{O}}| \propto (\beta + \bar{p}\mu_{\text{AB}})(\mu_{\text{A}} + \mu_{\text{B}} + \mu_{\text{O}}) - \mu_{\text{AB}}(\mu_{\text{A}} + \mu_{\text{B}})$$

$$|V^{\text{O-B}}| \propto \mu_{\text{B}} \left( (\beta + (1 - \bar{p})(\mu_{\text{O}} - \mu_{\text{AB}}) \right)$$

12. Remaining O-type NDDs and $V^{\text{AB-O}}$ vertices match with remaining under-demanded vertices, starting with $V^{\text{O-AB}}$. The remaining size of each vertex group is

    (1) $|N^{\text{O}}| \propto (\beta + \bar{p}\mu_{\text{AB}})(\mu_{\text{A}} + \mu_{\text{B}} + \mu_{\text{O}}) - \mu_{\text{AB}}(\mu_{\text{A}} + \mu_{\text{B}})$

    (2) $|V^{\text{O-AB}}| \propto \mu_{\text{AB}}\mu_{\text{O}}$

    (3) $|V^{\text{AB-O}}| = 0$

    After simplifying, the inequality $(1) < (2)$ can be written as

    $$\beta < \mu_{\text{AB}}(1 - \bar{p})$$

    which holds by assumption 2. Thus O-type NDDs are exhausted first, leaving some vertices remaining in $V^{\text{O-AB}}$, with

    $$|V^{\text{O-AB}}| \propto \left( (1 - \bar{p})\mu_{\text{AB}} - \beta \right) (1 - \mu_{\text{AB}})$$

In the efficient matching described above, the number of *matched* pairs in each under-demanded group is

$$|V^{\text{O-A}}| \propto \mu_{\text{A}} \left( \mu_{\text{AB}}(1 - \bar{p}) + \bar{p}\mu_{\text{O}} - \beta \right)$$

$$|V^{\text{O-B}}| \propto \mu_{\text{B}} \left( \mu_{\text{AB}}(1 - \bar{p}) + \bar{p}\mu_{\text{O}} - \beta \right)$$

$$|V^{\text{A-AB}}| \propto \mu_{\text{A}}\mu_{\text{AB}}$$

$$|V^{\text{B-AB}}| \propto \mu_{\text{B}}\mu_{\text{AB}}$$

$$|V^{\text{O-AB}}| \propto (\beta + \mu_{\text{AB}}\bar{p})(1 - \mu_{\text{AB}}) - \mu_{\text{AB}}(\mu_{\text{A}} + \mu_{\text{B}})$$

Combining these with the over-demanded and self-demanded vertices, the total size of the efficient matching is

$$\begin{aligned} u_E = {} & \mu_{\text{AB}}\mu_{\text{B}} + \mu_{\text{A}}\left(\mu_{\text{AB}} + 2\mu_{\text{B}}\right) + \beta\mu_{\text{O}} \\ & + \bar{p}\left[\mu_{\text{A}}^2 + \mu_{\text{A}}\mu_{\text{AB}} + \mu_{\text{AB}}^2 + \mu_{\text{AB}}\mu_{\text{B}} + \mu_{\text{B}}^2\right. \\ & \left. + 2(\mu_{\text{A}} + \mu_{\text{AB}} + \mu_{\text{B}})\mu_{\text{O}} + \mu_{\text{O}}^2\right] \end{aligned}$$

To calculate the price of fairness we now find the size of the fair matching. The only unmatched highly sensitized patients are in $V_H^{\text{O-AB}}$, some of which were matched in step 12 above. We now show that the number of matched vertices in $V^{\text{O-AB}}$ is smaller than the initial size of $V_H^{\text{O-AB}}$, so not all vertices in $V_H^{\text{O-AB}}$ can be matched. Let $M^{\text{O-AB}}$ be the number of matched vertices in $V^{\text{O-AB}}$, and $H^{\text{O-AB}}$ be the initial size of $V_H^{\text{O-AB}}$. The inequality $M^{\text{O-AB}} < H^{\text{O-AB}}$ can be written as

$$(\beta + \mu_{\text{AB}}\bar{p})(1 - \mu_{\text{AB}}) - \mu_{\text{AB}}(\mu_{\text{A}} + \mu_{\text{B}}) < (1 - \lambda)\mu_{\text{O}}\mu_{\text{AB}} \tag{B.1}$$

$$\beta < \mu_{\text{AB}}(1 - \bar{p}) - \lambda\mu_{\text{O}}\frac{\mu_{\text{AB}}}{1 - \mu_{\text{AB}}} \tag{B.2}$$

This inequality holds by assumption 5, and a.s. there are some unmatched vertices in $V_H^{\text{O-AB}}$. The number of unmatched highly sensitized vertices is

$$H^{\text{O-AB}} - M^{\text{O-AB}} \propto (1 - \mu_{\text{AB}})((1 - \bar{p})\mu_{\text{AB}} - \beta) - \lambda\mu_{\text{AB}}\mu_{\text{O}}$$

.

We match each of these remaining vertices by removing a 3-cycle of the form (AB-O, O-A, A-AB) and creating a 2-cycle (AB-O, O-AB). This matching used $|V^{\text{AB-O}}| \propto$

$\bar{p}\mu_{\text{O}}\mu_{\text{AB}}$ 3-cycles of this form, while $|V_H^{\text{O-AB}}| \propto (1-\lambda)\mu_{\text{O}}\mu_{\text{AB}}$. The model assumptions ensure that $|V^{\text{AB-O}}| > |V_H^{\text{O-AB}}|$, and all remaining vertices in $V_H^{\text{O-AB}}$ can be matched in this way.

To match each remaining vertex in $V_H^{\text{O-AB}}$, we remove from the matching one vertex from both $V^{\text{O-A}}$ and $V^{\text{A-AB}}$. Thus the total efficiency loss is $H^{\text{O-AB}} - M^{\text{O-AB}}$. The price of fairness is

$$\mathsf{POF}(\mathcal{M}, u_{LEX}) = \frac{(1-\mu_{\text{AB}})((1-\bar{p})\mu_{\text{AB}} - \beta) - \lambda\mu_{\text{AB}}\mu_{\text{O}}}{u_E}$$

With $u_E$ defined previously. □

**Proposition B.5.** *Assume*

***1*** $\beta > \mu_{AB}(1-\bar{p}) - \mu_{AB}\mu_O\bar{p}/(\mu_A + \mu_B)$

***2*** $\beta < \mu_{AB}(1-\bar{p}) - \lambda\mu_O\frac{\mu_{AB}}{1-\mu_{AB}}$

*These constraints imply* $\beta \in [0, 1/10]$. *Denote by* $\mathcal{M}$ *the set of matchings in* $G(n)$ *using cycles and chains up to length 3. Almost surely as* $n \to \infty$, *the price of fairness is*

$$\mathit{POF}(\mathcal{M}, u_{LEX}) = \frac{(1-\mu_{AB})((1-\bar{p})\mu_{AB} - \beta) - \lambda\mu_{AB}\mu_O}{u_E}$$

*with*

$$u_E = \mu_{AB}\mu_B + \mu_A(\mu_{AB} + 2\mu_B) + \beta\mu_O$$
$$+ \bar{p}\left[\mu_A^2 + \mu_A\mu_{AB} + \mu_{AB}^2 + \mu_{AB}\mu_B + \mu_B^2\right.$$

$$+ 2(\mu_A + \mu_{AB} + \mu_B)\mu_O + \mu_O^2]$$

*sketch.* We begin with matching $M^*$ as done in the proof of Lemma B.1, matching all highly sensitized vertices except for those in $V_H^{\text{AB-O}}$. We now complete the efficient matching using both 3-cycles and 3-chains as in [106].

7. A- and B-type NDDs donate to $V^{\text{A-AB}}$ and $V^{\text{B-AB}}$, respectively. Note that $|N^A| \propto \beta\mu_A$ and $|V^{\text{A-AB}}| \propto (1-\bar{p})\mu_A\mu_{\text{AB}}$. The inequality $\beta\mu_{\text{A}} < \mu_A\mu_{\text{AB}}(1-\bar{p})$ holds due to assumption **2**, and a.s. $|N^{\text{A}}| < |V^{\text{A-AB}}|$. By the same argument, a.s. $|N^{\text{B}}| < |V^{\text{B-AB}}|$. Thus, both $N^{\text{A}}$ and $N^{\text{B}}$ are exhausted, and $|V^{\text{A-AB}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{A}}$ and $|V^{\text{B-AB}}| \propto \mu_{\text{B}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{B}}$.

8. Create cycles of the form (AB-O, O-A, A-AB). The current size of each vertex group is

   (1) $|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$

   (2) $|V^{\text{O-A}}| \propto (1-\bar{p})\mu_{\text{A}}\mu_{\text{O}}$

   (3) $|V^{\text{A-AB}}| \propto \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{A}}$

   Note that the inequality $(3) < (1)$ can be written as

$$\beta > \mu_{\text{AB}}(1-\bar{p}) - \bar{p}\mu_{\text{AB}}\mu_{\text{O}}/\mu_{\text{A}}$$

   which holds by assumption **1** and a.s. $|V^{\text{A-AB}}| < |V^{\text{AB-O}}|$. The inequality $(3) < (2)$ can be written as

$$\beta > (1-\bar{p})(\mu_{\text{AB}} - \mu_{\text{O}})$$

which holds by the model assumptions, and a.s. $|V^{\text{A-AB}}| < |V^{\text{O-A}}|$. Executing these cycles exhausts $V^{\text{A-AB}}$ and leaves the following vertices remaining

$$|V^{\text{O-A}}| \propto (1-\bar{p})\mu_{\text{A}}(\mu_{\text{O}} - \mu_{\text{AB}}) + \mu_{\text{A}}\beta$$

$$|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{AB}}\mu_{\text{O}} - \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + \mu_{\text{A}}\beta$$

9. Create cycles of the form (AB-O, O-B, B-AB). The current size of each vertex group is

   (1) $|V^{\text{AB-O}}| \propto \bar{p}\mu_{\text{AB}}\mu_{\text{O}} - \mu_{\text{A}}\mu_{\text{AB}}(1-\bar{p}) + \mu_{\text{A}}\beta$

   (2) $|V^{\text{O-B}}| \propto (1-\bar{p})\mu_{\text{B}}\mu_{\text{O}}$

   (3) $|V^{\text{B-AB}}| \propto \mu_{\text{B}}\mu_{\text{AB}}(1-\bar{p}) - \beta\mu_{\text{B}}$

   Inequality $(3) < (2)$ can be written as

   $$\beta > (1-\bar{p})(\mu_{\text{AB}} - \mu_{\text{O}})$$

   which holds by the model assumptions. Inequality $(3) < (1)$ can be written as

   $$\beta > \mu_{\text{AB}}(1-\bar{p}) - \mu_{\text{AB}}\mu_{\text{O}}\bar{p}/(\mu_{\text{A}} + \mu_{\text{B}})$$

   which holds by assumption **1**.

   Executing these cycles exhausts $V^{\text{B-AB}}$ and leaves the following vertices remaining

   $$|V^{\text{AB-O}}| \propto (\beta - (1-\bar{p})\mu_{\text{AB}})(\mu_{\text{A}} + \mu_{\text{B}}) + \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$$

   $$|V^{\text{B-AB}}| \propto \mu_{\text{B}}(\beta - (1-\bar{p})(\mu_{\text{AB}} - \mu_{\text{O}}))$$

10. Create chains of the form (O,O-A,A-AB). Previous steps exhausted $V^{\text{A-AB}}$ so none of these chains occur.

11. Create chains of the form (O,O-B,B-AB). Previous steps exhausted $V^{\text{B-AB}}$ so none of these chains occur.

12. O-type NDDs and $V^{\text{AB-O}}$ match with remaining under-demanded vertices, starting with $V^{\text{O-AB}}$. The remaining size of each vertex group is

    (1) $|N^{\text{O}}| \propto \beta\mu_{\text{O}}$

    (2) $|V^{\text{AB-O}}| \propto (\beta - (1-\bar{p})\mu_{\text{AB}})(\mu_{\text{A}} + \mu_{\text{B}}) + \bar{p}\mu_{\text{AB}}\mu_{\text{O}}$

    (3) $|V^{\text{O-AB}}| \propto \mu_{\text{O}}\mu_{\text{AB}}$

    Note that the inequality $(1) + (2) < (3)$ can be written as

    $$\beta < \mu_{\text{AB}}(1-\bar{p})$$

    which holds by assumption **2** Thus O-type NDDs are exhausted first, leaving some vertices remaining in $V^{\text{O-AB}}$, with

    $$|V^{\text{O-AB}}| \propto ((1-\bar{p})\mu_{\text{AB}} - \beta)(1-\mu_{\text{AB}})$$

In the efficient matching described above, the number of *matched* pairs in each under-demanded group is

$|V^{\text{O-A}}| \propto \mu_{\text{A}}(\mu_{\text{AB}} + \bar{p}(\mu_{\text{O}} - \mu_{\text{AB}}) - \beta)$

$|V^{\text{O-B}}| \propto \mu_{\text{B}}(\mu_{\text{AB}} + \bar{p}(\mu_{\text{O}} - \mu_{\text{AB}}) - \beta)$

$|V^{\text{A-AB}}| \propto \mu_{\text{A}}\mu_{\text{AB}}$

$|V^{\text{B-AB}}| \propto \mu_{\text{B}}\mu_{\text{AB}}$.

$|V^{\text{O-AB}}| \propto (\beta + \mu_{\text{AB}}\bar{p})(1-\mu_{\text{AB}}) - \mu_{\text{AB}}(\mu_{\text{A}} + \mu_{\text{B}})$

Combining these with the over-demanded and self-demanded vertices, the total size of the efficient matching is

$$\begin{aligned} u_E = {} & \mu_{\text{AB}}\mu_{\text{B}} + \mu_{\text{A}}\left(\mu_{\text{AB}} + 2\mu_{\text{B}}\right) + \beta\mu_{\text{O}} \\ & + \bar{p}\left[\mu_{\text{A}}^2 + \mu_{\text{A}}\mu_{\text{AB}} + \mu_{\text{AB}}^2 + \mu_{\text{AB}}\mu_{\text{B}} + \mu_{\text{B}}^2\right. \\ & \left. + 2(\mu_{\text{A}} + \mu_{\text{AB}} + \mu_{\text{B}})\mu_{\text{O}} + \mu_{\text{O}}^2\right] \end{aligned}$$

To calculate the price of fairness we now find the size of the fair matching. The only unmatched highly sensitized patients are in $V_H^{\text{O-AB}}$, some of which were matched in step 12 above. We now show that the number of matched vertices in $V^{\text{O-AB}}$ is smaller than the initial size of $V_H^{\text{O-AB}}$, so not all vertices in $V_H^{\text{O-AB}}$ can be matched. Let $M^{\text{O-AB}}$ be the number of matched vertices in $V^{\text{O-AB}}$, and $H^{\text{O-AB}}$ be the initial size of $V_H^{\text{O-AB}}$. The inequality $M^{\text{O-AB}} < H^{\text{O-AB}}$ can be written as

$$(\beta + \mu_{\text{AB}}\bar{p})(1 - \mu_{\text{AB}}) - \mu_{\text{AB}}(\mu_{\text{A}} + \mu_{\text{B}}) < (1 - \lambda)\mu_{\text{O}}\mu_{\text{AB}} \tag{B.3}$$

$$\beta < \mu_{\text{AB}}(1 - \bar{p}) - \lambda\mu_{\text{O}}\frac{\mu_{\text{AB}}}{1 - \mu_{\text{AB}}} \tag{B.4}$$

This inequality holds by assumption 2, and a.s. there are some unmatched vertices in $V_H^{\text{O-AB}}$. The number of unmatched highly sensitized vertices is

$$H^{\text{O-AB}} - M^{\text{O-AB}} \propto (1 - \mu_{\text{AB}})((1 - \bar{p})\mu_{\text{AB}} - \beta) - \lambda\mu_{\text{AB}}\mu_{\text{O}}.$$

We match each of these remaining vertices by removing a 3-cycle of the form (AB-O, O-A, A-AB) and creating a 2-cycle (AB-O, O-AB). This matching used $|V^{\text{AB-O}}| \propto$

$\bar{p}\mu_{\mathrm{O}}\mu_{\mathrm{AB}}$ 3-cycles of this form, while $|V_H^{\text{O-AB}}| \propto (1-\lambda)\mu_{\mathrm{O}}\mu_{\mathrm{AB}}$. The model assumptions ensure that $|V^{\text{AB-O}}| > |V_H^{\text{O-AB}}|$, and all remaining vertices in $V_H^{\text{O-AB}}$ can be matched in this way.

To match each remaining vertex in $V_H^{\text{O-AB}}$, we remove from the matching one vertex from both $V^{\text{O-A}}$ and $V^{\text{A-AB}}$. Thus the total efficiency loss is $H^{\text{O-AB}} - M^{\text{O-AB}}$. The price of fairness is

$$\mathsf{POF}(\mathcal{M}, u_{LEX}) = \frac{(1-\mu_{\mathrm{AB}})((1-\bar{p})\mu_{\mathrm{AB}} - \beta) - \lambda\mu_{\mathrm{AB}}\mu_{\mathrm{O}}}{u_E}$$

With $u_E$ defined previously. □

Next we compare the price of fairness in Propositions B.2, B.3, B.4, and B.5 to the price of fairness in the efficient matching without NDDs, given in Dickerson et al. [108]:

$$\mathrm{POF}_0 = \frac{(1-\lambda)\mu_{\mathrm{O}}\mu_{\mathrm{AB}}}{u_E} \tag{B.5}$$

$$\begin{aligned} u_E = \bar{p}\Big[&2\mu_{\mathrm{AB}}\mu_{\mathrm{B}} + 2\mu_{\mathrm{AB}}\mu_{\mathrm{A}} + 3\mu_{\mathrm{AB}}\mu_{\mathrm{O}} \\ &+2\mu_{\mathrm{A}}\mu_{\mathrm{O}} + 2\mu_{\mathrm{B}}\mu_{\mathrm{O}} + \mu_{\mathrm{O}}^2 + \mu_{\mathrm{A}}^2 + \mu_{\mathrm{B}}^2 + \mu_{\mathrm{AB}}^2\Big] \\ &+2\mu_{\mathrm{A}}\mu_{\mathrm{B}} \end{aligned}$$

The following Lemmas state that $\mathrm{POF}_0$ is an upper bound on the price of fairness when NDDs are used, for each of the four cases when the price of fairness is nonzero.

**Lemma B.2.** *The price of fairness in Propositions B.2 and B.3 is bounded above by* $POF_0$.

*sketch.* The price of fairness in Propositions B.2 and B.3 is

$$\text{POF}_A = \frac{(1-\lambda)\mu_{\text{O}}\mu_{\text{AB}}}{u_E}$$

$$\begin{aligned} u_E = \bar{p}\Big[&2\mu_{\text{AB}}\mu_{\text{B}} + 2\mu_{\text{AB}}\mu_{\text{A}} + 3\mu_{\text{AB}}\mu_{\text{O}} \\ &+2\mu_{\text{A}}\mu_{\text{O}} + 2\mu_{\text{B}}\mu_{\text{O}} + \mu_{\text{O}}^2 + \mu_{\text{A}}^2 + \mu_{\text{B}}^2 + \mu_{\text{AB}}^2\Big] \\ &+2\mu_{\text{A}}\mu_{\text{B}} + \beta\left(\mu_{\text{A}} + \mu_{\text{B}} + 2\mu_{\text{O}}\right) \end{aligned}$$

Both $\text{POF}_0$ and $\text{POF}_A$ have the same numerator, and the denominator of $\text{POF}_A$ is equal to the denominator of $\text{POF}_0$, with the additional term $\beta\left(\mu_{\text{A}} + \mu_{\text{B}} + 2\mu_{\text{O}}\right)$. Thus when $\beta = 0$, $\text{POF}_0 = \text{POF}_A$, and when $\beta > 0$, $\text{POF}_0 > \text{POF}_A$, and the price of fairness in Propositions B.2 and B.3 is bounded above by $\text{POF}_0$. □

**Lemma B.3.** *The price of fairness in Propositions B.4 and B.5 is bounded above by $POF_0$.*

*sketch.* The price of fairness in Propositions B.4 and B.5 is

$$\text{POF}_B = \frac{(1-\mu_{\text{AB}})((1-\bar{p})\mu_{\text{AB}} - \beta) - \lambda\mu_{\text{AB}}\mu_{\text{O}}}{u_E}$$

$$\begin{aligned} u_E = &\mu_{\text{AB}}\mu_{\text{B}} + \mu_{\text{A}}\left(\mu_{\text{AB}} + 2\mu_{\text{B}}\right) + \beta\mu_{\text{O}} \\ &+ \bar{p}\big[\mu_{\text{A}}^2 + \mu_{\text{A}}\mu_{\text{AB}} + \mu_{\text{AB}}^2 + \mu_{\text{AB}}\mu_{\text{B}} + \mu_{\text{B}}^2 \\ &+ 2(\mu_{\text{A}} + \mu_{\text{AB}} + \mu_{\text{B}})\mu_{\text{O}} + \mu_{\text{O}}^2\big] \end{aligned}$$

To show that $\text{POF}_B < \text{POF}_0$ holds, we first show both (1) the numerator of $\text{POF}_B$ is smaller than that of $\text{POF}_0$, and (2) the denominator of $\text{POF}_B$ is larger than the denominator of $\text{POF}_0$.

**(1)** In both $\text{POF}_0$ and $\text{POF}_B$, the numerator is proportional to the number of remaining vertices in $V_H^{\text{O-AB}}$, after constructing the efficient matching. In Proposition B.4 and B.5 the efficient matching contains some vertices in $V_H^{\text{O-AB}}$; without NDDs, the efficient matching contains no vertices in $V_H^{\text{O-AB}}$. Thus, the numerator of $\text{POF}_B$ is strictly smaller the numerator of $\text{POF}_0$.

**(2)** Let the $D_0$ be the denominator of $\text{POF}_0$, and $D_B$ be the denominator of $\text{POF}_B$. We now show that the inequality $D_0 < D_B$ holds. First, note that this inequality can be written as

$$\mu_{\text{AB}} - (1-\bar{p})\mu_{\text{AB}}^2 + \beta\mu_{\text{O}} > \mu_{\text{AB}}(\bar{p} + \mu_{\text{O}}).$$

Rearranging, we have

$$\beta > (\mu_{\text{AB}}/\mu_{\text{O}})\big[(1-\bar{p})\mu_{\text{AB}} - (\mu_{\text{A}} + \mu_{\text{B}} + \mu_{\text{AB}} - \bar{p})\big]. \tag{B.6}$$

We now show that inequality B.6 is satisfied by the the following assumption on $\beta$, made in Propositions B.4 and B.5:

$$\mathbf{A} : \beta > \mu_{\text{AB}}(1-\bar{p}) - \mu_{\text{AB}}\mu_{\text{O}}\bar{p}/\mu_{\text{A}}.$$

Next, we show that assumption **A** implies inequality B.6, and thus assumption **A** implies $D_0 < D_B$. Assumption **A** implies B.6 if the right-hand side of **A** is larger than the right hand side of B.6, that is,

$$\begin{aligned}\mu_{\text{AB}}(1-\bar{p}) - \mu_{\text{AB}}\mu_{\text{O}}\bar{p}/\mu_{\text{A}} > (\mu_{\text{AB}}/\mu_{\text{O}})(1-\bar{p})\mu_{\text{AB}} \\ - (\mu_{\text{AB}}/\mu_{\text{O}})(\mu_{\text{A}} + \mu_{\text{B}} + \mu_{\text{AB}} - \bar{p})\end{aligned}$$

rearranging, we have

$$\frac{1-\bar{p}}{\bar{p}} > \frac{\mu_O}{\mu_A}\frac{1-\mu_{AB}-\mu_B}{1-\mu_B}$$

The random graph model assumes $\bar{p} \leq 2/5$, and $\mu_O \leq (3/2)\mu_A$, thus we have

$$\frac{1-\bar{p}}{\bar{p}} \geq \frac{3}{2} > \frac{3}{2}\frac{1-\mu_{AB}-\mu_B}{1-\mu_B} \geq \frac{\mu_O}{\mu_A}\frac{1-\mu_{AB}-\mu_B}{1-\mu_B}.$$

This shows that assumption **A** implies $D_0 < D_B$. Thus, the numerator of $\text{POF}_0$ is larger than the numerator of $\text{POF}_B$, and the denominator of $\text{POF}_0$ is smaller than the denominator of $\text{POF}_B$, and therefore $\text{POF}_B < \text{POF}_0$. □

Lemmas B.2 and B.3 show that with $\beta > 0$, the price of fairness has the same upper bound as when $\beta = 0$, given in Dickerson et al. [108]. That is, adding NDDs to the random graph model does not increase the price of fairness.

**Theorem 6.1.** *Adding NDDs to the random graph model ($\beta > 0$) does not increase the upper bound on the price of fairness found by Dickerson et al. [108].*

*Proof.* When $\beta > 0$, there are only four possible matchings with nonzero price of fairness, and the price of fairness for each case is given in Propositions B.2, B.3, B.4, and B.5. Lemmas B.2 and B.3 state that in each of these four cases, the matching with NDDs has a tighter bound on the price of fairness than the matching without NDDs, given in Dickerson et al. [108]. □

Next we show that the price of fairness is zero when $\beta > 1/8$, by finding the maximum possible $\beta$ for each of the four cases with nonzero price of fairness.

**Lemma B.4.** *In the matching described by Proposition B.2, $\beta < 1/8$.*

*Proof.* Proposition B.2 makes the following assumptions on $\beta$:

**1** $\beta < \mu_A (1 - \bar{p}) - \bar{p}\mu_{AB}$

**2** $\beta < \mu_{AB}(1 - \bar{p}) - \bar{p}\mu_{AB}\mu_O/\mu_A$

**3** $\beta < \mu_{AB}\left(\frac{\mu_A}{\mu_A+\mu_O} - \bar{p}\right)$

To determine an upper bound on $\beta$, we maximize the right hand side of constraint **3**. Note that the model assumes $\mu_{AB} < 1/4$, $\mu_A < 1/2$, and $\mu_A + \mu_O < 1$. Using these bounds, and $\bar{p} \to 0$, constraint **3** is bounded by

$$\beta < \mu_{AB}\left(\frac{\mu_A}{\mu_A + \mu_O} - \bar{p}\right) < (1/4)\frac{(1/2)}{1} = 1/8$$

$$\beta < 1/8$$

Constraints **1** and **2** are looser than constraint **3**: with the values $\bar{p} \to 0$, $\mu_A \to 1/4$, and $\mu_{AB} \to 1/4$, both constraints reduce to $\beta < 1/4$. □

**Lemma B.5.** *In the matching described by Proposition B.3, $\beta < 1/12$.*

*Proof.* Proposition B.3 makes the following assumptions

**1** $\beta < \mu_{AB}(1 - \bar{p}) - \mu_{AB}\mu_O\bar{p}/(\mu_A + \mu_B)$

**2** $\beta < \frac{\mu_A\mu_{AB}(1-\bar{p})+\mu_B\mu_O(1-\bar{p})-\bar{p}\mu_O\mu_{AB}}{\mu_A+\mu_O}$

**3** $\beta > \mu_{AB}(1 - \bar{p}) - \mu_{AB}\mu_O\bar{p}/\mu_A$

**4** $\beta < \mu_{AB}(1 - \bar{p}) - \bar{p}\mu_{AB}\mu_O/\mu_A + (1 - \bar{p})\mu_B\mu_O/\mu_A$

**5** $\beta < \mu_{AB}(1 - \bar{p}) - \mu_O\mu_{AB}/(1 - \mu_{AB})$

Combining **3** and **5**, we have

$$\mu_O\mu_{AB}/(1 - \mu_{AB}) < \mu_{AB}(1 - \bar{p}) - \beta < \mu_{AB}\mu_O\bar{p}/\mu_A$$

$$\mu_O\mu_{AB}/(1 - \mu_{AB}) < \mu_{AB}\mu_O\bar{p}/\mu_A$$

$$\mathbf{A} : \mu_{\mathrm{A}}/(1-\mu_{\mathrm{AB}}) < \bar{p}$$

Combining constraint **A** with **5** gives a new upper bound on $\beta$,

$$\begin{aligned}\beta &< \mu_{\mathrm{AB}}(1-\bar{p}) - \mu_{\mathrm{O}}\mu_{\mathrm{AB}}/(1-\mu_{\mathrm{AB}}) \\ &< \mu_{\mathrm{AB}}(1-\mu_{\mathrm{A}}/(1-\mu_{\mathrm{AB}})) - \mu_{\mathrm{O}}\mu_{\mathrm{AB}}/(1-\mu_{\mathrm{AB}})\end{aligned}$$

$$\beta < \mu_{\mathrm{AB}}\left(1 - \frac{\mu_{\mathrm{A}}+\mu_{\mathrm{O}}}{1-\mu_{\mathrm{AB}}}\right)$$

This bound is maximized when when $\mu_{\mathrm{AB}}$ is maximal, and $(\mu_{\mathrm{A}}+\mu_{\mathrm{O}})$ is minimal. In the random graph model, these values are $\mu_{\mathrm{AB}} \to 1/4$ and $(\mu_{\mathrm{A}}+\mu_{\mathrm{O}}) \to 1/2$, and the numerical bound is

$$\begin{aligned}\beta &< (1/4)\left(1 - \frac{(1/2)}{1-1/4}\right) = 1/12 \\ \beta &< 1/12\end{aligned}$$

□

**Lemma B.6.** *In the matching described by Proposition B.4, $\beta < 1/8$.*

*Proof.* Proposition B.4 makes the following assumptions on $\beta$

**1** $\beta > \mu_{\mathrm{AB}}(1-\bar{p}) - \mu_{\mathrm{AB}}\mu_{\mathrm{O}}\bar{p}/\mu_{\mathrm{A}}$

**2** $\beta < \mu_{\mathrm{AB}}(1-\bar{p}) - \mu_{\mathrm{AB}}\mu_{\mathrm{O}}\bar{p}/(\mu_{\mathrm{A}}+\mu_{\mathrm{B}})$

**3** $\beta < \mu_{\mathrm{AB}}(1-\bar{p}) - \bar{p}\mu_{\mathrm{AB}}\mu_{\mathrm{O}}/\mu_{\mathrm{A}} + (1-\bar{p})\mu_{\mathrm{B}}\mu_{\mathrm{O}}/\mu_{A}$

**4** $\beta > \mu_{\mathrm{AB}}\left((1-\bar{p}) - \frac{\mu_{\mathrm{O}}}{1-\mu_{\mathrm{AB}}}\right)$

**5** $\beta < \mu_{AB}(1-\bar{p}) - \lambda\mu_O\frac{\mu_{AB}}{1-\mu_{AB}}$

Combining **1** and **5** results in the following constraint, which is consistent with the above assumptions:

$$\mathbf{A} : \lambda\frac{\mu_A}{1-\mu_{AB}} < \bar{p}$$

Note that **5** is maximized when $\lambda$ is minimized; this occurs when $\lambda + \bar{p} \to 1$, and $\lambda \to 1 - \bar{p}$. In this case, **5** can be relaxed as

$$\begin{aligned}\beta &< \mu_{AB}(1-\bar{p}) - \lambda\mu_O\frac{\mu_{AB}}{1-\mu_{AB}} \\ &< \mu_{AB}(1-\bar{p}) - (1-\bar{p})\mu_{AB}\frac{\mu_O}{1-\mu_{AB}}\end{aligned}$$

$$\begin{aligned}\beta &< \mu_{AB}(1-\bar{p}) - (1-\bar{p})\mu_{AB}\frac{\mu_O}{1-\mu_{AB}} \\ &= (1-\bar{p})\frac{\mu_{AB}(\mu_A+\mu_B)}{1-\mu_{AB}}\end{aligned}$$

Finally, we have

$$\beta < (1-\bar{p})\frac{\mu_{AB}(\mu_A+\mu_B)}{1-\mu_{AB}}$$

The right hand side of this constraint is maximal when $\bar{p}$ is minimal; constraint **A** determines the lower bound for $\bar{p}$, with $\lambda \to 1 - \bar{p}$:

$$\begin{aligned}(1-\bar{p})\frac{\mu_A}{1-\mu_{AB}} &< \bar{p} \\ \frac{\mu_A}{1-\mu_{AB}} &< \bar{p}\left(1+\frac{\mu_A}{1-\mu_{AB}}\right)\end{aligned}$$

$$\frac{\mu_A}{1-\mu_{AB}+\mu_A} < \bar{p}$$

Using this lower bound on $\bar{p}$, we can further relax 5

$$\begin{aligned}
\beta &< (1-\bar{p})\frac{\mu_{AB}(\mu_A+\mu_B)}{1-\mu_{AB}} \\
&< (1-\frac{\mu_A}{1-\mu_{AB}+\mu_A})\frac{\mu_{AB}(\mu_A+\mu_B)}{1-\mu_{AB}} \\
&= \frac{1-\mu_{AB}}{1-\mu_{AB}+\mu_A}\frac{\mu_{AB}(\mu_A+\mu_B)}{1-\mu_{AB}} \\
&= \frac{\mu_{AB}(\mu_A+\mu_B)}{1-\mu_{AB}+\mu_A}
\end{aligned}$$

$$\beta < \frac{\mu_{AB}(\mu_A+\mu_B)}{1-\mu_{AB}+\mu_A}$$

The right hand side is maximal when $\mu_{AB}$ is maximal, and $\mu_{AB}, \mu_A, \mu_B, \mu_O \to 1/4$. This gives the final bound on $\beta$,

$$\beta < \frac{(1/4)(1/2)}{1} = 1/8$$

$$\beta < 1/8$$

□

**Lemma B.7.** *In the matching described by Proposition B.5, $\beta < 1/10$.*

*Proof.* Proposition B.5 makes the following assumptions on $\beta$

**1** $\beta > \mu_{AB}(1-\bar{p}) - \mu_{AB}\mu_O\bar{p}/(\mu_A+\mu_B)$

**2** $\beta < \mu_{AB}(1-\bar{p}) - \lambda\mu_O\frac{\mu_{AB}}{1-\mu_{AB}}$

Combining these assumptions results in the following constraint:

$$\mathbf{A} : \lambda \frac{\mu_A + \mu_B}{1 - \mu_{AB}} < \bar{p}$$

Note that assumption **2** is identical to assumption **5** in Lemma B.6. Following the same procedure used in the proof of Lemma B.6, **2** can be relaxed as

$$\beta < (1 - \bar{p}) \frac{\mu_{AB}(\mu_A + \mu_B)}{1 - \mu_{AB}}$$

The right hand side of this constraint is maximal when $\bar{p}$ is minimal; constraint **A** determines the lower bound for $\bar{p}$, with $\lambda \to 1 - \bar{p}$:

$$\begin{aligned}
(1 - \bar{p}) \frac{\mu_A + \mu_B}{1 - \mu_{AB}} &< \bar{p} \\
\frac{\mu_A + \mu_B}{1 - \mu_{AB}} &< \bar{p} \left( 1 + \frac{\mu_A + \mu_B}{1 - \mu_{AB}} \right) \\
\frac{\mu_A + \mu_B}{2\mu_A + 2\mu_B + \mu_O} &< \bar{p}
\end{aligned}$$

Using this lower bound on $\bar{p}$, we can further relax **2**

$$\begin{aligned}
\beta &< (1 - \bar{p}) \frac{\mu_{AB}(\mu_A + \mu_B)}{1 - \mu_{AB}} \\
&< (1 - \frac{\mu_A + \mu_B}{2\mu_A + 2\mu_B + \mu_O}) \frac{\mu_{AB}(\mu_A + \mu_B)}{1 - \mu_{AB}} \\
&= \frac{1 - \mu_{AB}}{2\mu_A + 2\mu_B + \mu_O} \frac{\mu_{AB}(\mu_A + \mu_B)}{1 - \mu_{AB}} \\
&= \frac{\mu_{AB}(\mu_A + \mu_B)}{2\mu_A + 2\mu_B + \mu_O}
\end{aligned}$$

$$\beta < \frac{\mu_{AB}(\mu_A + \mu_B)}{2\mu_A + 2\mu_B + \mu_O}$$

The right hand side is maximal when $\mu_{AB}$ is maximal, and $\mu_{AB}, \mu_A, \mu_B, \mu_O \to 1/4$. This gives the final bound on $\beta$,

$$\beta < \frac{(1/4)(1/2)}{5/4} = 1/10$$

$$\beta < 1/10$$

□

Combining Lemmas B.4, B.5, B.6, and B.7, we find that the price of fairness is zero when $\beta > 1/8$.

**Theorem 6.2.** *The price of fairness is zero when $\beta > 1/8$.*

*Proof.* There are only four matchings with nonzero price of fairness and $\beta > 0$, which are described in Propositions B.2, B.3, B.4, and B.5. Lemmas B.4, B.5, B.6, and B.7 state that the maximum $\beta$ for any of these matchings is $1/8$. When $\beta > 1/8$, the matching is not one of these four cases, and the price of fairness is zero. □

Theorems 6.1 and 6.2 are the two main theoretical results of Chapter 6: adding NDDs to the random graph model does not increase the upper bound on the price of fairness, and when the proportion of NDDs is high enough ($\beta > 1/8$), the price of fairness is zero. We show this by addressing each of the four efficient matchings on the random graph model with nonzero price of fairness. In each case, and $\beta < 1/8$, and the matching with NDDs has a smaller price of fairness than the matching without NDDs given in Dickerson et al. [108].

To further explore these results, we numerically find the maximum price of fairness for the matchings given in Propositions B.2, B.3, B.4, and B.5. For each matching, we find the maximum price of fairness for a range of $\beta$, within the defined

constraints, using the "NMaximize" function in Mathematica with the nonlinear interior point method.

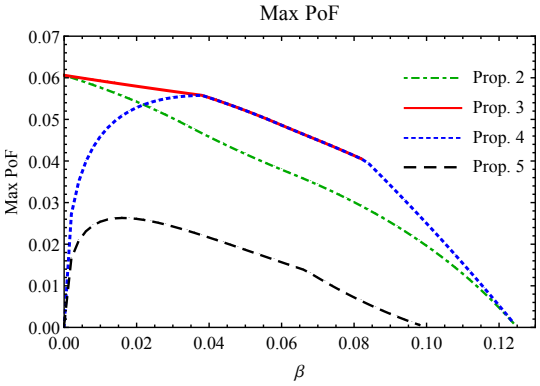



FIGURE B.2: Maixmum price of fairness for each of the four matchings addressed in Propositions B.2, B.3, B.4, and B.5.

Figure B.2 confirms both of our main theoretical results: adding NDDs to the efficient matching decreases the upper bound on the price of fairness, and when $\beta > 1/8$ the price of fairness is zero.

## B.2 Additional Experimental Results

This section contains worst-case price of fairness (PoF) and worst-case fairness (%$F$) for real UNOS graphs, and for simulated graphs; these results were produced using the method described in Section 6.5.

### B.2.1 UNOS Graphs

Figure B.3 shows the worst-case (maximum) PoF of each fair algorithm on the 314 UNOS graphs; Figure B.4 shows worst-case (minimum) %$F$.

Real exchange graphs are relatively sparse, and have very few feasible matchings. Each fair algorithm effectively chooses one of these matchings, based on a fairness criteria. Especially with sparse graphs, fairness is often achieved by using

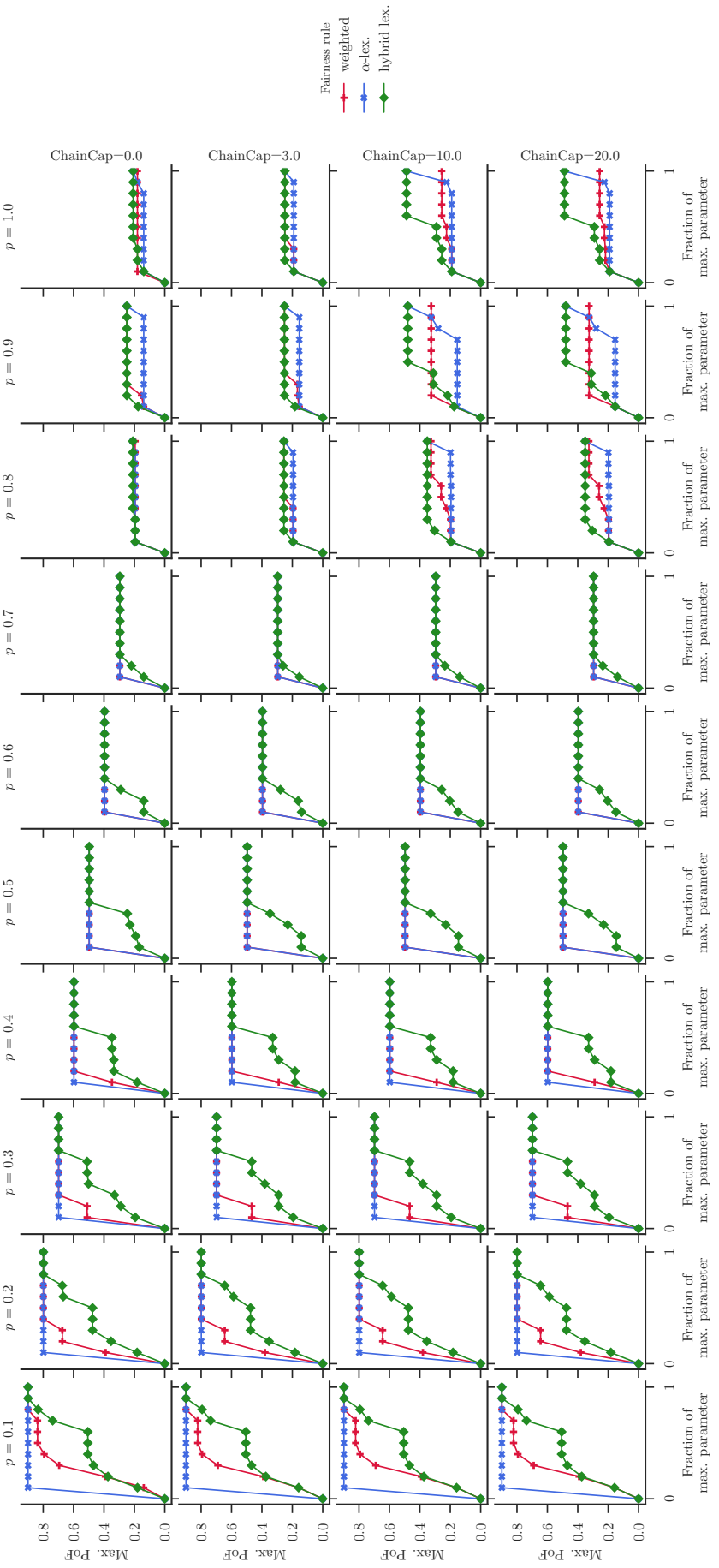



FIGURE B.3: Maximum PoF for each fair algorithm. Parameters for each algorithm are $\alpha \in [0, 1]$, $\beta \in [0, 20]$, and $\Delta \in [0, u(M_E)]$. Rows correspond to edge success probabilities from 0.1 to 1.0; columns correspond to different chain caps: 0, 3, and 20.

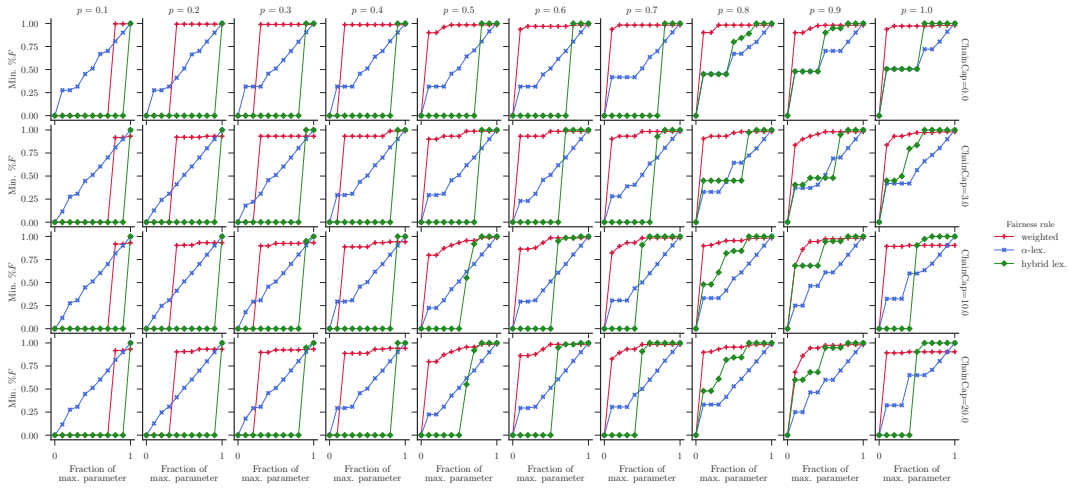



FIGURE B.4: Minimum fraction of the fair score for each fair algorithm. Parameters for each method are $\alpha \in [0,1]$, $\beta \in [0,20]$, and $\Delta \in [0, u(M_E)]$. Rows correspond to edge success probabilities from 0.1 to 1.0; columns correspond to different chain caps: 0, 3, and 20.

longer cycles or cycles to match highly sensitized vertices. When edge success probability $p$ is high, fairness has little effect on overall utility, and PoF is often below 0.3. With lower edge success probability, using longer cycles and chains causes a huge loss in efficiency: the expected utility of $n$-cycles and chains is proportional to $p^n$, which incurs a huge penalty for long cycles and chains when $p$ is small. Thus as $p$ decreases, very small $\alpha$ and $\beta$ values result in a high PoF. Our results show that for $p \leq 0.8$, even the smallest parameters for $\alpha$-lexicographic and weighted fairness ($\alpha = 0.1$ and $\beta = 2$) achieve the worst-case PoF. As expected, hybrid-lexicographic fairness limits PoF according to Theorem 6.9. With two classes of patients (highly- and lowly-sensitized), the theoretical price of fairness is bounded by $\mathsf{POF}(\mathcal{M}, u_\Delta) \leq 2\Delta / u(M_E)$; in the Figures, $\Delta$ is scaled by $u(M_E)$, so the upper bound on the price of fairness has a slope of two.

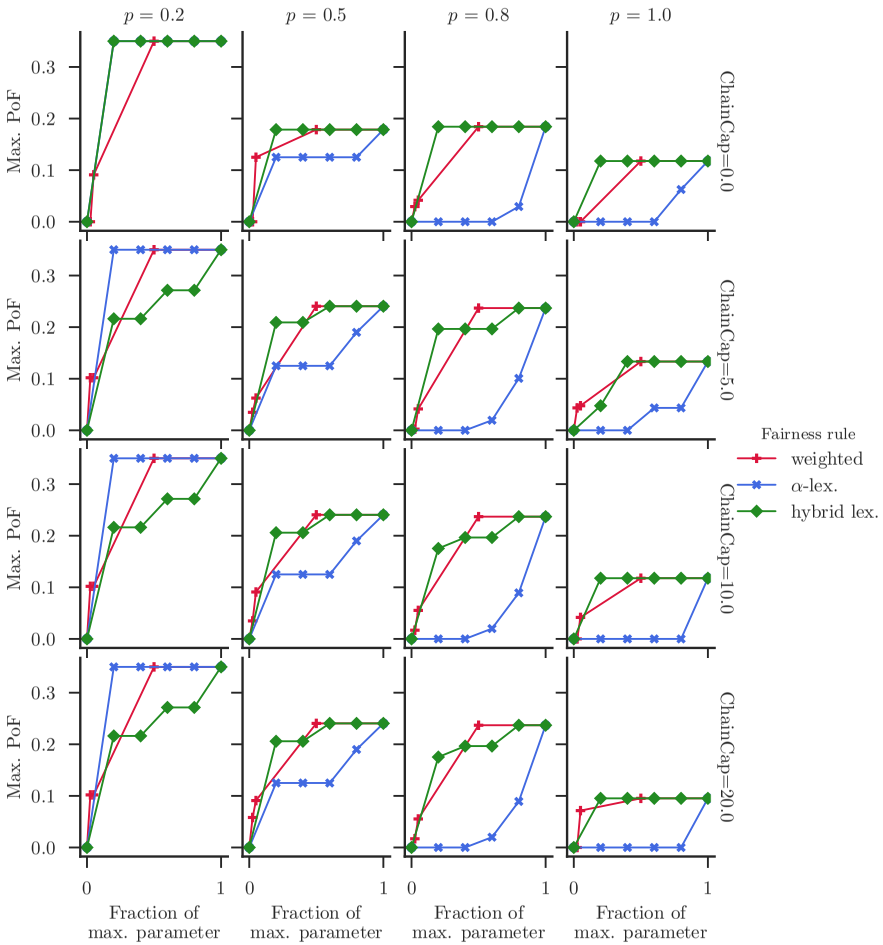
To illustrate the other side of the fairness-efficiency trade-off, we consider worst case $\%F$. Figure B.4 shows the minimum (worst case) $\%F$ over all UNOS graphs for each fair algorithm, and for various edge success probabilities and chain caps.

As expected, $\alpha$-lexicographic fairness guarantees at $\%F \geq \alpha$; weighted and hybrid-lexicogrpahic fairness do not make this guarantee. Small edge success probabilities make it impossible to match highly sensitized patients without large efficiency loss; when $p$ becomes small hybrid-lexicographic fairness matches no highly sensitized patients in the worst case.

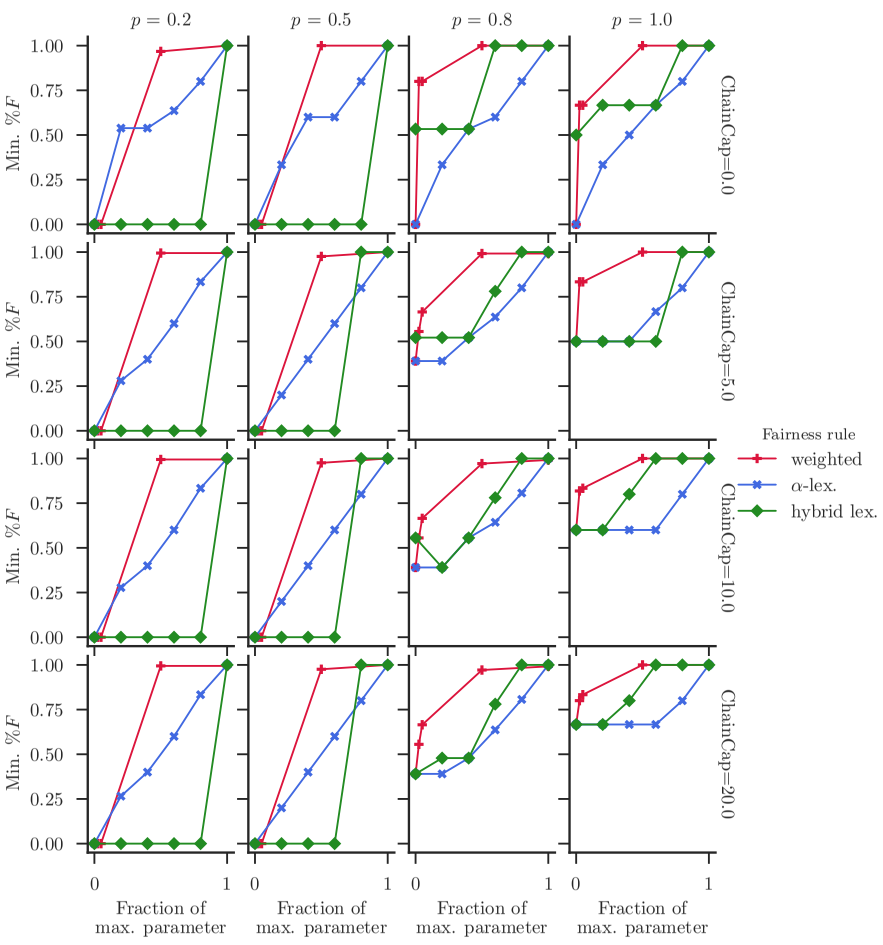
These results demonstrate the balance between fairness and efficiency offered by both $\alpha$-lexicographic and hybrid-lexicographic fairness. If fairness is more important than efficiency, then the $\alpha$-lexicographic algorithm can be used to guarantee that the resulting matching achieves at least fraction $\alpha$ of the maximum possible fair score. Alternatively, if efficiency is more important than fairness, hybrid-lexicographic fairness can bound the price of fairness using parameter $\Delta$.

### B.2.2 Simulated Exchange Graphs

Simulated exchange graphs were drawn from previous UNOS exchanges, using the same method as Dickerson et al. [107]. These graphs are typically denser than real graphs, and have a much lower price of fairness. Figures B.5 and B.6 show the worst-case PoF and $\%F$ on 32 simulated exchanges of size 64 and 128.

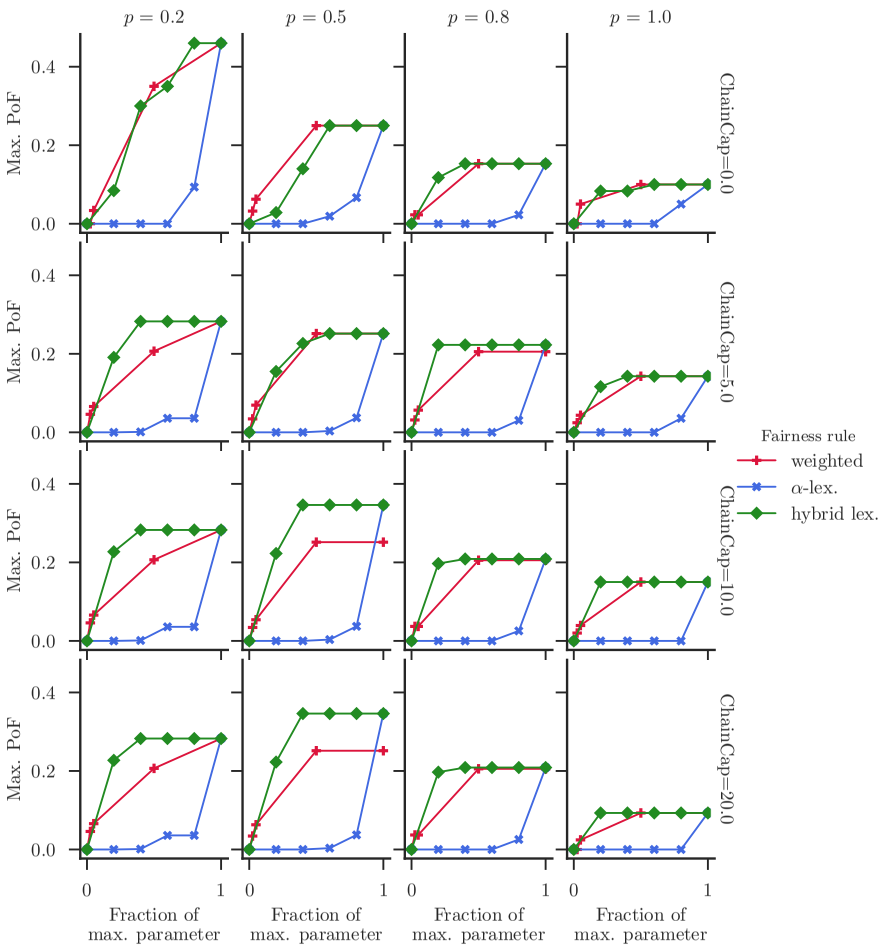



(A) Maximum PoF.

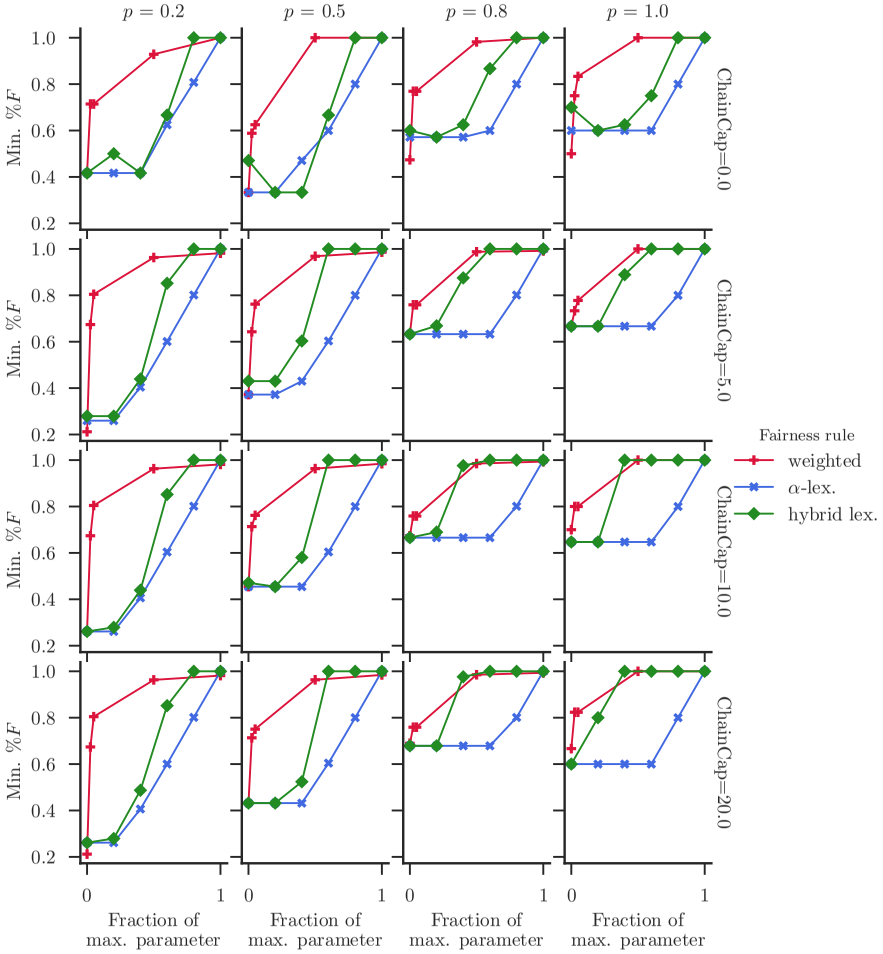



(B) Minimum %$F$.

FIGURE B.5: Worst-case PoF and %$F$ for 32 64-vertex random graphs. Parameters for each method are $\alpha \in [0,1]$, $\beta \in [0,20]$, and $\Delta \in [0, u(M_E)]$. Rows correspond to edge success probabilities from 0.1 to 1.0; columns correspond to different chain caps: 0, 3, and 20.




(A) Maximum PoF.




(B) Minimum %$F$.

FIGURE B.6: Worst-case PoF and %$F$ for 32 128-vertex random graphs. Parameters for each method are $\alpha \in [0,1]$, $\beta \in [0,20]$, and $\Delta \in [0, u(M_E)]$. Rows correspond to edge success probabilities from 0.1 to 1.0; columns correspond to different chain caps: 0, 3, and 20.

# Appendix C: Appendix to Chapter 8

## C.1 Estimating The Objective of Problem 8.1

The objective of the single-stage edge selection problem requires evaluating all rejection scenarios $\boldsymbol{r} \sim \mathbb{P}_R(\boldsymbol{q})$, and the support of this distribution grows exponentially in the number of edges $|\boldsymbol{q}|$. In computational experiments, to estimate the objective of Problem 8.1, we sample up to 1000 scenarios from $\mathbb{P}_R(\boldsymbol{q})$. More explicitly: we *exactly* evaluate the objective of edge sets with fewer than 10 edges; for larger edge sets, we sample the objective using 1000 draws from $\mathbb{P}_R(\boldsymbol{q})$.

Using bootstrapping experiments we demonstrate that our sampling approach is sufficient to accurately estimate the true objective, even for large edge sets. For 152 UNOS graphs, we computed edge sets by running `Greedy` with edge budgets ranging from 1 to 100. For each edge set, we then sample a subset of $N \in \{10, 30, 50, 100, 1000\}$ rejection scenarios, with replacement, from the set of all sampled edge outcomes. For each edge set and choice of $N$ we repeat 200 times and calculate the sample mean for each replication. We then compute the standard deviations of these bootstrap sample means to estimate the variance due to sampling. For each $N$, we calculate the mean sample standard deviation, normalized by the sample mean. Table C.1 shows the median normalized standard deviation for all experiments under each $N$, with edge budgets aggregated into 10 bins. We find that with $N = 1000$ samples, the standard deviation was on average only about 2% of the overall mean value, even

| Edge budgets | $N = 10$ | $N = 30$ | $N = 50$ | $N = 100$ | $N = 1000$ |
|---|---|---|---|---|---|
| 1-10 | 0.10 | 0.06 | 0.04 | 0.03 | 0.01 |
| 11-20 | 0.12 | 0.07 | 0.05 | 0.04 | 0.01 |
| 21-30 | 0.13 | 0.08 | 0.06 | 0.04 | 0.01 |
| 31-40 | 0.14 | 0.08 | 0.06 | 0.04 | 0.01 |
| 41-50 | 0.14 | 0.08 | 0.06 | 0.04 | 0.01 |
| 51-60 | 0.15 | 0.08 | 0.07 | 0.05 | 0.01 |
| 61-70 | 0.15 | 0.09 | 0.07 | 0.05 | 0.02 |
| 71-80 | 0.16 | 0.09 | 0.07 | 0.05 | 0.02 |
| 81-90 | 0.17 | 0.10 | 0.08 | 0.05 | 0.02 |
| 91-100 | 0.18 | 0.10 | 0.08 | 0.06 | 0.02 |

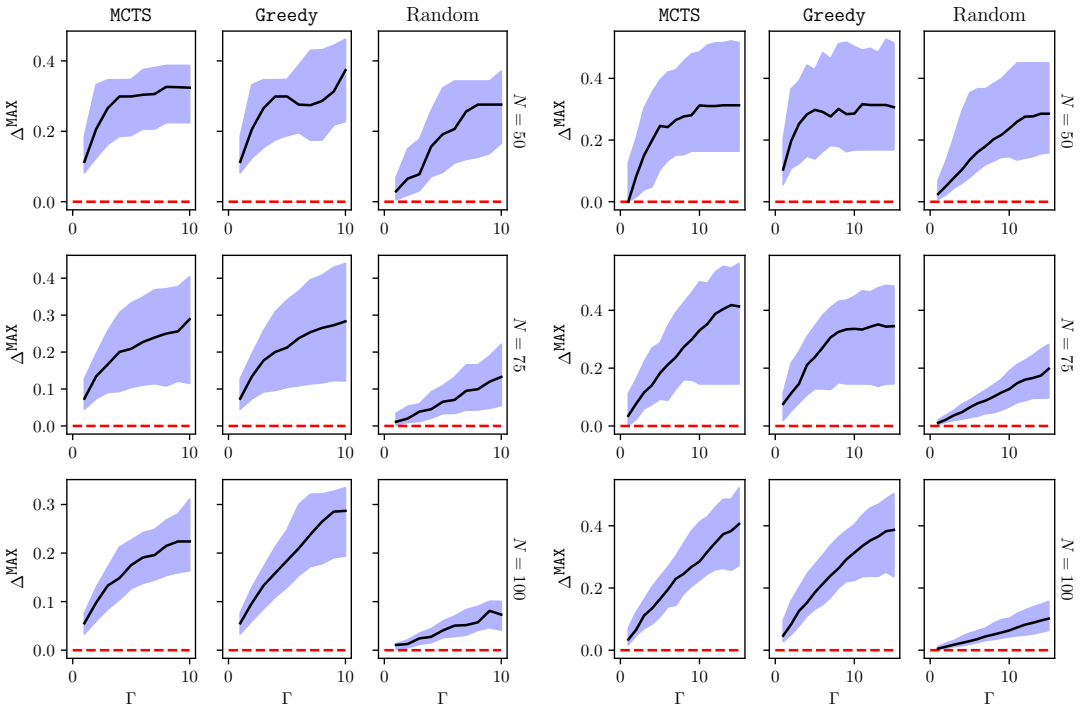
TABLE C.1: Median normalized standard deviation of the bootstrap mean, over 200 bootstrap samples for each sample size $N$, binned by edge budget.

for large edge budgets.

## C.2 Additional Computational Results

First we show results for both single-stage and multi-stage edge selection on random graphs (see § 8.4 for a description of these graphs). For $N = 50$, 75, and 100, we generate 30 random graphs with $N$ vertices and $p = 0.01$. For each graph we run single-stage experiments with $\Gamma = 1, \ldots, 10$ and multi-stage experiments with $\Gamma = 1, \ldots, 15$. Unlike experiments on UNOS graphs we use a time limit of 20 minutes per edge; all other parameters are the same. Figure C.1a and C.1b show single-stage and multi-stage results for all random graphs, respectively. Table C.2 shows comparisons to `IIAB` and Fail-Aware for random graphs with $N = 50$, 75, and 100.

As with UNOS graphs, results for `MCTS` and `Greedy` are quite similar, and both methods achieve larger $\Delta^{\texttt{MAX}}$ than Random, `IIAB`, and Fail-Aware. We make two observations: (1) `Greedy` appears to achieve larger $\Delta^{\texttt{MAX}}$ than `MCTS` in the single-stage setting, likely because of insufficient training time for `MCTS`; (2) in the multi-stage setting, `MCTS` performs *at least* as well as `Greedy`, and often better. Observation (2) is consistent with our experiments on UNOS graphs, and is somewhat surprising




(A) Single-stage. (B) Multi-stage.

FIGURE C.1: Results for 30 random graphs with edge probability $p = 0.01$ and $N = 50$ vertices (top row), $N = 75$ (middle row), and $N = 100$ (bottom row). All experiments use the *Simple* edge distribution. In all plots, a solid line indicates median $\Delta^{\texttt{MAX}}$ over all 30 random graphs, and shading is between the $10^{th}$ and $90^{th}$ percentiles; a dotted line indicates the baseline.

TABLE C.2: Single-stage results on random graphs with the *Simple* edge distribution, using the variable IIAB edge budget (top rows), and the failure-aware method (bottom row). Columns $P_X$ indicates the $X^{th}$ percentile of $\Delta^{\texttt{MAX}}$ over all 30 random graphs, for graphs with $N = 50$, 75, and 100 vertices.

| | $N = 50$ | | | $N = 75$ | | | $N = 100$ | | |
|---|---|---|---|---|---|---|---|---|---|
| **Method** | $P_{10}$ | $P_{50}$ | $P_{90}$ | $P_{10}$ | $P_{50}$ | $P_{90}$ | $P_{10}$ | $P_{50}$ | $P_{90}$ |
| `MCTS` | 0.22 | 0.30 | 0.38 | 0.11 | 0.33 | 0.46 | 0.23 | 0.33 | 0.38 |
| `Greedy` | 0.21 | 0.30 | 0.38 | 0.12 | 0.32 | 0.48 | 0.27 | 0.39 | 0.43 |
| Random | 0.12 | 0.19 | 0.23 | 0.10 | 0.19 | 0.28 | 0.12 | 0.19 | 0.23 |
| `IIAB` | 0.07 | 0.24 | 0.34 | 0.11 | 0.22 | 0.41 | 0.07 | 0.24 | 0.34 |
| Fail-Aware | 0.00 | 0.02 | 0.10 | 0.00 | 0.06 | 0.18 | 0.00 | 0.02 | 0.10 |

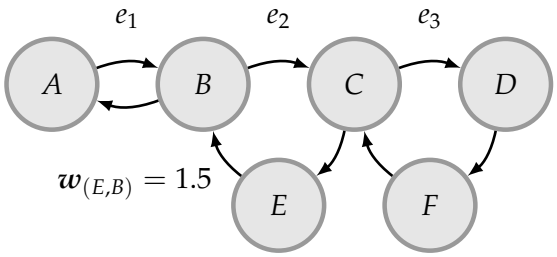



FIGURE C.2: Exchange graph for Propositions 8.1 and 8.2. All edges have weight 1 except for edge $(E, B)$, which has weight 1.5.

given that MCTS used less training time in these experiments. This suggests that MCTS may substantially improve over Greedy in the multi-stage setting; we leave further investigation to future work.

## C.3 Proofs for Section 8.2

In the proofs of Proposition 8.1 and Proposition 8.2 we consider a setting where all edges' pre-match rejections and post-match failures are i.i.d., where $P_R = 0.5$ is the pre-match rejection probability, $P_Q = 1.0$ is the post-match success probability if the edge is queried-and-accepted, and $P_N = 0.5$ is the success probability if $e$ is not queried. That is, queried edges have rejection probability 0.5, accepted edges have zero failure probability, and non-queried edges have failure probability 0.5.

### C.3.1 Proof of Proposition 8.1

(Proof by counterexample.) We provide an example where querying a single edge results in a *lower* objective value in Problem 8.1 (i.e., final expected matching weight) than querying no edges–when using the max-weight matching policy $M^{\texttt{MAX}}(\cdot)$.

Consider the exchange graph in Figure C.2; edge $(E, B)$ has weight 1.5, while all other edges have weight 1. First we consider the objective due to querying no edges, $V^S(\mathbf{0})$. In this case, no edges can be rejected pre-match, the max-weight matching

includes cycle $(C, D, F)$ (expected weight $3 \times (1/2)^3 = 3/8$) and cycle $(A, B)$ (expected weight $2 \times (1/2)^2 = 1/2$), with total expected matching weight $7/8$. That is, $V^S(\mathbf{0}) = 7/8$.

Next consider the objective due to querying only edge $e_3 = (C, D)$, and let $\boldsymbol{q}'$ denote edge set $\{e_3\}$. With probability $1/2$, $e_3$ is rejected and cycle $(B, C, E)$ is the max-weight matching – with expected weight $3.5/8$. With probability $1/2$, $e_3$ is accepted and the max-weight matching includes cycles $(A, B)$ (with expected weight $1/2$) and $(C, D, F)$ (with expected weight $3/4$); this matching has total expected weight $5/4$. Thus, $V^S(\boldsymbol{q}) = 27/32 < 7/8 = V^S(\mathbf{0})$, which concludes the proof.

### C.3.2 Proof of Proposition 8.2

(Proof by counterexample.) We provide an example where the objective value in Problem 8.1 (i.e., final expected matching weight) is non-submodular–when using the max-weight matching policy $M^{\texttt{MAX}}(\cdot)$. We use the same rejection and failure distribution as in the proof of Proposition 8.1.

Consider the exchange graph in Figure C.2; edge $(E, B)$ has weight 1.5, while all other edges have weight 1. With some abuse of notation, we will denote by $V^S(\{e_a, \ldots, e_N\})$ the objective of Problem 8.1 due to edge set $\{e_a, \ldots, e_N\}$. Our counterexample for submodulartiy is that, for this graph,

$$V^S(X \cup \{e_1, e_2\}) + V^S(X) > V^S(X \cup \{e_1\}) + V^S(X \cup \{e_2\}),$$

with set $X \equiv \{e_3\}$. That is, the objective increase due to of querying *both* edges $e_1$ and $e_3$ is greater than the combined increase due to querying both edges separately. Next we explicitly calculate each of the above terms.

$V^S(X) = V^S(\{e_3\})$**.** There are two cases to consider:

- $e_3$ is accepted, with probability $1/2$. The max-weight matching is cycles $(A, B)$ and $(C, D, F)$, with expected weight $(1/2 + 3/4)$,
- $e_3$ is rejected, with probability $1/2$. The max-weight matching is cycle $(B, C, E)$, with expected weight $3.5/8$.

Thus, $V^S(X) = (1/2)(1/2 + 3/4) + (1/2)(3.5/8) = 27/32$.

$V^S(X \cup \{e_1\}) = V^S(\{e_1, e_3\})$. There are four cases to consider:

- $e_1$ and $e_3$ are accepted, with probability $1/4$. The max-weight matching is cycles $(A, B)$ and $(C, D, F)$, with expected weight $(1 + 3/8)$,
- $e_1$ is rejected and $e_3$ is accepted, with probability $1/4$. The max-weight matching is cycle $(B, C, E)$, with expected weight $3.5/8$.
- $e_1$ is accepted and $e_3$ is rejected, with probability $1/4$. The max-weight matching is cycle $(B, C, E)$, with expected weight $3.5/8$.
- $e_1$ and $e_3$ are rejected, with probability $1/4$. The max-weight matching is cycle $(B, C, E)$, with expected weight $3.5/8$.

Thus the objective is $V^S(X \cup \{e_3\}) = (1/4)(1 + 3/8) + (3/4)(3.5/8) = 43/64$.

$V^S(X \cup \{e_2\}) = V^S(\{e_2, e_3\})$. There are three cases to consider

- $e_3$ is accepted, with probability $1/2$. The max-weight matching is cycles $(A, B)$ and $(C, D, F)$, with expected weight $(1/2 + 3/4)$,
- $e_3$ is rejected and $e_3$ is accepted, with probability $1/4$. The max-weight matching is cycle $(B, C, E)$, with expected weight $3.5/4$,
- $e_3$ and $e_2$ are rejected, with probability $1/4$. The max-weight matching is cycle $(A, B)$, with expected weight $1/2$.

Thus the objective is $V^S(X \cup \{e_2\}) = (1/2)(1/2+3/4) + (1/4)(3.5/4) + (1/4)(1/2) = 31/32$.

$V^S(X \cup \{e_1, e_2\}) = V^S(\{e_1, e_2, e_3\})$. There are four cases to consider:

- $e_1$ and $e_3$ are accepted, with probability $1/4$. The max-weight matching is cycles $(A, B)$ and $(C, D, F)$, with expected weight $(1 + 3/4)$,
- $e_1$ is accepted and $e_2$ is rejected, with probability $1/4$ (the response from $e_3$ is irrelevant). The max-weight matching is $(A, B)$ and $(C, D, F)$, with expected weight $1 + 3/8$.
- $e_1$ is rejected and $e_2$ is accepted (the response from $e_3$ is irrelevant), with probability $1/4$. The max-weight matching is cycle $(B, C, E)$, with expected weight $3.5/4$.
- $e_1$ and $e_2$ are rejected (the response from $e_3$ is irrelevant), with probability $1/4$. The max-weight matching is cycle $(C, D, F)$, with expected weight $3/8$.

Thus the objective is $V^S(X \cup \{e_1, e_2\}) = (1/4)(1+3/4) + (1/4)(1+3/8) + (1/4)(3.5/4) + (1/4)(3/8) = 35/32$.

Finally, we have:

$$V^S(X \cup \{e_1, e_2\}) + V^S(X) = 35/32 + 27/32$$
$$= 1.9375$$

and

$$V^S(X \cup \{e_1\}) + V^S(X \cup \{e_2\}) = 43/64 + 31/32$$
$$= 1.640625$$

Therefore, $V^S(X \cup \{e_1, e_2\}) + V^S(X) > V^S(X \cup \{e_1\}) + V^S(X \cup \{e_2\})$, which concludes the proof.

### C.3.3 Proof of Proposition 8.3

For the proof of Proposition 8.3 we make one assumption about the distribution of edge rejections and failures: querying *additional* edges cannot increase the overall probability of rejection or failure for any edge.

First we prove a handful of useful results.

**Lemma C.1.** *If all edges are independent and Assumption 8.1 holds, then additional edge queries cannot decrease expected post-match cycle and chain weights. Formally,*

$$\mathbb{E}\left[F(c, \boldsymbol{r} + \boldsymbol{f}) \mid \boldsymbol{q}, \boldsymbol{r}\right] \leq \mathbb{E}\left[F(c, \boldsymbol{r} + \boldsymbol{r}' + \boldsymbol{f}) \mid \boldsymbol{q} + \boldsymbol{q}', \boldsymbol{r}\right]$$

*for any* $\boldsymbol{q}, \boldsymbol{q}' \in \{0,1\}^{|E|}$ *such that* $\boldsymbol{q} + \boldsymbol{q}' \in \{0,1\}^{|E|}$, *for any* $\boldsymbol{r} \in \{0,1\}^{|E|}$, *and for all* $c \in \mathcal{C}$.

*Proof.* We address cycles and chains separately.

**Cycles.** Conditional on fixed $\boldsymbol{q}$ and $\boldsymbol{r}$, the expected weight of cycle $c = (e_1, \ldots, e_L)$ is expressed as

$$\begin{aligned}\mathbb{E}\left[F(c, \boldsymbol{r} + \boldsymbol{f}) \mid \boldsymbol{q}, \boldsymbol{r}\right] &= \left(\sum_{e \in c} w_e\right) \mathbb{E}\left[\prod_{e \in c} (1 - \boldsymbol{r}_e - \boldsymbol{f}_e) \mid \boldsymbol{q}, \boldsymbol{r}\right] \\ &= \left(\sum_{e \in c} w_e\right) \prod_{e \in c} \left(1 - \mathbb{E}\left[\boldsymbol{r}_e + \boldsymbol{f}_e \mid \boldsymbol{q}, \boldsymbol{r}\right]\right)\end{aligned}$$

where the second step is due to the fact that all $f_e$ are independent. Similarly, for fixed $q'$,

$$\mathbb{E}\left[F(c, r + r' + f) \mid q + q', r\right] = \left(\sum_{e \in c} w_e\right) \prod_{e \in c} \left(1 - \mathbb{E}\left[r_e + r'_e + f_e \mid q + q', r\right]\right) .$$

Due to Assumption 8.1, the following inequality holds for all edges $e \in E$

$$\mathbb{E}\left[r_e + f_e \mid q, r\right] \geq \mathbb{E}\left[r_e + r'_e + f_e \mid q + q', r\right] ,$$

and it follows that

$$\mathbb{E}\left[F(c, r + f) \mid q, r\right] \leq \mathbb{E}\left[F(c, r + r' + f) \mid q + q', r\right] .$$

**Chains.** Similarly, the expected weight of chain $c = (e_1, \ldots, e_L)$ is expressed as

$$\begin{aligned} \mathbb{E}\left[F(c, r + f) \mid q, r\right] &= \sum_{k=1}^{L} \left(\sum_{j=1}^{k} w_j\right) \mathbb{E}\left[\prod_{j=1}^{k} (1 - r_{e_j} - f_{e_j}) \mid q, r\right] \\ &= \sum_{k=1}^{L} \left(\sum_{j=1}^{k} w_j\right) \prod_{j=1}^{k} \left(1 - \mathbb{E}\left[r_{e_j} + f_{e_j} \mid q, r\right]\right) , \end{aligned}$$

where the second step is due to the fact that $f_e$ are independent. Similarly,

$$\mathbb{E}\left[F(c, r + r' + f) \mid q + q', r\right] = \sum_{k=1}^{L} \left(\sum_{j=1}^{k} w_j\right) \prod_{j=1}^{k} \left(1 - \mathbb{E}\left[r_{e_j} + r'_{e_j} + f_{e_j} \mid q + q', r\right]\right) .$$

as before, due to Assumption 8.1 it follows that

$$\mathbb{E}\left[F(c, r + f) \mid q, r\right] \leq \mathbb{E}\left[F(c, r + r' + f) \mid q + q', r\right] .$$

□

**Lemma C.2.** *With a failure-aware matching policy, and if all edges are independent, adding a single edge to any edge query set weakly improves the objective of Problem 8.1. Formally, for any* $\boldsymbol{q}, \boldsymbol{q}' \in \{0,1\}^{|E|}$ *with* $\boldsymbol{q} + \boldsymbol{q}' \in \{0,1\}^{|E|}$ *and* $|\boldsymbol{q}'| = 1$, *and* $M(\boldsymbol{r}) \equiv M^{\texttt{FA}}(\boldsymbol{r})$,

$$V^S(\boldsymbol{q}) \leq V^S(\boldsymbol{q} + \boldsymbol{q}')$$

*Proof.* The objective of Problem 8.1 for edge set $\boldsymbol{q}$ is expressed as

$$\begin{aligned} V^S(\boldsymbol{q}) &= \mathop{\mathbb{E}}_{\boldsymbol{r}|\boldsymbol{q}} \left[ \mathop{\mathbb{E}}_{f|\boldsymbol{q},\boldsymbol{r}} \left[ \sum_{c \in \mathcal{C}} M_c^{\texttt{FA}}(\boldsymbol{r}) F(c, \boldsymbol{r} + \boldsymbol{f}) \right] \right] \\ &= \sum_{\boldsymbol{r} \in \{0,1\}^{|\boldsymbol{q}|}} P_{\boldsymbol{q}}(\boldsymbol{r}) \mathop{\mathbb{E}}_{f|\boldsymbol{q},\boldsymbol{r}} \left[ \sum_{c \in \mathcal{C}} M_c^{\texttt{FA}}(\boldsymbol{r}) F(c, \boldsymbol{r} + \boldsymbol{f}) \right] \\ &= \sum_{\boldsymbol{r} \in \{0,1\}^{|\boldsymbol{q}|}} P_{\boldsymbol{q}}(\boldsymbol{r}) \sum_{c \in \mathcal{C}} M_c^{\texttt{FA}}(\boldsymbol{r}) \mathop{\mathbb{E}}_{f|\boldsymbol{q},\boldsymbol{r}} [F(c, \boldsymbol{r} + \boldsymbol{f})] \end{aligned}$$

For edge set $\boldsymbol{q} + \boldsymbol{q}'$ we partition response variables into $\boldsymbol{r}, \boldsymbol{r}' \in \{0,1\}^{|E|}$, where $\boldsymbol{r}_e$ is the response variable for all edges $e \in \boldsymbol{q}$, and $\boldsymbol{r}_e = 0$ for all other edges (including the edge in $\boldsymbol{q}'$). Similarly, $\boldsymbol{r}'_e$ is the response variable for edge $\boldsymbol{q}'$, and $\boldsymbol{r}'_e = 0$ for all other edges. The objective of $\boldsymbol{q} + \boldsymbol{q}'$ is expressed as

$$\begin{aligned} V^S(\boldsymbol{q} + \boldsymbol{q}') &= \mathop{\mathbb{E}}_{\boldsymbol{r},\boldsymbol{r}'|\boldsymbol{q}+\boldsymbol{q}'} \left[ \mathop{\mathbb{E}}_{f|\boldsymbol{q}+\boldsymbol{q}',\boldsymbol{r}+\boldsymbol{r}'} \left[ \sum_{c \in \mathcal{C}} M_c^{\texttt{FA}}(\boldsymbol{r} + \boldsymbol{r}') F(c, \boldsymbol{r} + \boldsymbol{r}' + \boldsymbol{f}) \right] \right] \\ &= \sum_{\boldsymbol{r} \in \{0,1\}^{|\boldsymbol{q}|}} P_{\boldsymbol{q}+\boldsymbol{q}'}(\boldsymbol{r}) \mathop{\mathbb{E}}_{\boldsymbol{r}'|\boldsymbol{q}+\boldsymbol{q}'} \left[ \mathop{\mathbb{E}}_{f|\boldsymbol{q}+\boldsymbol{q}',\boldsymbol{r}+\boldsymbol{r}'} \left[ \sum_{c \in \mathcal{C}} M_c^{\texttt{FA}}(\boldsymbol{r} + \boldsymbol{r}')^\top F(c, \boldsymbol{r} + \boldsymbol{r}' + \boldsymbol{f}) \right] \right] \\ &= \sum_{\boldsymbol{r} \in \{0,1\}^{|\boldsymbol{q}|}} P_{\boldsymbol{q}}(\boldsymbol{r}) \mathop{\mathbb{E}}_{\boldsymbol{r}'|\boldsymbol{q}+\boldsymbol{q}'} \left[ \mathop{\mathbb{E}}_{f|\boldsymbol{q}+\boldsymbol{q}',\boldsymbol{r}+\boldsymbol{r}'} \left[ \sum_{c \in \mathcal{C}} M_c^{\texttt{FA}}(\boldsymbol{r} + \boldsymbol{r}') F(c, \boldsymbol{r} + \boldsymbol{r}' + \boldsymbol{f}) \right] \right], \end{aligned}$$

where in the final line we replace $P_{\boldsymbol{q}+\boldsymbol{q}'}(\boldsymbol{r})$ with $P_{\boldsymbol{q}}(\boldsymbol{r})$, because each $\boldsymbol{r}_e$ is conditionally independent, given $\boldsymbol{q}_e$.

Next, by definition

$$\mathop{\mathbb{E}}_{f|q+q',r+r'}\left[\sum_{c\in\mathcal{C}} M_c^{\texttt{FA}}(r+r')F(c,r+r'+f)\right] \geq \mathop{\mathbb{E}}_{f|q+q',r+r'}\left[\sum_{c\in\mathcal{C}} x_c F(c,r+r'+f)\right] \quad \forall x\in\mathcal{M}.$$

That is, $M^{FA}$ is guaranteed to maximize this expectation, and thus

$$\begin{aligned} V^S(q+q') &\geq \sum_{r\in\{0,1\}^{|q|}} P_q(r) \mathop{\mathbb{E}}_{r'|q+q'}\left[\mathop{\mathbb{E}}_{f|q+q',r+r'}\left[\sum_{c\in\mathcal{C}} M_c^{\texttt{FA}}(r)F(c,r+r'+f)\right]\right] && \text{(B)}\\ &= \sum_{r\in\{0,1\}^{|q|}} P_q(r) \sum_{c\in\mathcal{C}} M_v^{\texttt{FA}}(r) \mathop{\mathbb{E}}_{r'|q+q'}\left[\mathop{\mathbb{E}}_{f|q+q',r+r'}\left[F(c,r+r'+f)\right]\right] && \text{(C)} \end{aligned}$$

Finally, combining (B) and (C) with Lemma C.1, the following inequality holds

$$V^S(q) \leq V^S(q+q').$$

□

Using the above lemmas, the proof of Proposition 8.3 is straightforward:

**Proposition 8.3** *If edges are independent, and Assumption 8.1 holds, then with a failure-aware matching policy the objective of Problem 8.1 is monotonic in the set of queried edges.*

*Proof.* Let $q', q'' \in \mathcal{E}$ be two edge sets such that $q' \subseteq q''$. It remains to show that, with matching policy $M(r) \equiv M^{\texttt{FA}}(r)$,

$$V^S(q'') \leq V^S(q').$$

First note that because $\mathcal{E}$ is a matroid, there is a sequence of edges $(q^{e_1},\ldots,q^{e_L})$ (with each $|q^{e_i}| = 1$) such that $q'' + q^{e_1} + \cdots + q^{e_L} = q'$. Due to Lemma C.2, the

following sequence of inequalities hold:

$$\begin{aligned} V(\boldsymbol{q}'') &\leq V(\boldsymbol{q}'' + \boldsymbol{q}^{e_1}) \\ &\leq V(\boldsymbol{q}'' + \boldsymbol{q}^{e_1} + \boldsymbol{q}^{e_2}) \\ &\dots \\ &\leq V(\boldsymbol{q}'' + \boldsymbol{q}^{e_1} + \dots + \boldsymbol{q}^{e_L}) \\ &= V(\boldsymbol{q}') \end{aligned}$$

which concludes the proof. □

## C.4 Algorithm Descriptions

Here we describe more explicitly the algorithms for `Greedy` and `MCTS`, for both the single-stage and multi-stage settings.

### C.4.1 UCB Value Estimates for `MCTS`

Both the single- and multi-stage versions of `MCTS` use the method of [186] to select the next child node to explore. The formula used to estimate a node's UCB value is

$$\frac{\frac{U}{N} - V^{min}}{V^{max} - V^{min}} + \sqrt{N^P / N}$$

where $U$ is the "UCB value estimate" calculated by `MCTS`, $N$ is the number of visits to the node, $N^P$ is the number of visits to the node's parent, and $V^{max}$ and $V^{min}$ are the largest and smallest *node values* encountered during search. In single-stage `MCTS`, all nodes have both a *node value* (the objective value of Problem 8.1) and a UCB value estimate; as described below, in multi-stage `MCTS` only query nodes have

a UCB value estimate, and only leaf nodes have a *node value* (expected matched weight, after observing responses from all queried edges).

### C.4.2 Greedy Single-Stage Edge Selection

Algorithm 9 gives a pseudocode description of `Greedy` for the single-stage setting.

**Algorithm 9** `Greedy`: Greedy Search Heuristic for Single-Stage Edge Selection

**Require:** $\mathcal{E}$: legal edge sets
  $q^R \leftarrow \mathbf{0}$  the root node (no edges)
  $V^* \leftarrow$ objective value of $q^R$ Problem 8.1
  **while** $q^R$ has children **do**
    $q' \leftarrow$ child node of $q^R$ with maximal objective value in Problem 8.1
    $q^R \leftarrow q'$
  **return** $q^R$

### C.4.3 Multi-Stage Edge Selection

In the following sections we describe multi-stage versions of `MCTS` and `Greedy`. Unlike in the single-stage setting, these algorithms take as input a set of previously-queried edges $\boldsymbol{q} \in \{0,1\}^{|E|}$ and a corresponding set of observed rejections $\boldsymbol{r} \in \{0,1\}^{|E|}$; they output the *next* edge to query.

**Multi-Stage `MCTS`.** The multi-stage search tree is somewhat more complicated than in the single-stage setting, as each node in the search tree corresponds to both a set of queried edges and a set of observed rejections. For this purpose we use two types of nodes: *outcome* nodes, and *query* nodes. Outcome nodes consist of previously-queried edges $\boldsymbol{q}$ and previously-observed rejections $\boldsymbol{r}$, and are represented by tuple $(\boldsymbol{q}, \boldsymbol{r})$. (The root of the search tree corresponds to *no* queries or observed rejections, $(\mathbf{0}, \mathbf{0})$.) The children of an outcome node are *query* nodes, represented by the next

edge to query from the parent (outcome), represented by tuple $(\boldsymbol{q}, \boldsymbol{r}, e)$. Each outcome node has one child for every edge that has not yet been queried:

$$C^O(\boldsymbol{q}, \boldsymbol{r}) \equiv \{(\boldsymbol{q}, \boldsymbol{r}, e) \mid \forall e \in E : \boldsymbol{q} + \boldsymbol{u}^e \in \mathcal{E}\}$$

where $\boldsymbol{u}^e$ is the unit vector for element $e$ ($\boldsymbol{u}^e_i = 0$ for all $i \neq e$, and $\boldsymbol{u}^e_e = 1$). Each query node has exactly two children: one where the queried edge is accepted, and one where the queried edge is rejected,

$$C^Q(\boldsymbol{q}, \boldsymbol{r}, e) \equiv \{(\boldsymbol{q} + \boldsymbol{u}^e, \boldsymbol{r}), (\boldsymbol{q} + \boldsymbol{u}^e, \boldsymbol{r} + \boldsymbol{u}^e)\}$$

As before, the *level* of a node refers to the number of queried edges: $|\boldsymbol{q}|$ for outcome nodes, and $|\boldsymbol{q}| + 1$ for query nodes.

As before we refer to nodes with no children as leaf nodes; note that only outcome nodes are leaf nodes. Unlike the single-stage version of `MCTS`, in the multi-stage setting we only consider the value of leaf nodes[1]. The value of a leaf (outcome) is

$$V^O(\boldsymbol{q}, \boldsymbol{r}) \equiv W(M(\boldsymbol{r}); \boldsymbol{q}, \boldsymbol{r}),$$

where as before $M(\boldsymbol{r})$ denotes the matching policy, and $W(\boldsymbol{x}; \boldsymbol{q}, \boldsymbol{r})$ denotes the expected matching weight of $\boldsymbol{x}$, subject to $\boldsymbol{q}$ and $\boldsymbol{r}$. The value of leaf outcome nodes is used to by `QSample` and `OSample` to guide multi-stage `MCTS`.

Algorithm 10 describes the multi-stage version of `MCTS`, taking previously-queried edges and observed responses as input. This algorithm initializes the value estimate $U[\cdot]$ and number of visits $N[\cdot]$ for query nodes in the next $L$ levels–these quantities are used in the UCB calculation.

[1]This decision was made in part because initial results indicate that edge selection is essentially monotonic.

**Algorithm 10** Multi-Stage `MCTS`

**Require:** $\mathcal{E}$: legal edge sets, $K$: maximum size of any legal edge set, $T$: time limit, $L$: number of look-ahead levels, $\boldsymbol{q}^R$: previously-queried edges, $\boldsymbol{r}^R$: previously-observed rejections

$M \leftarrow \min\{N + L, K\}$

$Q \leftarrow$ all query nodes which are descendants of $(\boldsymbol{q}^R, \boldsymbol{r}^R)$, up to level $M$

$U[(\boldsymbol{q}, \boldsymbol{r}, e)] \leftarrow 0 \; \forall (\boldsymbol{q}, \boldsymbol{r}, e) \in Q$ UCB value estimate

$N[(\boldsymbol{q}, \boldsymbol{r}, e)] \leftarrow 0 \; \forall \boldsymbol{q} \in Q$ number of visits

**while** less than time $T$ has passed **do**

`QSample`($\boldsymbol{q}^R$, $\boldsymbol{r}^R$ $M$)

$(\boldsymbol{q}^R, \boldsymbol{r}^R, e^*) \leftarrow$ child node of $(\boldsymbol{q}^R, \boldsymbol{r}^R)$ with the greatest UCB estimate **return** $e^*$

**Algorithm 11** `QSample`: Function for sampling query nodes in multi-stage `MCTS`

**Require:** $(\boldsymbol{q}, \boldsymbol{r})$: outcome node, $M$: maximum level to sample from

**if** $(\boldsymbol{q}, \boldsymbol{r})$ has no children **then return** $V^O(\boldsymbol{q}, \boldsymbol{r})$ (return the value of this outcome node)

**if** $(\boldsymbol{q}, \boldsymbol{r})$ has children **then**

**if** $|\boldsymbol{q}| < M - 1$ **then**

$(\boldsymbol{q}, \boldsymbol{r}, e') \leftarrow$ child node of $(\boldsymbol{q}, \boldsymbol{r})$ with the greatest UCB estimate

`OSample`($\boldsymbol{q}$, $\boldsymbol{r}$, $e$)

**else**

$(\boldsymbol{q}', \boldsymbol{r}') \leftarrow$ random leaf node, descendant from $(\boldsymbol{q}, \boldsymbol{r})$ **return** $V^O(\boldsymbol{q}', \boldsymbol{r}')$

**Algorithm 12** `OSample`: Function for sampling outcome nodes in multi-stage `MCTS`

**Require:** $(\boldsymbol{q}, \boldsymbol{r}, e)$: query node

$N[(\boldsymbol{q}, \boldsymbol{r}, e)] \leftarrow N[(\boldsymbol{q}, \boldsymbol{r}, e)] + 1$

$\boldsymbol{q}' \leftarrow \boldsymbol{q} + \boldsymbol{u}^e$ (new query vector with edge $e$ added)

$Z \leftarrow$ randomly sample a response to edge $e$ (0 if accept, 1 if reject)

$\boldsymbol{r}' \leftarrow \boldsymbol{r} + Z\boldsymbol{u}^e$ (updated rejection vector)

$U[(\boldsymbol{q}, \boldsymbol{r}, e)] \leftarrow U[(\boldsymbol{q}, \boldsymbol{r}, e)] +$ `QSample`$(\boldsymbol{q}', \boldsymbol{r}')$

Algorithm 11 (`QSample`) samples query nodes from an outcome node, while Algorithm 12 (`OSample`) samples outcome nodes from a query node (and updates the query node's UCB value estimate).

**Multi-Stage `Greedy`.** Algorithm 13 gives a pseudocode description of the multi-stage version of `Greedy`. This search heuristic returns the next edge to query with the highest expected final matching weight, *ignoring all future queries*. In other words, this approach treats every edge as the *last* edge; one might call this heuristic "myopic" as well as greedy.

**Algorithm 13** `Greedy` Heuristic for Multi-Stage Edge Selection

**Require:** $\mathcal{E}$: legal edge sets, $\boldsymbol{q}$: previously-queried edges, $\boldsymbol{r}$: previously-observed rejections
  $e^* \leftarrow \emptyset$
  $V^* \leftarrow 0$
  **for** all $\boldsymbol{q}'$ in $\boldsymbol{q}$'s children **do**
    $e' \leftarrow$ the new edge queried in child node $\boldsymbol{q}'$
    $\boldsymbol{r}^A \leftarrow \boldsymbol{r}$
    $\boldsymbol{r}^R \leftarrow \boldsymbol{r}$
    $\boldsymbol{r}^A_{e'} \leftarrow 0$ (response scenario where $e'$ is accepted, and $\boldsymbol{r}_{e'} = 0$)
    $\boldsymbol{r}^R_{e'} \leftarrow 1$ (response scenario where $e'$ is rejected, and $\boldsymbol{r}_{e'} = 1$)
    $p^A \leftarrow$ probability that $e$ is accepted, conditional on previous responses
    $p^R \leftarrow$ probability that $e$ is rejected, conditional on previous responses
    $V' \leftarrow p^A \cdot W(M(\boldsymbol{r}^A); \boldsymbol{q}', \boldsymbol{r}^A) + p^R W(M(\boldsymbol{r}^R); \boldsymbol{q}', \boldsymbol{q}^R)$ (value of querying edge $e'$)
    **if** $V' > V^*$ **then**
      $\boldsymbol{e}^* \leftarrow e'$
      $V^* \leftarrow V'$
  **return** $\boldsymbol{e}^*$

# Appendix D: Appendix to Chapter 9

## D.1 Computational Simulations using Synthetic Data

Here we provide additional simulation results using publicly-available data. All code used in this section is available online.[1] We draw random donor and recipient locations from population distributions from four large cities around the world: Jakarta (Indonesia), Istanbul (Turkey), São Paulo (Brazil), and San Francisco (United States). All population distributions are generated using data from the Socioeconomic Data and Applications Center (SEDAC) ([280]); distance between each donor and recipient is calculated using the Haversine approximation.

**Edges:** Edges are created for all donor-recipient pairs within 15km of each other. Edge weights are generated according to random attributes assigned to donors and recipients: each recipient is randomly assigned a "nominal" edge weight $w_0 \sim U[0.01, 0.08]$, and each recipient is randomly assigned a decay parameter $k \in [5, 10, 20]$. Edge weights are calculated using the expression $w_0 \times \exp(-D/k)$, where $w_0$ is the recipient's nominal edge weight, $k$ is the donor's decay rate, and $D$ is the distance between donor and recipient (in km). These parameters are selected to roughly model the heterogeneity of real donation settings: some recipients are more popular or have a greater online presence than others (thus, higher $w_0$); some donors are more willing to travel long distances than others (thus, higher $k$).

**Recipient availability:** Half of all recipients are randomly assigned to be static

[1] https://github.com/duncanmcelfresh/blood-matching

(always available), while the other half are dynamic. Dynamic recipients have availability parameters $p_{vt}$ generated as follows: we generate alternating sequences of *low* probability ($p_{vt} = 0.1$) and *high* probability ($p_{vt} = 0.9$); each sequence has random Poisson-distributed length, with mean 4. These sequences are appended together to create $p_{vt}$ for all $t \in \mathcal{T}$; the first sequence is randomly chosen to be low or high probability. For each matching scenario, we draw a single realization of recipient availability using parameter $p_{vt}$, and this realization remains fixed for the remainder of the experiment.

**Matching Simulation:** We simulate an online donation scenario over 30 days, where each donor is notified exactly once every 7 days; each donor receives their first notification on a random day between the first and sixth day, so each donor is notified either 4 or 5 times in each simulation. We calculate recipient normalization scores by running 100 trials of `Rand`; normalization scores $m_v$ are the average weight matched with each recipient $v$ over all trials.

**Results:** For each policy we calculate the total matched weight, and the fraction of the maximum possible weight, matched by policy `Max`. To report proportionality we first calculate the normalized weight for each recipient $Y_v/m_v$: the total weight matched with a recipient, divided by their normalization score). For each policy we calculate a measure of proportionality *Gamma*, defined as:

$$Gamma \equiv \max\{\gamma \in [0,1] \mid \gamma Y_v/m_v \leq Y_{v'}/m_{v'} \forall v, v' \in V\}.$$

That is, *Gamma* is an empirical measure of proportionality for an allocation.

Figure D.1 shows simulation results for all four cities, with matching using policies `Max`, `Rand`, and `AdaptMatch` (with $\gamma = 0.0, 0.1, 0.2, \ldots, 1.0$).

The top row of this figure shows the total weight matched by each policy, and

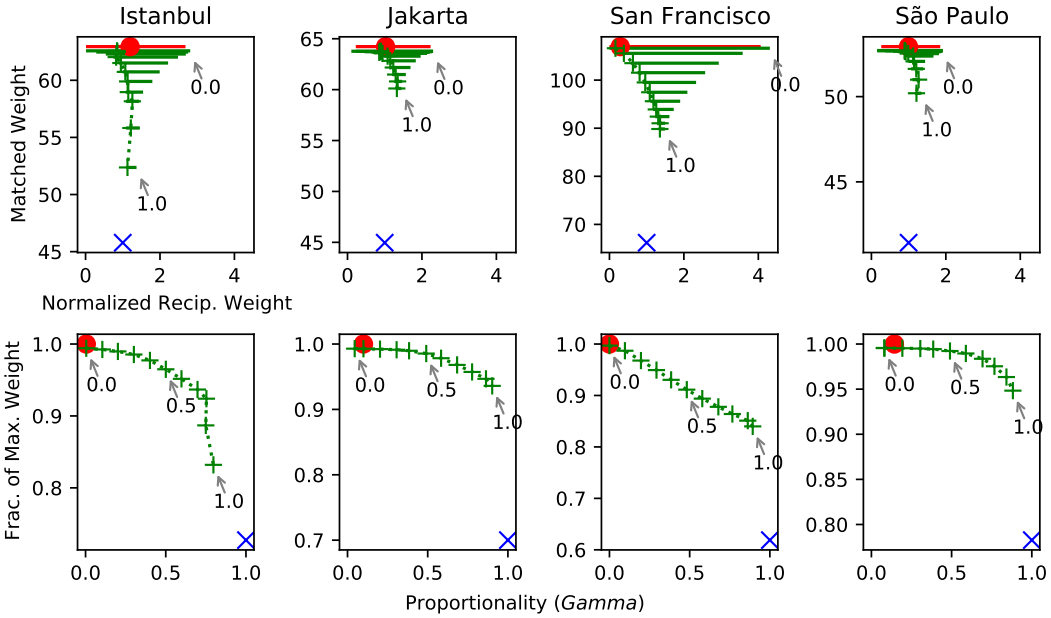



FIGURE D.1: Simulation results for four cities, for matching policy `Max` (red circle), `Rand` (blue "×") and `AdaptMatch` with $\gamma = 0.0, 0.1, \ldots, 1.0$ (green "+"). Top Row: The vertical axis shows total matched weight for `Max`, and the average matched weight for `Rand` and `AdaptMatch`; the horizontal axis shows the range of normalized recipient outcomes $Y_v/m_v$; the plot markers show the median value of the range. Bottom Row: The vertical axis shows total matched weight as a fraction of `Max`; the horizontal axis shows proportionality metric *Gamma*. Arrows on all plots indicate the $\gamma$ values for `AdaptMatch`.

the normalized recipient outcomes; horizontal error bars show the range of normalized recipient outcomes. A wider range corresponds to a less-proportional outcome, since some recipients receive much greater normalized matched weight than others. For example in San Francisco, policy `Max` matches some recipients with normalized weight of 4, while most other agents receive normalized weight near 0.

The bottom row shows matched weight as a fraction of `Max`, and proportionality *Gamma*. As expected, `Max` maximizes matched weight, though there is a wide range of recipient outcomes: for both Istanbul and San Francisco, at least one recipient remains unmatched by `Max` (and thus *Gamma* is zero).

On the other hand `Rand` by definition guarantees a proportional outcome, with $Gamma = 1$. This comes at a cost of matched weight: `Rand` matches between 60% and 80% of the weight matched by `Max`.

Policy `AdaptMatch` mediates between these two extremes, varying the trade-off between weight and proportionality with parameter $\gamma$.

Our two primary observations from these experiments are (1) while policy `Max` maximizes matched weight, it clearly treats recipients unequally; in the worst case, some recipients are never matched; (2) while policy `Rand` treats recipients equally, it results in a 20-30% reduction in matched weight. Policy `AdaptMatch` moderates smoothly between `Max` and `Rand`, using parameter $\gamma$; often, this policy yields a Pareto improvement over both extremes.

## D.2 Real-World Online Experiments

Figure D.2 shows 95% confidence intervals (Wilson score) for MA rate in the online experiment. The top plot shows the aggregated MA rate, using the cumulative number of notifications and MAs up to each day in the experiment. The bottom plot

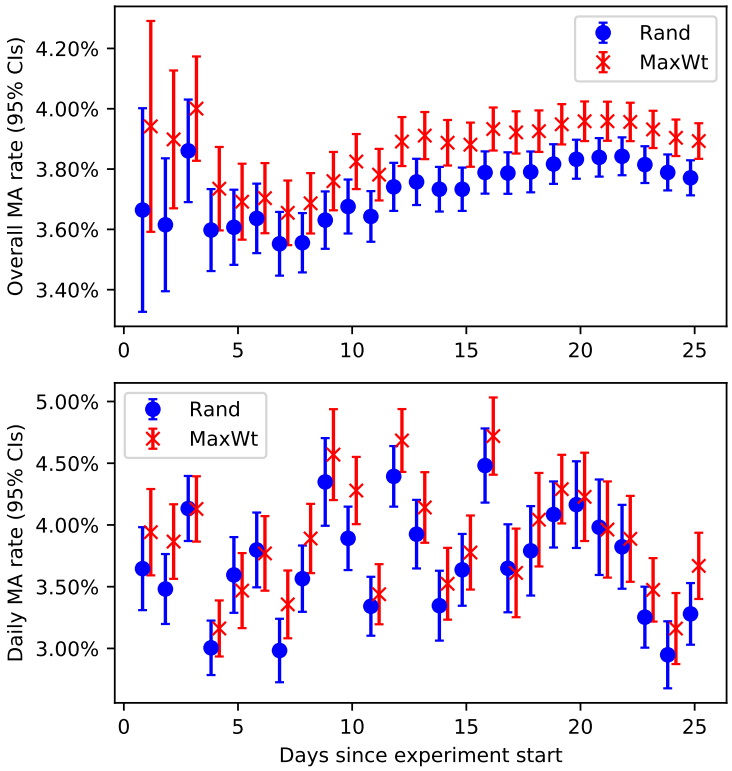



FIGURE D.2: (Top) Aggregate MA rate for both `Rand` and `Max`, for each day in the experiment. Rates are calculated using the cumulative number of notifications and MAs at each day in the experiment. Error bars show the 95% confidence interval (Wilson score interval), and points indicate the center of the interval. (Bottom) Daily MA rates, calculated using only the MAs and notifications for each day.

shows MA rates for each individual day, using only notifications sent on each day.

## D.3 Proofs

**Proof of Theorem 9.1**

*Proof.* This proof uses a reduction from $k$-EQUAL-SUM-SUBSET and PARTITION, each of which are defined as follows:

**$k$-EQUAL-SUM-SUBSET:** given a multiset $\mathcal{S}$ of positive integers $x_1, \ldots, x_N$, determine whether there are $k$ non-empty disjoint subsets $S_1, \ldots, S_K \subset \mathcal{S}$ such that the sum of integers in each subset is equal. This problem is NP-complete for any $k > 1$, and strongly NP-complete when $k$ varies as a function of $N$ and $k = \Omega(N)$ [86].

**PARTITION:** given a set $\mathcal{S}$ of positive integers $x_1, \ldots, x_N$, determine whether there is a partition of $S$ into subsets $S_1, S_2 \subset S$, with $S_1 \cup S_2 = S$, such that the sum of $S_1$ and $S_2$ are equal. This problem is NP-complete, though efficient pseudo-polynomial time algorithms exist.

We consider two cases separately: $\gamma = 1$ and $\gamma \in (0, 1)$:

- **Case 1:** $\gamma = 1$. *reduction from k-EQUAL-SUM-SUBSET.* Given an instance of $k$-EQUAL-SUM-SUBSET we construct a blood donor matching scenario as follows: let there be $k$ recipients (one for each subset) and $N$ donors (one for each integer $x_i$). Each donor $i$ has edge weight $x_i$ to every recipient, thus $G$ is a complete bipartite graph. Let all recipients have the same normalization score $m_v = 1$. In this case a non-empty $\gamma$-proportional allocation awards the same matched weight to every recipient, since all recipients have the same normalization score. If such an allocation exists, it can be used to construct an equal-sum partitioning of integers $x_i, \ldots, x_N$ into $k$ non-empty, disjoint subsets as follows: let $M_j$ be the set of donor indices matched with recipient $j$, and let subsets $S_1, \ldots, S_k$ be defined as $S_j \equiv \{x_{i'} \mid i' \in \{1, \ldots, N\}, i' \in M_j\}$; thus, $S_1, \ldots S_k$ are non-empty disjoint equal-sum subsets of integers $S$.

- **Case 2:** $\gamma \in (0, 1)$. *reduction from PARTITION.* Given an instance of PARTITION we construct a blood donor matching scenario with $N + 1$ donors and 3 recipients. Donors 1 through $N$ correspond to integers $x_1, \ldots, x_N$, and recipients 1 and 2 correspond to subsets $S_1$ and $S_2$; as before, all recipient normalization scores are $m_v = 1$. All donors 1 through $N$ are adjacent to both recipients 1 and 2, where all edges adjacent to donor $i$ have edge weight $x_i$. Donor $N + 1$ and recipient 3 are adjacent *only to each other*, with edge weight $\sum_i x_i/(2\gamma)$. In this case, a non-empty $\gamma$-proportional allocation *must* match

recipient 3, resulting in normalized matched weight $\sum_i x_i/(2\gamma)$. Due to proportionality constraints both recipients 1 and 2 must be matched with normalized matched weight at least $\sum_i x_i/2$; thus, both recipients must be matched with *exactly* edge weight $\sum_i x_i/2$. If such an allocation exists, it can be used to construct an equal sum partition: let $M_1$ and $M_2$ be the indices of donors matched with recipients 1 and 2, respectively; let subsets $S_1$ and $S_2$ be defined as $S_j \equiv \{x_{i'} \mid i' \in \{1,\ldots,N\}, i \in M_j\}$. By definition, both $S_1$ and $S_2$ are equal-sum subsets of integers $S$, and $S_1 \cup S_2 = S$.

□

**Proof of Lemma 9.2: $EP = 0$ for `Max`**

*Proof.* We provide a simple example where `Max` is 0-proportional. Let there be one donor and two recipients ($A$ and $B$); the edge to recipient $A$ has weight 0.9, while the edge to recipient $B$ has weight 1.0. Suppose there is only one time step. `Rand` matches recipient $A$ and $B$ with equal probability, while `Max` never matches $A$. Thus for policy `Max`, $E[Y_A] = 0$ and $m_A > 0$; this means that there is no $\gamma > 0$ such that this outcome is $\gamma$-proportional. □

**Proof of Lemma 9.3: $CR = 1$ for `Max`, and with $\gamma = 0$, `Max` is equivalent to `OPT`(0)**

*Proof.* First we show that the edges matched by `Max` are an optimal solution to Problem 9.1 without proportionality constraints, meaning that `Max` is an optimal solution `OPT`(0).

*Proof by contradiction.* Let $x_{et}$ be the decision variables representing edges matched by `Max` (i.e., $x_{et}$ is 1 is $e$ is matched at time $t$ by `Max`, and 0 otherwise). Suppose that $x_{et}$ is not an optimal solution to Problem 9.1. Note that without proportionality constraints, Problem 9.1 can be decomposed by both donors $u \in U$ and time steps $t \in \mathcal{T}$.

If $x_{et}$ is not an optimal solution, then there is a donor $u \in U$ and time $t \in \mathcal{T}$ such that $\sum_{e \in E^t_{u:}} x_{et} w_{et}$ which is not optimal, i.e., $e$ is not a maximal-weight edge for donor $u$ at time $t$. In this case, solution $x_{et}$ does not match a maximal-weight edge from $E^t_{u:}$, and thus $x_{et}$ was not produced by `Max`, a contradiction. □

**Proof of Lemma 9.4: $CR = 1/N$ for `Rand`**

*Proof.* Consider an example donation graph with $N$ recipients and one donor; there is one edge from the donor to each recipient, and one time step during which all edges are available. One "high-weight" recipient has edge weight 1, while the remaining $N-1$ "low-weight" recipients have edge weight $\epsilon \simeq 0$. Policy `Max` matches the high-weight recipient with total weight 1 (due to Lemma 9.3, while `Rand` matches all recipients with equal probability, with expected weight $1/N + \epsilon(N-1)/N$. As $\epsilon \to 0$, the expected matched weight of `Rand` is $1/N$, and thus $CR = 1/N$. □

**Proof of Lemma 9.6: $Z_{LP} \geq E[\texttt{OPT}(\gamma)]$**

*Proof.* Let $(x^*_{et} \mid \hat{p}_{vt})$ denote the optimal solution of Problem 9.1 for demand realization $\hat{p}_{vt}$, and let $\overline{x}^*_{et}$ denote the *expected value* of $(x^*_{et} \mid \hat{p}_{vt})$ over all demand realizations drawn from distribution $p_{et}$. Note that $\overline{x}^*_{et}$ is a feasible solution to Problem 9.1-LP: by taking the expected value of both sides of all constraints in Problem 9.1, we exactly recover Problem 9.1-LP (note that, by definition, $E[\hat{p}_{vt}] = p_{vt}$). Due to linearity of expectation, the expected objective of the offline optimal solution $(x^*_{et} \mid \hat{p}_{vt})$ is exactly equal to the objective of $\overline{x}^*_{et}$ in Problem 9.1-LP—we denote this expected objective by $E[\texttt{OPT}(\gamma)]$. In summary, the *expected* solution to Problem 9.1, $\overline{x}^*_{et}$, is a feasible solution to Problem 9.1-LP and the expected objective value of Problem 9.1 is exactly equal to the objective of $\overline{x}^*_{et}$ in Problem 9.1-LP. Therefore, $\texttt{LP}(\gamma) \geq E[\texttt{OPT}(\gamma)]$. □

**Proof of Lemma 9.7: the unconditional probability of matching $e$ at $t$ with for $\texttt{NAdapLP}(\alpha, \gamma)$ is $\alpha x^*_{et}$**

*Proof.* Let $R^t_v$ be the event that recipient $v$ is available at time $t$, when using policy $\texttt{NAdapLP}(\alpha, \gamma)$. Let $X^t_{uv}$ be the event that $u$ is matched by $\texttt{NAdapLP}(\alpha, \gamma)$ using edge $e = (u, v)$ at time $t$; note that $X^t_{uv}$ and $R^t_v$ are independent By conditioning on $R^t_v$, the probability of $X^t_{uv}$ as follows

$$\begin{aligned} X^t_{uv} &= P[X^t_{uv} | R^t_v] = \alpha \frac{x^*_{et}}{p_{vt}} p_{vt} \\ &= \alpha x^*_{et} \end{aligned}$$

□

**Proof of Lemma 9.8: $\texttt{NadapLP}(1/D, \gamma)$ is always valid**

*Proof.* Corollary 9.7.1 states that the weight matched by $\texttt{NAdapLP}(\alpha, \gamma)$ is proportional to the optimal objective of Problem D.1-LP, thus the competitive ratio of $\texttt{NAdapLP}(\alpha, \gamma)$ is $\alpha$. It remains to show that this policy is valid for $\alpha = 1/D$.

Constraints in Problem D.1-LP state that $x_{et}/p_{vt} \leq 1$; therefore $\sum_{e \in E^t_{u:}} x^*_{et}/p_{vt} \leq |E^t_{u:}| \leq D$ and $\frac{1}{D} \sum_{e \in E^t_{u:}} x^*_{et}/p_{vt} \leq 1$, meaning that this policy is valid for $\gamma = 1/D$.

□

**Proof of Lemma 9.9: $EP = \gamma$ and $CR \geq 1/D$ for $\texttt{NAdapOpt}(\gamma)$**

*Proof.* First, since $y^*_{et}$ is a feasible solution for Problem 9.2, Policy $\texttt{NAdapOpt}(\gamma)$ has expected proportionality $EP = \gamma$ due to constraints in Problem 9.2. Furthermore, if $y^*_{et}$ is an optimal solution, then the corresponding $\texttt{NAdapOpt}(\gamma)$ policy has both $EP = \gamma$, and maximal competitive ratio $CR$.

Since policy NAdapLP$(1/D, \gamma)$ achieves competitive ratio $CR = 1/D$, it follows that NAdapOpt_Fixedtime achieves a competitive ratio at least $1/D$. To further illustrate this, consider the pre-match distribution used by policy NAdapLP$(1/D)$: edge $e$ is matched at time $t$ with probability $\alpha x^*_{et}/p_{vt}$, where $x^*_{et}$ is an optimal solution to Problem 9.1-LP. Note that $\overline{y}_{et} \equiv \frac{1}{D}\frac{x^*_{et}}{p_{vt}}$ is a feasible solution to Problem 9.2 (condition $\sum_{e \in E^t_u} \overline{y}_{et} \leq 1$ is met, due to constraints in Problem 9.1-LP). Since this non-adaptive policy achieves $CR = 1/D$, an optimal non-adaptive policy (corresponding to an optimal solution of Problem 9.2) achieves competitive ratio $CR \geq 1/D$. □

## D.4 Rate-Limited Notification Policies

Rather than fixing the time steps when donors can be notified ("fixed time" policies), here we consider policies which also determine *when* to notify donors, subject to a rate-limiting constraint. As discussed in Section 9.4 it is necessary to limit the frequency that donors receive notifications; here, we require that donors are notified *at most* once every $K$ days. As in the previous section, we first describe the offline-optimal policy for a known demand realization $\hat{p}_{vt}$; this policy is identified using an

optimal solution to Problem D.1.

$$
\begin{aligned}
\max \quad & \sum_{t\in\mathcal{T}}\sum_{e\in E} w_{et}x_{et} \\
\text{s.t.} \quad & x_{et} \in \{0,1\} && \forall e \in E\ t \in \mathcal{T} \\
& a_{ut} \in \{0,1\} && \forall u \in U\ t \in \mathcal{T} \\
& s_v \in \mathbb{R} && \forall v \in V \\
& x_{et} \le \hat{p}_{vt} && \forall e=(u,v) \in E,\ t \in \mathcal{T} \\
& x_{et} \le a_{ut} && \forall e=(u,v) \in E,\ t \in \mathcal{T} \\
& \sum_{e\in E_{u:}} x_{et} \le a_{ut} && \forall u \in U,\ t \in \mathcal{T} \\
& a_{ut} = 1 - \sum_{t'=t-K+1}^{t-1}\sum_{e\in E^{t}_{u:}} x_{et} && \forall u \in U,\ t \in \mathcal{T} \\
& s_v = \tfrac{1}{m_v}\sum_{t\in\mathcal{T}}\sum_{e\in E_{:v}} x_{et}w_{et} && \forall v \in V \\
& \gamma s_v \le s_{v'} && \forall v,v' \in V,\ v \neq v'.
\end{aligned}
\tag{D.1}
$$

This problem differs from the fixed-time setting (Problem 9.1) in that donor availability $a_{ut}$ is not pre-determined, rather it depends on past matching decisions: on time $t$, if donor $u$ has been matched in the prior $K-1$ time steps, then $a_{ut}=1$, and otherwise $a_{ut}=0$; thus, $a_{ut}\in\{0,1\}$ is an auxiliary variable defined using constraint $a_{ut} = 1 - \sum_{t'=t-K+1}^{t-1}\sum_{e\in E^{t}_{u:}} x_{et}$. Using an optimal solution to Problem D.1, offline optimal policy $\texttt{OPT}(\gamma)$ and competitive ratio $CR$ are defined identically here as in the fixed-time setting.

Further, both baseline policies `Rand` and `Max`, as well as expected proportionality metric $EP$ are defined identically here as in the fixed-time setting; however, in the rate-limited setting donors are *available* only if they have not been matched in any of the previous $K-1$ time steps. As before, `Rand` is 1-proportional by definition, while `Max` is still 0-proportional in the worst case (using the same example as in Lemma 9.2).

However, unlike in the fixed-time setting, `Max` does not always maximize competitive ratio. This is intuitive: policies `Rand` and `Max` are myopic, in the sense that they ignore changes in edge weights or donor availability over time. Instead they match donors as soon as they are available (once every $K$ days at most, if there is an available edge), which can lead to a matching with arbitrarily low weight. Consider an example donation graph with one donor and one recipient, with two time steps and $K = 2$ (the donor may be matched once). For $t = 1$ the edge weight is $\epsilon \simeq 0$, while for $t = 2$ the edge weight is 1. Since both `Max` and `Rand` both match the donor on the first time step $t = 1$, the competitive ratio $CR$ can be arbitrarily small.

**Lemma D.1.** *In the rate-limited setting, the competitive ratio for both `Max` and `Rand` is $CR = \epsilon$, where $\epsilon$ is the smallest edge weight in the graph.*

As in the previous section, we investigate stochastic non-myopic policies. Mirroring our analysis of the fixed-time setting, we first investigate non-adaptive policies, and we extend these to develop approximate adaptive policies.

**Non-Adaptive Rate-Limited Policies** The policies in this section are analogous to the non-adaptive fixed-time policies, but for a rate-limited setting. Surprisingly, the guarantees on competitive ratio and expected proportionality for these policies are the identical to those in the fixed-time setting.

We begin with a policy based on the an LP relaxation or Problem D.1, which refer to as Problem D.1-LP. As before, this relaxation is almost identical to Problem D.1; the only difference being that variables $x_{et}$ and $a_{et}$ are continuous on $[0, 1]$ rather than binary. As before, this problem yields a valid upper bound on the objective of Problem D.1.

**Lemma D.2.** *Let $Z_{LP}$ denote the optimal objective of Problem D.1-LP for matching problem $\mathcal{P} = (U, V, E, m_v, p_{vt}, \mathcal{T})$ and $\gamma \in [0, 1]$. Let $E[\texttt{OPT}(\gamma)]$ be the expected objective of the*

*offline-optimal policy, over all demand realizations. Then,* $Z_{LP} \geq E[\texttt{OPT}(\gamma)]$.

The proof of this lemma is nearly identical to that of Lemma 9.6, and we omit it here.

The first non-adaptive policy for the rate-limited setting is based on an optimal solution to Problem D.1-LP, and is analagous to `NadapLP` from the previous section:

**Definition 14** (`NAdapLP_Rate`$(\alpha, \gamma)$)**.** Let $x^*_{et}$ denote an optimal solution to Problem D.1-LP, with proportionality parameter $\gamma$. For each time step $t \in \mathcal{T}$ and each donor $u \in U$, edge $e \in E_{u:}$ is pre-matched with probability $\alpha x^*_{et} / \beta_{ut} p_{vt}$, and the donor is not pre-matched with probability $1 - \alpha \sum_{e=(u,v) \in E_{u:}} \frac{x^*_{et}}{\beta_{ut} p_{vt}}$. Each parameter $\beta_{ut}$ is equal to the probability that donor $u$ is available at time $t$ under this policy; these parameters are estimated via simulation.[2] At each time step, all donors with a pre-matched edge for the time step are matched—if both the donor and recipient are available.

Somewhat surprisingly, each of the important properties of `NAdapLP` also apply to `NAdapLP_Rate`; the proofs are nearly equivalent to the corresponding proofs in the fixed-time setting, and we omit them here.

**Lemma D.3.** *Let* $x^*_{et}$ *be the optimal solution used in policy* `NAdapLP_Rate`$(\alpha, \gamma)$*. The unconditional probability that edge* $e$ *is matched at time* $t$ *by policy* `NAdapLP_Rate` *is* $\alpha x^*_{et}$.

**Corollary D.3.1.** `NAdapLP_Rate`$(\alpha, \gamma)$ *achieves competitive ratio* $CR = \alpha$.

**Corollary D.3.2.** `NAdapLP_Rate`$(\alpha, \gamma)$ *is always* $\gamma$*-proportional in expectation.*

As in the fixed-time setting, policy `NAdapL_Rate`$(\alpha, \gamma)$ can only be implemented if $\alpha$ is small enough that the policy is valid.

**Lemma D.4.** *Policy* `NAdapLP_Rate`$(1/(2D), \gamma)$ *is always valid and achieves a competitive ratio of* $CR \geq 1/(2D)$ *for all* $\gamma \in [0, 1]$*, where* $D$ *is the maximum degree of any donor:* $D \equiv \max_{u \in U} |E_{u:}|$.

[2] Please see [111] for a discussion of this method, which inspired this policy.

*Proof.* First we observe that NAdapLP_Rate$(\alpha, \gamma)$ is valid if $\alpha \leq \beta_{ut}/D$, where $D$. Next, we show that $\beta_{ut} \geq 1/2$ for policy NAdapLP_Rate$(1/(2D), \gamma)$; thus we set $\alpha \leftarrow 1/(2D)$ for the remainder of this proof. To demonstrate this, we assume that all donors are available at the first time step ($\beta_{u1} = 1$), and thus $\beta_{u1} \geq 1/2$. For all other time steps, $\beta_{ut}$ is expressible as

$$\beta_{ut} \equiv 1 - \sum_{t'=t-K+1}^{t-1} P_{ut}$$

where $X_{et}$ is the probability that $u$ is matched at time $t$. Thus, we can express $\beta_{ut}$ in terms of the decision variables $x^*_{et}$ used to define policy NAdapLP_Rate$(\alpha, \gamma)$:

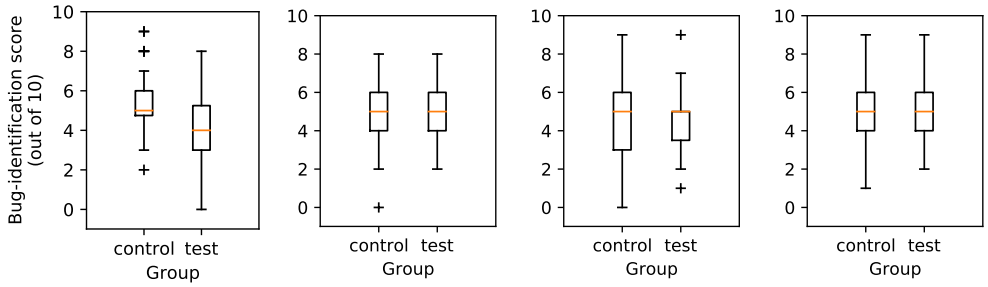
$$\begin{aligned}
\beta_{ut} &= 1 - \sum_{t'=t-K+1}^{t-1} \sum_{e \in E_{:u}} \alpha x^*_{et} \\
&\geq 1 - \frac{\alpha}{D} \\
&= 1/2
\end{aligned}$$

Thus, for $\alpha = 1/(2D)$, $\beta_{ut} \geq 1/2$, and $\alpha \leq \beta_{ut}/D$. Therefore policy NAdapLP_Rate$(1/(2D))$ is always valid; due to Corollary D.3.1 this policy achieves competitive ratio $CR = 1/(2D)$. □

# Appendix E: Appendix to Chapter 12

## E.1 Survey Results

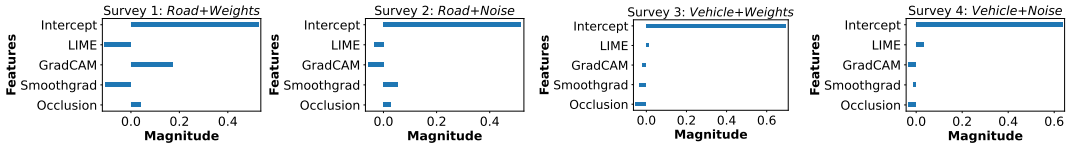
Figure E.1 shows box plots of participant bug scores in both `Control` and `Test` for each survey.




(A) Survey 1 (B) Survey 2 (C) Survey 3 (D) Survey 4

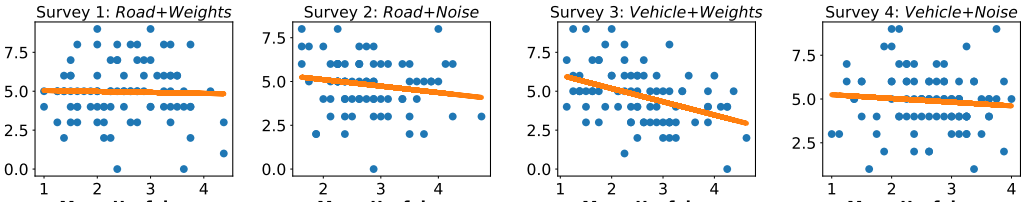
FIGURE E.1: Box plots of participant scores in Part II. Scores are equal to the number of examples (out of 10) in Part II where participants correctly identified whether or not there the example was generated by a buggy model.

**Participant usefulness ratings are not correlated with bug score.** Figure E.2 shows a scatter plot and linear regression for participants' bug scores and the mean usefulness rating (from Part I) for the explanation method used in Part II. There is no significant correlation between usefulness ratings and bug scores, and if anything a slight negative correlation in Surveys 2 and 3. This suggests that asking for users' ratings of explanation methods is not necessarily a good way to identify methods that help users complete downstream tasks.




(A) Survey 1 (B) Survey 2 (C) Survey 3 (D) Survey 4

FIGURE E.3: [Linear Regression] Coefficients of Linear Regression used to predict a user's probability of correctly identifying bugs. In Survey 3 (*Vehicle + Weights*, Fig E.3c) all method's scores are negatively correlated with the probability of correct prediction.




(A) Survey 1 (B) Survey 2 (C) Survey 3 (D) Survey 4

FIGURE E.2: The mean usefulness rating (self-reported by participants) of the explanation method shown during Part II (evaluation phase) of the survey. We see that in most cases there's no correlation between the user-reported usefulness of an explanation method and the ability of that method to help the user diagnose bugs.

TABLE E.1: Regression models to predict correctness in identifying bugs (per user) given the self-reported scores of each explanation method as an input features. We see that for most part, the self-reported scores of users are not a good predictor for the probability that the user will correctly identify the bug. There is one noticeable exception in survey 3 (*Vehicle + Weights*). However, even in this case, we see a negative correlation between features (self-reported usefulness scores of each method) and the probability to predict a bug correctly (see Fig E.3). This is further evidence that users cannot identify which explanation method is useful for the downstream task.

| | Ridge | | | Lasso | | | ElasticNet | | | Linear Regression | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Survey | $R^2$ | RMSE | MAE | $R^2$ | RMSE | MAE | $R^2$ | RMSE | MAE | $R^2$ | RMSE | MAE |
| 1 (*Road+Weights*) | -0.110 | 0.026 | 0.136 | -0.161 | 0.027 | 0.139 | -0.161 | 0.027 | 0.139 | -0.267 | 0.030 | 0.146 |
| 2 (*Road+Noise*) | -0.125 | 0.039 | 0.152 | -0.150 | 0.040 | 0.150 | -0.150 | 0.040 | 0.150 | -0.160 | 0.040 | 0.166 |
| 3 (*Vehicle+Weights*) | 0.094 | 0.028 | 0.120 | -0.010 | 0.031 | 0.137 | -0.010 | 0.031 | 0.137 | 0.094 | 0.028 | 0.127 |
| 4 (*Vehicle+Noise*) | -0.182 | 0.035 | 0.155 | -0.168 | 0.035 | 0.158 | -0.168 | 0.035 | 0.158 | -0.206 | 0.036 | 0.154 |

## E.2 Model, Explanations, and Dataset Details

All code used for generating the target images and explanations is available online.[1]

[1] https://github.com/duncanmcelfresh/learning-explanations

TABLE E.2: Probability of correct bug prediction given a particular explanation in part II.

| | Showed Explanation Method | | | |
|---|---|---|---|---|
| Survey | A | B | C | D |
| 1 (*Road+Weights*) | $0.512 \pm 0.145$ | $\mathbf{0.544 \pm 0.173}$ | $0.517 \pm 0.163$ | $0.395 \pm 0.210$ |
| 2 (*Road+Noise*) | $\mathbf{0.516 \pm 0.114}$ | $0.467 \pm 0.156$ | $0.447 \pm 0.189$ | $0.507 \pm 0.144$ |
| 3 (*Vehicle+Weights*) | $0.483 \pm 0.174$ | $\mathbf{0.529 \pm 0.202}$ | $0.377 \pm 0.172$ | $0.470 \pm 0.162$ |
| 4 (*Vehicle+Noise*) | $0.536 \pm 0.116$ | $0.417 \pm 0.142$ | $0.414 \pm 0.188$ | $\mathbf{0.560 \pm 0.165}$ |

### E.2.1 Dataset

All target images were generated using random 250x250 crops from the Cityscapes dataset. Target images are normalized for both training and testing, and training images are flipped both horizontally and vertically, each with probability 1/2. All image labels are derived from pixel-level semantic labeling from the Cityscapes dataset.

***Road* Scenario** Target images are labeled 1 (road) if more than 10% of the target image pixels have semantic label "road" (label 7) in the Cityscapes dataset; otherwise the target image is labeled 0 (no road). Images which have more than 60% road pixels are discarded during training and testing, since the explanations generated for these images were not visually informative.

***Vehicle* Scenario** Target images are labeled 1 (vehicle) if more than 25% of the target image pixels have a vehicle semantic label, including cars, trucks, buses, rail cars, trailers, license plates, or caravans in the Cityscapes dataset (labels 26, 27, 28, 29, 30, 31, -1). Otherwise the target image is labeled 0 (no vehicle).

### E.2.2 Model

We use a pre-trained Resnet50 model implemented in Pytorch [240] To use this model for binary classification tasks *Vehicle* and *Road*, we replace the final Resnet50 layer with a fully connected linear layer with two output features. For both *Vehicle*

and *Road* scenarios, we train this model using 30 epochs on a random 50% subset of the Cityscapes dataset; for each epoch, one random crop is taken from each training image.

### E.2.3 Explanations

We use four explanation methods for each survey, described below. For each explanation method we create a masked image, where unmasked regions are "important" for the positive target label (road or vehicles). For each method we test several parameter values, and we manually select explanations that appear visually informative to us. In some cases, certain parameter values yield useless explanations (an empty or complete mask), and sometimes all explanations were nearly identical irrespective of parameter values. Methods A, B, and C return an importance "image" (or attribution image), with a numerical value for each pixel. For these methods we first normalize this importance image to have range $[0, 1]$, and then create a mask by "masking out" all pixels with normalized importance less than threshold $t \in [0, 1]$. All pixels with normalized importance below this threshold are displayed in grayscale and with stripes in the explanation.

**Method A (Occlusion).** We use the occlusion (or "sliding window") approach of Zeiler and Fergus [329], implemented in Captum.[2] We generate attribution images using the following parameter sets: (stride=10, window=50, $t = 0.8$), (stride=8, window=20, $t = 0.6$), (stride=20, window=60, $t = 0.5$)/

**Method B (Smoothgrad).** We use a publicly available[3] implementation of Guided Smoothgrad [287], with thresholds $t \in \{0.5, 0.7, 0.9\}$.

[2] https://captum.ai/
[3] https://github.com/hs2k/pytorch-smoothgrad

**Method C (Guided GradCAM).** We use the Captum implementation of Guided GradCAM [282], with thresholds $t \in \{0.1, 0.2, \ldots, 0.9\}$.

**Method D (LIME).** We use the publicly available[4] implementation of LIME [261]. To generate different LIME explanations we vary the number of *features* (superpixels) returned by LIME. All LIME explanations are for the positive class (road or vehicle), and are generated using 1000 samples. We generate different explanations by varying the number of features returned; we use 5, 50, and 100 features.

## E.3 Survey Details

Each of the four surveys were conducted via Qualtrics, and participants were recruited via Amazon Mechanical Turk. Prior to completing the survey, each participant agreed to a consent form. Section E.3.1 contains the consent form, Section E.3.2 contains the survey transcript of all questions in the *Road* scenario (Surveys 1 and 2), and Section E.3.3 contains the survey transcript of all questions in the *Vehicle* scenario (Surveys 3 and 4). Section E.3.4 contains all questions shown to all participants, including demographic questions. Surveys 1 and 2 differ only in the images and explanations shown to each participant; the same is true of Surveys 3 and 4.

### E.3.1 Consent Form

**Project Title** Machine Learning Model Explanation

**Purpose of the Study** This research is being conducted by [PI Name] at [Institution]. We are inviting you to participate in this research project because you are

[4] https://github.com/marcotcr/lime

above 18, you are fluent in English, you are living in the U.S., and you have an acceptance rate of at least 95% on Amazon Mechanical Turk. The purpose of this research project is to understand the usefulness of machine learning explanation methods.

**Procedures** The procedures will start with reading a brief description of a scenario where machine learning is used by a self-driving car. You will then be shown some examples of the machine learning input and output, and asked to answer some questions about the model performance. The questions will ask about how useful the machine learning output is, and you will be asked to predict the machine learning model behavior. The entire survey will take approximately 20 minutes or less.

**Potential Risks and Discomforts** There are several questions to answer over the course of this study, so you may find yourself growing tired towards the end. Outside of this, there are minimal risks to participating in this research study. All data collected in this study will be maintained securely (see Confidentiality section) and will be deleted at the conclusion of the study.

However, if at any time you feel that you wish to terminate your participation for any reason, you are permitted to do so.

**Potential Benefits** There are no direct benefits from participating in this research. We hope that, in the future, other people might benefit from this study through better explanation of machine learning models.

**Confidentiality** Any potential loss of confidentiality will be minimized by storing all data (including information such as MTurk IDs and demographics) (a) in a password-protected computer located at [Institution] or (b) using a trusted third party (Qualtrics). Personally identifiable information that is collected (MTurk IDs, IP

addresses, cookies) will be deleted upon study completion. All other data gathered will be stored for three years post study completion, after which it will be erased.

The only persons that will have access to the data are the Principle Investigator and the Co-Investigators.

If we write a report or article about this research project, your identity will be protected to the maximum extent possible. Your information may be shared with representatives of [Institution] or governmental authorities if you or someone else is in danger or if we are required to do so by law. Compensation You will receive $3 for completing this survey, and you will receive a bonus of $2 (total compensation $5) if you correctly identify bugs in the model on more than 60% of the images during the second part of this survey. You will be responsible for any taxes assessed on the compensation.

If you will earn $100 or more as a research participant in this study, you must provide your name, address and SSN to receive compensation.

If you do not earn over $100 only your name and address will be collected to receive compensation.

**Right to Withdraw and Questions** Your participation in this research is completely voluntary. You may choose not to take part at all. If you decide to participate in this research, you may stop participating at any time. If you decide not to participate in this study or if you stop participating at any time, you will not be penalized or lose any benefits to which you otherwise qualify.

If you decide to stop taking part in the study, if you have questions, concerns, or complaints, or if you need to report an injury related to the research, please contact the investigator: [PI Name]

**Participant Rights** If you have questions about your rights as a research participant or wish to report a research-related injury, please contact: [Institution]

**Statement of Consent** By agreeing below you indicate that you are at least 18 years of age; you have read this consent form or have had it read to you; your questions have been answered to your satisfaction and you voluntarily agree to participate in this research study. Please ensure you have made a copy of the above consent form for your records.

### E.3.2 Survey Transcript: *Road* Scenario

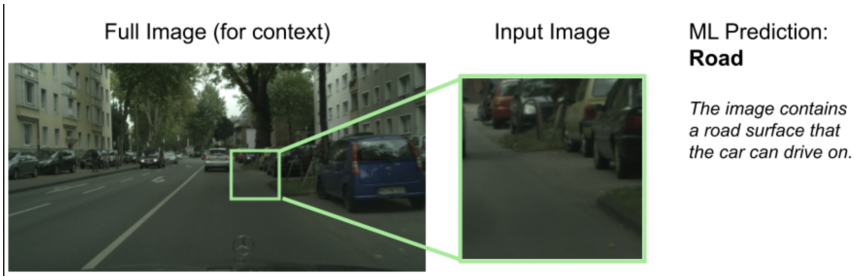
We are training a machine learning (ML) model for a self-driving car and we need your help. This ML model takes a small image (250 by 250 pixels) and decides to label the image as either:

**"Road":** the image contains a road surface that the car can drive on.

**"Not-Road":** the image does not contain surfaces that the car can drive on.

The correct labels (Road or Not-Road) are calculated by human experts for each image. The ML model is designed to correctly label images. Sometimes there is a bug in the model, and a bug can cause the model to incorrectly label the images.

For example, below is an example of the input image and ML prediction. The full image is shown for context.




[Page Break]

**Identifying Bugs.**

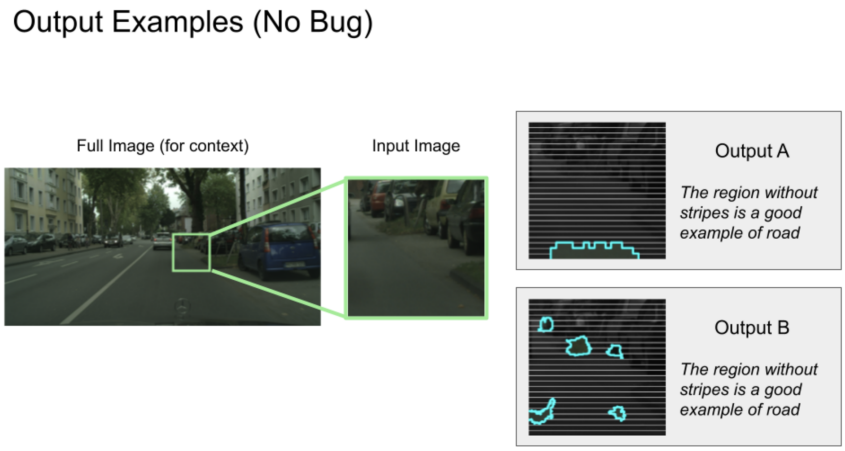
In the last part of this survey we will ask you to predict whether or not there is a bug in a model, using additional model output.

If you correctly predict whether or not there is a bug for at least 60% of the images during the second part of this survey you will earn an additional $2.00, so your total compensation for this task will be $5.00 rather than $3.00.

**Additional Model Output.**

To help you identify bugs, we provide additional output from the model. There are four types of output you can use, and we want to figure out which output is most useful to you.

Here are examples of the two different model outputs, which we call "A" and "B". Each output is itself an image. We also show the full image and the ML model label for context.




[Page Break]

**Model Output Examples**

Next we will show you several examples of four different outputs from the ML model:

- Output A
- Output B

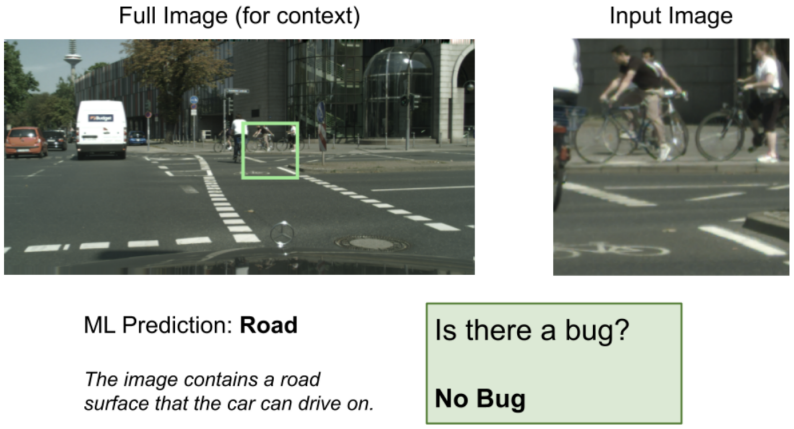
- Output C
- Output D

These outputs can help you identify whether or not there is a bug. We encourage you to pay attention to each type of output, because you will use one of these outputs during the second part of the survey.

[Page Break]

[8 examples are shown, in random order. Four of these have bugs and four do not. Only one buggy and non-buggy example are shown here.]

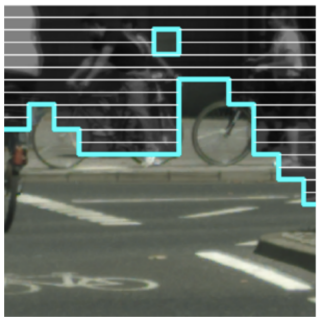
Below is an example input, with the model output. There is **No Bug** in this example.




Please rate each of the four model output images according to how useful they are for identifying whether or not there is a bug.

[The four explanations are shown in random order]

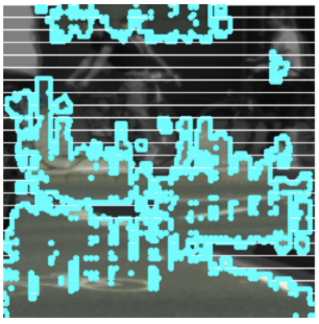
Output A: The region without stripes is a good example of road.



[Multiple choice:]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

Output B: The region without stripes is a good example of road.

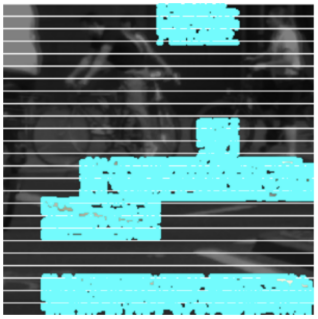


[Multiple choice]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful

- Not at all useful

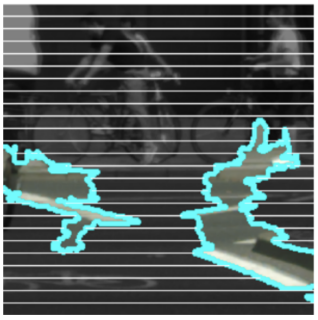
Output C: The region without stripes is a good example of road.



[Multiple choice]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

Output D: The region without stripes is a good example of road.



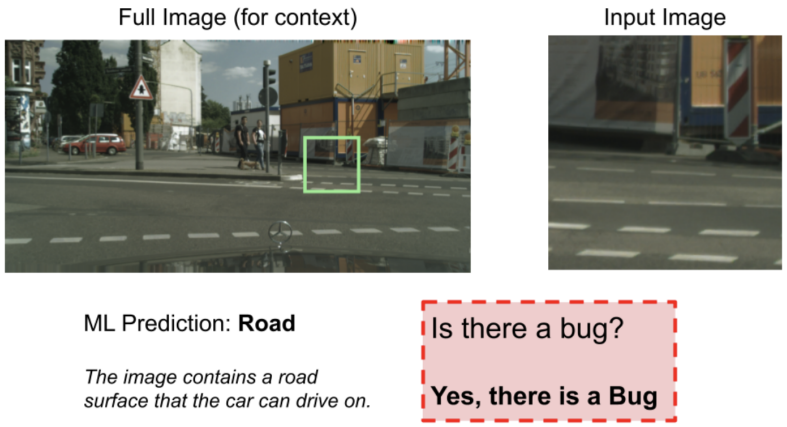
[Multiple choice]

- Extremely useful
- Very useful

- Moderately useful
- Slightly useful
- Not at all useful

[Page Break]

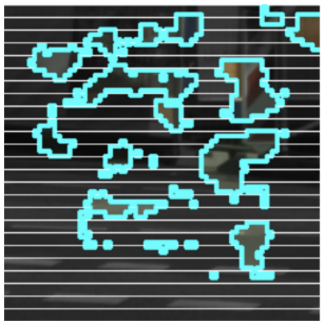
Below is an example input, with the model output. There **is a Bug** in this example.




Please rate each of the four model output images according to how useful they are for identifying whether or not there is a bug.

[The four explanations are shown in random order]

Output A: The region without stripes is a good example of road.

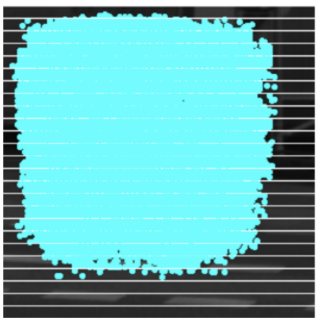


[Multiple choice:]

- Extremely useful

- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

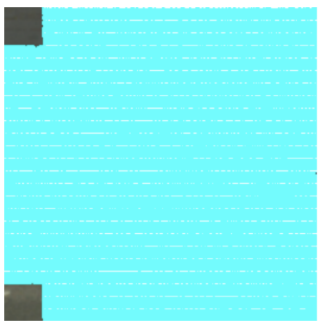
Output B: The region without stripes is a good example of road.



[Multiple choice]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

Output C: The region without stripes is a good example of road.



[Multiple choice]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

Output D: The region without stripes is a good example of road.

[Multiple choice]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

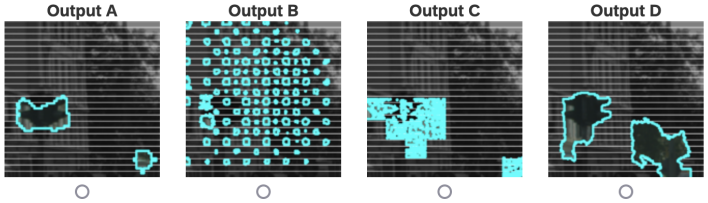
[Page Break]

Now we would like you to predict whether or not there is a bug, for 10 input images. To help you, we will provide additional model output from one of the methods we showed you earlier.

**If you correctly predict how the ML model labels these images you will earn an additional $2.00, so your total compensation for this task will be $5.00 rather than $3.00.**

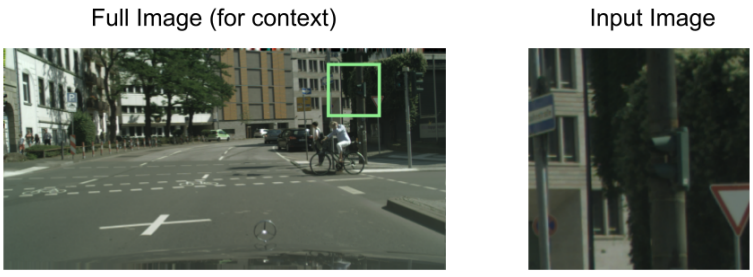
Please select the type of model output you found most useful to identify bugs in the model. Below are examples of each type of model output.

[Multiple Choice]




[Page Break]

[10 examples are shown in random order. Only one is shown here. This example has a *Noise* bug.]




ML Prediction: **No Road**

*The image does not contain a road surface that the car can drive on.*

[Only one explanation is shown, based on the participant's group (`Control` or `Test`). Method C is shown here.]

Model Output:

[Multiple Choice] Is there a bug in the model that produced this output?

- Yes, there is a bug
- No, there is not a bug
- I'm not sure

[Multiple Choice] To what extent do you agree with the following statement: The extra model output helped me identify bugs.

- Strongly agree
- Somewhat agree
- Neither agree not disagree
- Somewhat disagree
- Strongly disagree

### E.3.3 Survey Transcript: *Vehicle* Scenario

It is very important that you read each question carefully and think about your answers. The success of our research relies on our respondents being thoughtful and taking this task seriously.

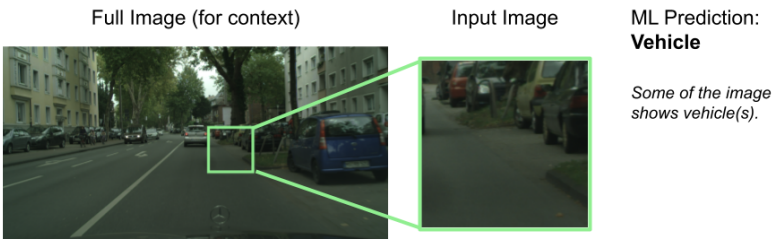
We are training a machine learning (ML) model for a self-driving car and we need your help. This ML model takes a small image (250 by 250 pixels) and decides to label the image as either:

**“Vehicle(s)”:** some of the image shows vehicle(s).

**“No Vehicle(s)":** the image does not show any vehicle(s).

The correct labels (Vehicle(s) or No Vehicle(s)) are calculated by human experts for each image. The ML model is designed to correctly label images. Sometimes there is a bug in the model, and a bug can cause the model to incorrectly label the images.

For example, below is an example of the input image and ML prediction. The full image is shown for context.




[Page Break]

**Identifying Bugs.**

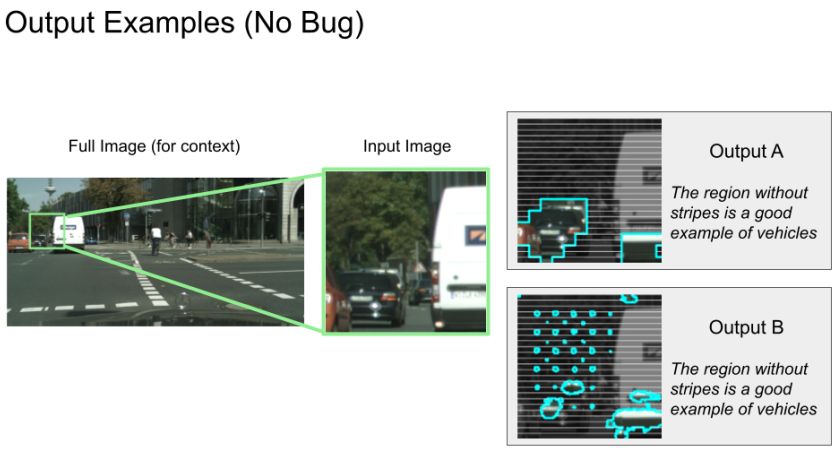
In the last part of this survey we will ask you to predict whether or not there is a bug in a model, using additional model output.

If you correctly predict whether or not there is a bug for at least 60% of the images during the second part of this survey you will earn an additional $2.00, so your total compensation for this task will be $5.00 rather than $3.00.

**Additional Model Output.**

To help you identify bugs, we provide additional output from the model. There are four types of output you can use, and we want to figure out which output is most useful to you.

Here are examples of the two different model outputs, which we call "A" and "B". Each output is itself an image. We also show the full image and the ML model label for context.




[Page Break]

**Model Output Examples**

Next we will show you several examples of four different outputs from the ML model:

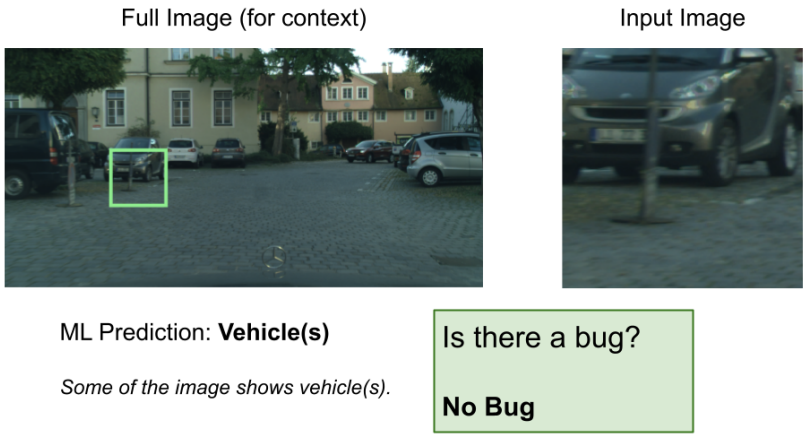
- Output A
- Output B
- Output C
- Output D

These outputs can help you identify whether or not there is a bug. We encourage you to pay attention to each type of output, because you will use one of these outputs during the second part of the survey.

[Page Break]

[8 examples are shown, in random order. Four of these have bugs and four do not. Only one buggy and non-buggy example are shown here.]

Below is an example input, with the model output. There is **No Bug** in this example.




Please rate each of the four model output images according to how useful they are for identifying whether or not there is a bug.

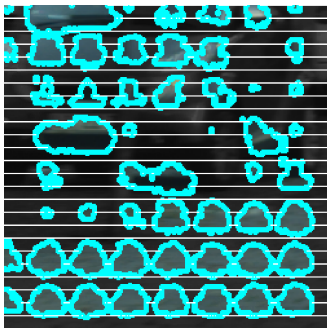
[The four explanations are shown in random order]

Output A: The region without stripes is a good example of vehicle.

[Multiple choice:]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

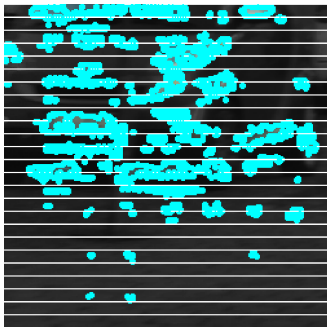
Output B: The region without stripes is a good example of vehicle.



[Multiple choice]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

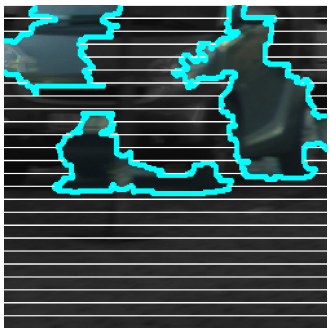
Output C: The region without stripes is a good example of vehicle.



[Multiple choice]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

Output D: The region without stripes is a good example of vehicle.

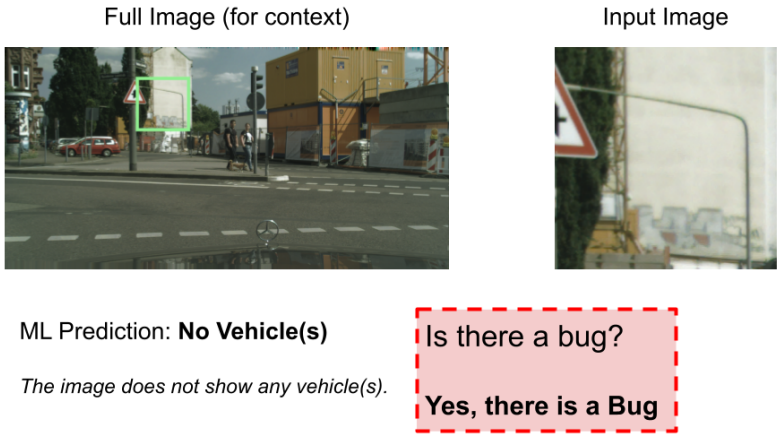


[Multiple choice]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

[Page Break]

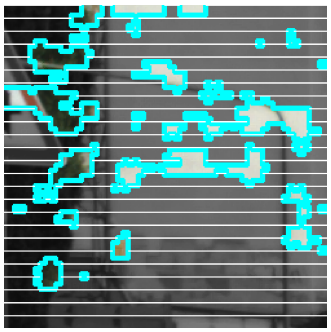
Below is an example input, with the model output. There **is a Bug** in this example.




Please rate each of the four model output images according to how useful they are for identifying whether or not there is a bug.

[The four explanations are shown in random order]

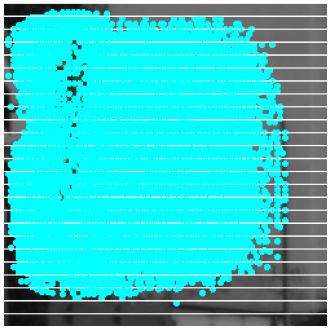
Output A: The region without stripes is a good example of vehicle.



[Multiple choice:]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

Output B: The region without stripes is a good example of vehicle.



[Multiple choice]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

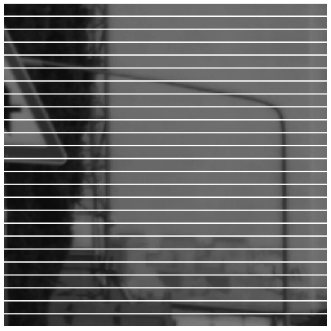
Output C: The region without stripes is a good example of vehicle.

[Multiple choice]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful

- Not at all useful

Output D: The region without stripes is a good example of vehicle.



[Multiple choice]

- Extremely useful
- Very useful
- Moderately useful
- Slightly useful
- Not at all useful

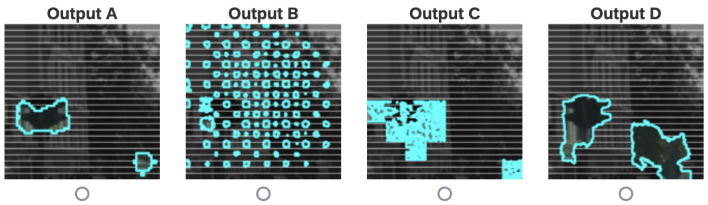
[Page Break]

Now we would like you to predict whether or not there is a bug, for 10 input images. To help you, we will provide additional model output from one of the methods we showed you earlier.

**If you correctly predict how the ML model labels these images you will earn an additional $2.00, so your total compensation for this task will be $5.00 rather than $3.00.**

Please select the type of model output you found most useful to identify bugs in the model. Below are examples of each type of model output.

[Multiple Choice]




[Page Break]

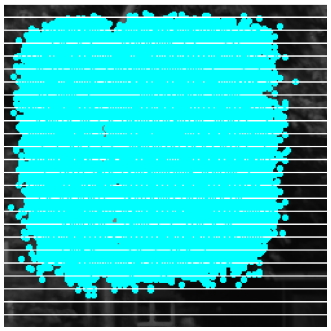
[10 examples are shown in random order. Only one is shown here. This example has a *Noise* bug.]


ML Prediction: **No Vehicle(s)**

*The image does not show any vehicle(s).*

[Only one explanation is shown, based on the participant's group (`Control` or `Test`). Method C is shown here.]

Model Output:



[Multiple Choice] Is there a bug in the model that produced this output?

- Yes, there is a bug

- No, there is not a bug
- I'm not sure

[Multiple Choice] To what extent do you agree with the following statement: The extra model output helped me identify bugs.

- Strongly agree
- Somewhat agree
- Neither agree not disagree
- Somewhat disagree
- Strongly disagree

### E.3.4 General and Demographics Questions

To what extent do you agree with each of the following statements:

**Q1.** I understand how to identify bugs in the model.

**Q2.** The model output was helpful for identifying bugs.

**Q3.** This ML model can accurately label images as Vehicle(s) or No Vehicle(s).

[All above questions are multiple choice, with the following response choices]

- Strongly agree
- Somewhat agree
- Neither agree not disagree
- Somewhat disagree
- Strongly disagree

[Page Break]

[Multiple Choice] Please specify the gender with which you most closely identify:

- Male
- Female
- Other
- Prefer not to answer

[Free response] Please specify your year of birth

[Check box] Please specify your ethnicity (you may select more than one):

- White
- Hispanic or Latinx
- Black or African American
- American Indian or Alaska Native
- Asian, Native Hawaiian, or Pacific Islander
- Other

[Multiple Choice] Please specify the highest degree or level of school you have completed:

- High school graduate, diploma or the equivalent (for example: GED)
- Some college credit, no degree
- Trade/technical/vocational training
- Associate's degree

- Bachelor's degree
- Master's degree
- Professional or doctoral degree (JD, MD, PhD)

[Multiple Choice] How much experience do you have in each of the following areas?

**Q4.** IT infrastructure/systems administration

**Q5.** Computer science/programming

**Q6.** Machine learning and data science

[For each of the above questions, the following options are given]

- No experience
- Limited experience
- Significant experience
- Expert

# Appendix F: Appendix to Chapter 13

## F.1 Online Survey Experiments

This appendix describes our survey experiments in greater detail. F.1.1 describes the online platform we used for this survey, Section F.1.2 describes Study 1 and our analysis, and Section F.1.3 describes the design of Study 2.

### F.1.1 Online Platform

All online experiments were conducted using a custom online survey platform. After agreeing to an online consent form, participants were shown background information on kidney allocation and about the patient features in this survey, shown below:

> Sometimes people with certain diseases or injuries require a kidney transplant. If they don't have a biologically compatible friend or family member who is willing to donate a kidney to them, they must wait to receive a kidney from a stranger.
>
> Choose which of two patients should receive a sole available kidney. Information about Patient A will always be on the left. Information about Patient B will always be on the right. The characteristics of each patient will change in each trial. Patients who do receive the kidney will undergo an operation that is almost always successful. Patients who do not

receive the kidney will remain on dialysis and are likely to die within a year.

After completing an online consent form, participants were asked to respond to a series of comparisons between two potential kidney recipients. Each recipient is represented by three features: "number of child dependent(s)", "years old", and "drinks per day prediagnosis." Figure F.1 shows a screenshot of the decision scenario.

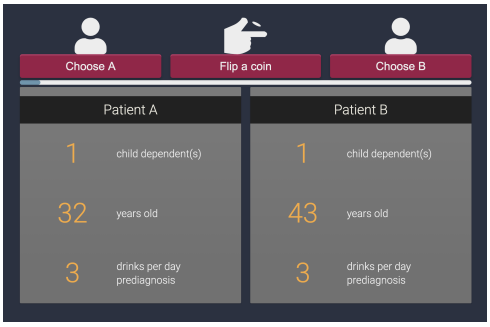



FIGURE F.1: Screenshot of a comparison question from our online survey (Study 1). This screenshot is for the group *Indecisive*; for participants in group *Strict*, the middle response option "Flip a coin" was not shown.

All participants were recruited on Amazon Mechanical Turk[1] (MTurk). We included only participants in the United States, who have completed more than 500 HITs, with HIT approval rate at least 98%, and who have not participated in any previous studies for this project.

### F.1.2 Study 1

We recruited 120 participants via MTurk. One participant was excluded from the cohort due to incompleteness, leaving us a sample of N= 119 (32% female and 68% male; mean age = 35.2, SD = 10.12, 82% white) with N=60 for group *Indecisive* and N=59 for group *Strict*. On our online platform, both groups were asked to make decisions on a set of 15 pairs of hypothetical patients, whose features were predetermined *a priori*. Both groups were given the same sequence of scenarios; the

[1] https://www.mturk.com/

features of each patient in these scenarios is included in our dataset (included in the supplement and online, see below).

The *Indecisive* group were given the additional option to flip a coin, instead of choosing one of the two patients.

Anonymized responses from Study 1 are available online.[2]

**Study 1 Analysis: Hypothesis Testing** For this analysis we refer to each strict response as a "vote". For example if a participant expresses the preference for patient $A$ over patient $B$, we say this is a vote for $A$. To test hypotheses **H0-1** and **H0-2** we first identify the *majority patient* (the patient who received more votes than the other patient); the other patient is referred to as the *minority patient*. Coincidentally, the majority and minority patients were the same for both groups, *Indecisive* and *Strict*. Table F.1 shows the number of votes for the minority and majority patient for each question, for both groups.

### F.1.3 Study 2

We first recruited 150 participants using MTurk for the *Indecisive* group. Each participant was assigned a randomly generated sequence of 40 pairs of hypothetical patients, and they were presented with the option to either give a kidney to one of the patients, or flip a coin (see Figure F.1). Patient features were generated uniformly at random from the ranges:

- # dependents: 0, 1, 2
- age: 25, . . . , 70
- # drinks: 1, 2, 3, 4, 5

[2] https://github.com/duncanmcelfresh/indecision-modeling

| Q# | Group *Indecisive* | | | Group *Strict* | |
|---|---|---|---|---|---|
| | #Maj. | #Min. | #Flip | #Maj. | #Min. |
| 1 | 31 | 5 | 2 | 38 | 22 |
| 2 | 48 | 2 | 12 | 50 | 10 |
| 3 | 43 | 2 | 17 | 57 | 3 |
| 4 | 40 | 13 | 9 | 42 | 18 |
| 5 | 37 | 0 | 25 | 51 | 9 |
| 6 | 55 | 0 | 7 | 57 | 3 |
| 7 | 43 | 1 | 18 | 56 | 4 |
| 8 | 37 | 9 | 16 | 48 | 12 |
| 9 | 29 | 8 | 25 | 43 | 17 |
| 10 | 22 | 5 | 35 | 54 | 6 |
| 11 | 41 | 12 | 9 | 43 | 17 |
| 12 | 51 | 3 | 8 | 54 | 6 |
| 13 | 29 | 4 | 29 | 55 | 5 |
| 14 | 42 | 1 | 19 | 56 | 4 |
| 15 | 33 | 9 | 20 | 47 | 13 |

TABLE F.1: Number of votes for the *majority patient* (#Maj.) and *minority patient* (#Min.) for each group. The number of "flip a coin" votes (#Flip) is shown for group *Indecisive*. The right column Q# indicates the order in which the comparison was shown to each participant.

In addition, 3 or 4 attention-check pairs, in which the participant is presented with the choice between an already deceased patient and a "favorable" patient,[3] were randomly distributed in each sequence. After data collection, 18 participants were excluded for failing at least one attention check, i.e., choosing to give the kidney to the deceased patient. This leaves us N=132 participants (age distribution was 31%: 18-29, 48%: 39-30, 10%: 40-49, 6%: 50-59, 3%: 60+; gender distribution was 29%: female, 70%: male; racial distribution was 75%: white, 25% nonwhite).

Next we recruited 153 participants for group *Strict*; these participants were given the exact same task as the *Indecisive* group, but without the option to flip a coin. 21 participants were excluded from the analysis due to attention check failures, leaving us with a final sample of N=132 (age distribution was 26%: 18-29, 46%: 39-30, 17%: 40-49, 10%: 50-59, 2%: 60+; gender distribution was 36%: female, 63%: male; racial distribution was 72%: white, 28% nonwhite).

[3] A 30-year-old patient who consumed 1 alcoholic drink per week, with 2 dependents.

Anonymized responses from Study 2 are available online.[4]

## F.2 Fitting Indecision Models

In this appendix we provide additional details on the indecision models from Section 13.4, as well as details of our computational experiments.

First, in F.2.1 we motivate the score-based decision models (from Section 13.4) using an intuitive—and equivalent—representation as *response functions*. In F.2.2 we provide additional motivation for the *strict* response model from Section 13.4.1. In F.2.3 we provide additional details on our group indecision models. Finally, in F.2.4 we describe the implementation of our computational experiments.

### F.2.1 Response Functions vs Score-Based Models

In the score-based models from Section 13.4, the agent responds by evaluating a "score" for each possible response. Here we provide an intuitive motivation for each of these indecision models, framed as response *functions*. As in Section 13.4, an agent response function $R : \mathcal{I} \times \mathcal{I} \to \{0, 1, 2\}$ maps a pair of items (a comparison question) to a response. In this section, all agent response functions are expressed in terms of the agent utility function $u(\cdot)$ and threshold $\lambda$. Each response function identifies a set of *feasible* responses for the agent, which depend on the agent utility function and threshold. If there are multiple feasible responses, the agent chooses one uniformly at random. Importantly, we show below that these response functions are identical to the score-based response functions for models in Section 13.4, when the agent observes no "noise."

Below we formalize each response function, grouped by by their "causes" (see Section 13.3).

[4] https://github.com/duncanmcelfresh/indecision-modeling

Each of the functions here appears We emphasize that each of these response "functions" is in fact a multifunction, as multiple responses may be possible. However

**Difference-Based Response Functions:** `Min-`$\delta$**,** `Max-`$\delta$ Here the agent is indecisive when the utility difference between alternatives is either smaller than threshold $\lambda$ (`Min-`$\delta$) or greater than $\lambda \in \mathbb{R}_+$ (`Max-`$\delta$). The corresponding response functions are

$$\texttt{Min-}\delta: \quad R(i,j) \equiv \begin{cases} 1 & \text{if } u(i) - u(j) \geq \lambda \\ 2 & \text{if } u(i) - u(j) \leq \lambda \\ 0 & \text{if } |u(i) - u(j)| \leq \lambda \end{cases}$$

$$\texttt{Max-}\delta: \quad R(i,j) \equiv \begin{cases} 1 & \text{if } 0 \leq u(i) - u(j) \leq \lambda \\ 2 & \text{if } -\lambda \leq u(i) - u(j) \leq 0 \\ 0 & \text{if } |u(i) - u(j)| \geq \lambda \end{cases}$$

Note that in these definitions, multiple responses may be feasible (i.e., the conditions may be met for multiple responses. In this case, we assume the agent selects a feasible response uniformly at random. For example, for both models `Min-`$\delta$ and `Max-`$\delta$, if $u(i) - u(j) = \lambda$ then the agent selects a response randomly with either 1 or 0.

In these models the agent response depends on the utility difference between $i$ and $j$, $(u(i) - u(j))$. Depending on how this utility difference compares with threshold $\lambda$, the agent may be indecisive. Since the agent is indecisive only when the absolute difference in item utility ($|u(i) - u(j)|$) is too large or too small, negative $\lambda$ is not meaningful here—thus, we only consider $\lambda > 0$.

**Desirability-Based Models:** `Min-`$U$**,** `Max-`$U$ Here the agent is indecisive when the utility of *both* alternatives is below threshold $\lambda \in \mathbb{R}$ (`Min-`$U$), or when the utility of both alternatives is greater than $\lambda$ (`Max-`$U$). Unlike the difference-based models, $\lambda$ here may be positive or negative. The response functions for these models are

$$\texttt{Min-}U: \quad R(i,j) \equiv \begin{cases} 1 & \text{if } u(i) \geq \max\{u(j), \lambda\} \\ 2 & \text{if } u(j) \geq \max\{u(i), \lambda\} \\ 0 & \text{if } \lambda \geq \max\{u(i), u(j)\} \end{cases}$$

$$\texttt{Max-}U: \quad R(i,j) \equiv \begin{cases} 1 & \text{if } u(j) \leq \min\{u(i), \lambda\} \\ 2 & \text{if } u(i) \leq \min\{u(j), \lambda\} \\ 0 & \text{if } \lambda \leq \min\{u(i), u(j)\} \end{cases}$$

As before, if there are multiple feasible responses, the agent selects one feasible response uniformly at random.

Unlike the difference-based models, both positive and negative $\lambda$ are reasonable here. For example: suppose an agent is only indecisive when both alternatives are very undesirable (e.g., both items have utility less than $-100$). This agent's decisions might be best modeled by `Min-`$U$, with $\lambda = -100$.

**Conflict-Based Model:** `Dom` Here the agent is indecisive unless one alternative *dominates* the other in all features, by threshold at least $\lambda \in \mathbb{R}$. For this indecision model, we need a utility measure associated with each feature of each item; for this purpose, let $u_n(i)$ be the utility associated with feature $n$ of item $i$. As before we assume $\lambda$ may

be positive or negative. The response function for this model is

$$\texttt{Dom}: \quad R(i,j) \equiv \begin{cases} 1 & \text{if } M_{ij} \geq \max\{M_{ji}, \lambda\} \\ 2 & \text{if } M_{ji} \geq \max\{M_{ij}, \lambda\} \\ 0 & \text{if } \lambda \geq \max\{M_{ij}, M_{ji}\} \end{cases}$$

where $M_{ij} \equiv \min_{n=1,\ldots,N}\{u_n(i) - u_n(j)\}$ and $M_{ji} \equiv \min_{n=1,\ldots,N}\{u_n(j) - u_n(i)\}$. In other words, $M_{ij}$ is the *minimum* difference between the feature utilities of $i$ and $j$: if $M_{ij}$ is positive, then all features of alternative $i$ are strictly better than those of $j$. If neither $i$ nor $j$ "dominates" the other by at least threshold $\lambda$, then the agent is indecisive. As before, the agent selects uniformly at random from all feasible responses.

While these response functions appear qualitatively different from the score functions in Section 13.4, they are in fact identical under certain circumstances.

**Proposition F.1.** *For each indecision model (*`Min-`$\delta$, `Max-`$\delta$, `Min-`$U$, `Max-`$U$, `Dom`*), the response function given in Appendix F.2.1 is identical to the response function induced by score functions $S_0(\cdot,\cdot)$ and $S_1(\cdot,\cdot)$ as in Section 13.4, when the agent observes no score error. This score-induced response function is expressed as*

$$R^S(i,j) \equiv \arg\max_{r \in \{0,1,2\}} S_r(i,j)$$

*where if multiple scores are maximal (i.e., the corresponding response is* feasible*), the agent selects a response with maximal score uniformly at random.*

*Proof.* We prove equivalence for each indecision model separately. Note that, if both response functions $R^S(i,j)$ and $R(i,j)$ have the same set of *feasible* responses for a given comparison $(i,j)$, then these responses are identical–since both response function chooses a feasible response uniformly at random. Thus, we prove that the set of

*feasible responses* is the same for both $R^S(i,j)$ and $R(i,j)$, for an arbitrary comparison $(i,j)$.

`Min-`$\delta$ For score-based response function $R^S(i,j)$, response 1 is feasible if the following conditions are met

$$\begin{array}{ll} S_1(i,j) \geq S_0(i,j) & \\ S_1(i,j) \geq S_2(i,j) & \end{array} \iff \begin{array}{l} u(i) - u(j) \geq \lambda \\ u(i) - u(j) \geq 0 \end{array}$$

where the left and right side are equivalent. Note that the right side conditions are equivalent to the conditions for response 1 in $R(i,j)$, since $\lambda$ is positive. Note that the same argument holds for response 2.

Next, for score-based response function $R^S(i,j)$, response 0 is feasible if the following conditions are met

$$\begin{array}{ll} S_0(i,j) \geq S_1(i,j) & \\ S_0(i,j) \geq S_2(i,j) & \end{array} \iff \begin{array}{l} \lambda \geq u(i) - u(j) \\ \lambda \geq u(j) - u(i) \end{array}$$

and these conditions are equivalent to $|u(i) - u(j)| \leq \lambda$, since $\lambda$ is positive. This condition is equivalent to the condition for response 0 in $R(i,j)$.

`Max-`$\delta$ For score-based response function $R^S(i,j)$, response 1 is feasible if the following conditions are met

$$\begin{array}{l} S_1(i,j) \geq S_0(i,j) \\ S_1(i,j) \geq S_2(i,j) \end{array} \iff \begin{array}{l} u(i) - u(j) \\ \geq 2|u(i) - u(j)| - \lambda \\ \\ u(i) - u(j) \geq 0. \end{array}$$

Note that the first constraint right side reduces to $u(i) - u(i) \leq \lambda$; thus, these conditions are equivalent to the conditions for response 1 in $R(i,j)$. Note that the same argument holds for response 2.

Next, for score-based response function $R^S(i,j)$, response 0 is feasible if the following conditions are met

$$\begin{array}{ll} S_0(i,j) \geq S_1(i,j) & \\ & \iff \quad 2|u(i) - u(j)| - \lambda \geq u(i) - u(j) \\ S_0(i,j) \geq S_2(i,j) & \\ & \iff \quad 2|u(i) - u(j)| - \lambda \geq u(j) - u(i). \end{array}$$

There are two cases: (1) if $u(i) \geq u(j)$, then the first condition on the right side reduces to $|u(i) - u(j) \geq \lambda$, and the second condition on the right side holds trivially; (2) if $u(i) < u(j)$, then the second condition on the right side reduces to $|u(i) - u(j) \geq \lambda$, and the first condition on the right side holds trivially. In both cases, these conditions are equivalent to the conditions for response 0 in $R(i,j)$.

`Min-`$U$ For score-based response function $R^S(i,j)$, response 1 is feasible if the following conditions are met

$$\begin{array}{ll} S_1(i,j) \geq S_0(i,j) & \\ & \iff \quad u(i) \geq \lambda \\ S_1(i,j) \geq S_2(i,j) & \\ & \iff \quad u(i) \geq u(j) \end{array}$$

where the right-side conditions reduce to $u(i) \geq \max\{u(i), \lambda\}$, which is equivalent to the condition for response 1 in $R(i,j)$. Note that the same argument holds for response 2.

Next, for score-based response function $R^S(i,j)$, response 0 is feasible if the following conditions are met

$$\begin{array}{l} S_0(i,j) \geq S_1(i,j) \\ S_0(i,j) \geq S_2(i,j) \end{array} \iff \begin{array}{l} \lambda \geq u(i) \\ \lambda \geq u(j) \end{array}$$

which is equivalent to $\lambda \geq \max\{u(i), u(j)\}$, the condition for response 0 in $R(i,j)$.

`Max-`$U$ For score-based response function $R^S(i,j)$, response 1 is feasible if the following conditions are met

$$\begin{array}{l} S_1(i,j) \geq S_0(i,j) \\ S_1(i,j) \geq S_2(i,j) \end{array} \iff \begin{array}{l} u(i) - u(j) \\ \quad \geq \max\{u(i), u(j)\} - \lambda \\ \\ u(i) \geq u(j). \end{array}$$

The first condition on the right side reduces to $u(j) \leq \lambda$; thus, the right side conditions are equivalent to $u(j) \leq \min\{u(i), \lambda\}$, which is the condition for response 1 in function $R(i,j)$. Note that the same argument holds for response 2.

Next, for score-based response function $R^S(i,j)$, response 0 is feasible if the following conditions are met

$$\begin{array}{l} S_0(i,j) \geq S_1(i,j) \\ S_0(i,j) \geq S_2(i,j) \end{array} \iff \begin{array}{l} \max\{u(i), u(j)\} - \lambda \\ \quad \geq u(i) - u(j) \\ \\ \max\{u(i), u(j)\} - \lambda \\ \quad \geq u(j) - u(i). \end{array}$$

There are two cases: (1) if $u(i) \geq u(j)$, then the first condition on the right side

reduces to $u(j) \geq \lambda$, and the second condition on the right side reduces to $2u(j) - u(j) \geq \lambda$ (which holds trivially); (2) if $u(i) < u(j)$, then the second condition reduces to $u(i) \geq \lambda$ (and the first condition holds trivially). In both cases, these conditions are equivalent to $\lambda \leq \min\{u(i), u(j)\}$, which is the condition for response 0 in $R(i,j)$.

`Dom` This proof is identical to that of `Min-`$U$: let $u(i)$ and $u(j)$ be replaced by $M_{ij}$ and $M_{ji}$, respectively, and the proof is identical.

□

### F.2.2 Strict Decision Models

In Section 13.4.1 we describe how indecision models can be used to model scenarios where an indecisive agent is *required* to express a strict preference. Here we assume that the agent uses a two-step process to respond, represented in Figure F.2.

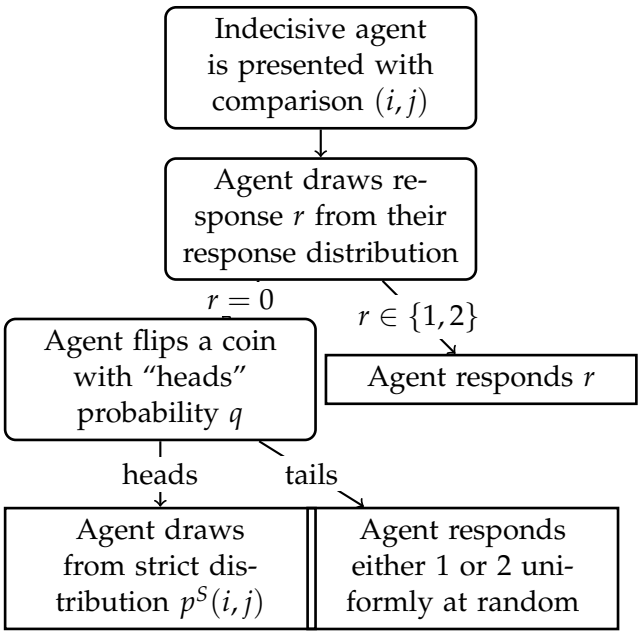



FIGURE F.2: Flowchart describing our model for an indecisive agent who is required to express a strict preference.

If the agent's coin flip is "heads" (with probability $q$), then the agent draws from a *strict* version of their response distribution, defined as

$$p^S(i,j,r) \equiv \frac{e^{S_r(i,j)}}{e^{S_1(i,j)} + e^{S_2(i,j)}}$$

for $r \in \{0,1\}$. Note that this is similar to the agent's *true* response distribution (Equation 13.1), but assigns zero probability to response 0.

The overall response distribution described in Figure F.2 has a closed-form expression, since the probability-$q$ coin flip is independent from each draw of the agent's decision function. As stated in Section 13.4.1, this distribution is

$$p_{strict}(i,j,r) \equiv \begin{cases} q\left(\frac{e^{S(i,j)}+(1/2)e^{S_0(i,j)}}{C}\right) & \\ +\frac{1-q}{D}\left(e^{S_1(i,j)}\right) & \text{if } r=1 \\ q\left(\frac{e^{S_2(i,j)}+(1/2)e^{S_0(i,j)}}{C}\right) & \\ +\frac{1-q}{D}e^{S_2(i,j)} & \text{if } r=2 \end{cases}$$

where, $C \equiv e^{S_0(i,j)} + e^{S_1(i,j)} + e^{S_2(i,j)}$ , and $D \equiv e^{S_1(i,j)} + e^{S(_2(i,j)}$. The (heads) condition from above has another interpretation: the agent chooses to sample from a "strict" logit, induced by only the score functions for strict responses, $S_1(i,j)$ and $S_1(i,j)$. We discuss this model in more detail, and provide an intuitive example, in Appendix F.2.

### F.2.3 Group Decision Models

Here we outline the mathematical group decision models from Section 13.5.2.

A set of $L$ observed responses is represented by vectors $\boldsymbol{i} \in \mathcal{I}^L$, $\boldsymbol{j} \in \mathcal{I}^L$, $\boldsymbol{r} \in \{0,1,2\}^L$, where $\boldsymbol{i}_k$ and $\boldsymbol{j}_k$ are the indices of items $i$ and $j$ in query $j$, and $\boldsymbol{r}_k$ is the observed agent's response.

`VMixture` This model is parameterized only by the best-fit models for each of its constituent voters. Let $V \in \mathbb{Z}$ be the number of voters, and let $S_r^v(\cdot,\cdot)$ be the best-fit score function for voter $v$ and response $r$. Since we take an MLE approach, the goodness-of-fit metric for these models is the log-likelihood of the model, given observed responses.

The log-likelihood for model `VMixture` is

$$\sum_{l=1}^{L} \log\left(\sum_{v=1}^{V} \frac{1}{V} p^v(\boldsymbol{i}_l, \boldsymbol{j}_l, \boldsymbol{r}_l)\right)$$

where

$$p^v(i,j,r) \equiv \frac{e^{S_r^v(i,j)}}{e^{S_0^v(i,j)} + e^{S_1^v(i,j)} + e^{S_2^v(i,j)}}$$

is the response distribution for voter $v$.

*k*`-Mixture` This model class is parameterized by $k$ distinct sets of submodel parameters: each submodel consists of a utility vector $\boldsymbol{u} \in \mathbb{R}^N$ and threshold $\lambda \in \mathbb{R}$; the *type* of each model is also a variable (i.e., a categorical variable). Weight parameters $\boldsymbol{w}$ indicate the importance of each submodel. Let $S_r^k(\cdot,\cdot)$ be the score function for model $l \in \{1,\ldots,k\}$ and response $r \in \{0,1,2\}$; these score functions depend on the type of each model (see Section 13.4). For the *k*`-Mixture` model, the log-likelihood is

$$\sum_{l=1}^{L} \log\left(\sum_{k'=1}^{k} \frac{e^{\boldsymbol{w}_{k'}}}{\sum_{n=1}^{k} e^{\boldsymbol{w}_n}} p^{k'}(\boldsymbol{i}_l, \boldsymbol{j}_l, \boldsymbol{r}_l)\right)$$

where

$$p^{k'}(i,j,r) \equiv \frac{e^{S_r^{k'}(i,j)}}{e^{S_0^{k'}(i,j)} + e^{S_1^{k'}(i,j)} + e^{S_2^{k'}(i,j)}}$$

is the response distribution for model $k'$.

### F.2.4 Experiments and Implementation

All code used for our computational experiments is available online,[5] and attached in our supplementary material. All code is written in Python 3.7, and uses packages Ax[6] for random sampling. All experiments were run on a single Intel Xeon E5-2690 node with 16GB memory.

For all experiments, models were fit by sampling several random parameter sets using a Sobol process (implemented using Ax). Each model is "trained" using a different number or random Sobol points in our experiments:

- Individual indecision models (Table 13.1): 1,000 points for *Indecisive*, and 5,000 for *Strict* (which uses an additional parameter $q$).
- Group indecision models (Table 13.2, models `Min-`$\delta$, `Max-`$\delta$, `Min-`$U$, `Max-`$U$, `Dom`, `Logit`): 5,000 points
- `VMixture`: 500 points for group *Indecisive* and 1,000 points for *Strict*, for each individual model.
- $k$-`Mixture`, $k$-`Min-`$\delta$: 100,000 points

[5] https://github.com/duncanmcelfresh/indecision-modeling
[6] https://ax.dev/

# Appendix G: Appendix to Chapter 14

## G.1 Methods

### G.1.1 Cognitive Interviews

We recruited 9 participants from the DC Metropolitan area using Craigslist. We required participants to be over 18 years of age and fluent in English. Participants ranged between the ages of 20 and 66. These interviews took place on the University of Maryland campus and lasted about 1 hour. All participants signed a written consent form prior to the interview, and were paid $30 for their time.

During these interviews, participants completed a preliminary version of the survey used in Study-1. After each survey question, we asked the participants several interview questions related to their comprehension of and feelings toward the survey. We found that some participants tended to use their own personal notions of fairness when answering comprehension questions rather than using the definition we provided. We were concerned that this would limit our ability to effectively measure comprehension. To address this problem, we rewrote several parts of our survey and added two new questions (Q14 and Q15).

### G.1.2 Non-Expert Verification

We designed this study to assess *non-expert* understanding and opinions of ML fairness metrics. To this end, we asked respondents to self-rate their level of expertise

in a variety of fields, including ML, at the end of the survey (see Appendix G.4.3). A number of participants did report having “expert" level experience in ML ($n = 2$ out of 147 in Study-1, and $n = 15$ out of 349 in Study-2). We considered removing these participants from the analyses, but ultimately did not because there was no relationship between self-reported ML expertise and comprehension score (Spearman’s rho, for both studies).

## G.2 Study-1: Detailed Results

### G.2.1 Our Survey Effectively Captures Rule Comprehension

We find that our survey is internally consistent, and effectively measures participant comprehension of demographic parity. The former we evaluated using Cronbach’s $\alpha$ and item-total correlation (discussed in §14.4.1.1), and the latter using two self-report measures and one free response question.

See Fig. G.1 for participant performance per question.

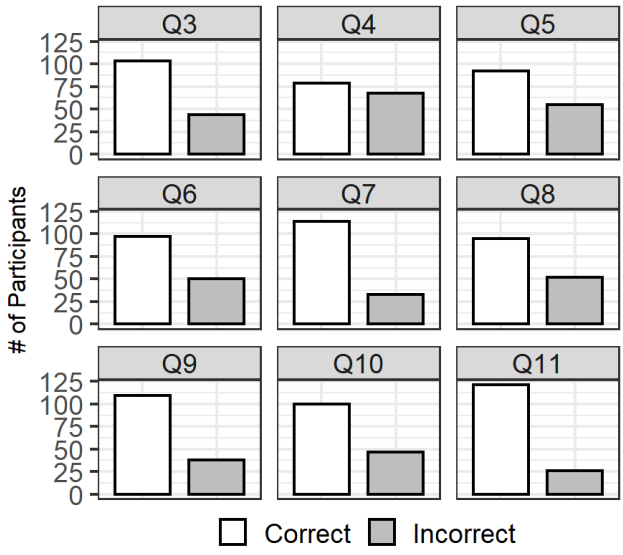



FIGURE G.1: Number of participants answering each question correctly. Each panel contains all 147 participants.

#### G.2.1.1 Self-reported rule understanding and use are reflected in comprehension score

First, we compared comprehension score to self-reported rule understanding (Q13). Higher comprehension scores were associated with greater confidence in understanding (Spearman's rho), suggesting that participants were accurately assessing their ability to apply the rule (see Fig. G.2).

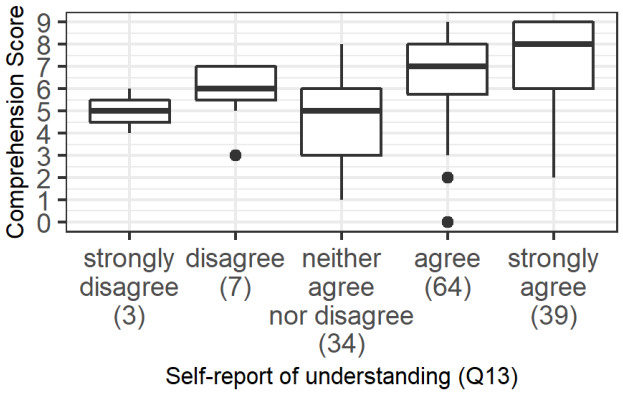



FIGURE G.2: Comprehension score grouped by response to Q13. Self-reported understanding of the rule was associated with higher comprehension scores. X-axis is reversed for figure and correlation test.

Next, we compared comprehension score to a self-report question about the participant's use of the rule (Q14) Participants who claimed to use only the rule tended to score higher than those who used their own notions of fairness or a combination thereof (K-W test, and post-hoc M-WU), suggesting that participants are answering somewhat honestly: when they try to apply the rule, comprehension scores improve (see Fig. G.3).

#### G.2.1.2 Participants with higher comprehension scores are better able to explain the rule

To further validate our comprehension score, we asked participants to explain the rule in their own words (Q12). Responses were qualitatively coded as one of five

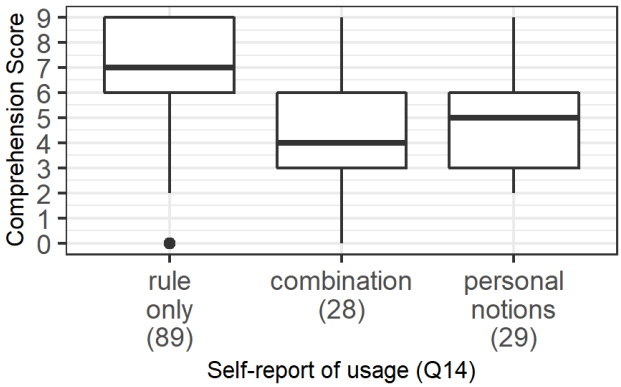



FIGURE G.3: Comprehension score grouped by response to Q14. Rule compliance (leftmost on the x-axis) was associated with higher comprehension scores. One participant who did not provide a response was excluded from the figure and relevant analysis.

categories: **correct**, **partially correct**, **neither**, **incorrect**, or **none** (as discussed in §14.4.1.1). The results of this coding can be seen can be seen in Fig. G.4. Participants providing correct explanations of the rule attained higher comprehension scores (k-W test, and post-hoc M-WU), further corroborating our claim that our comprehension score is a valid measure of fairness rule comprehension.

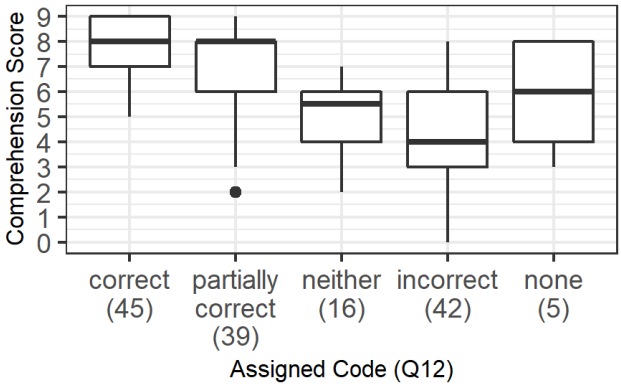



FIGURE G.4: Comprehension score grouped by code assigned to Q12 response. Participants who provided either correct or partially correct responses tended to perform better.

### G.2.2 Education Influences Comprehension

During the cognitive interview phase, we observed a possible trend of comprehension scores being lower for older participants and those with less educational attainment. If true, this would suggest that fairness explanations should be carefully validated to ensure they can be used with diverse populations. We investigated this hypothesis, in an exploratory fashion, using Poisson regression models.

Three models were tested. The first regressed score against all four demographic categories as predictors (gender, age, ethnicity, and education), the second omitted education, and the third tested only education. Models were compared using Akaike information criterion (AIC), a standard method of evaluating model quality and performing model selection [9]. Comparison by AIC revealed that model 1 (all four categories) was a better predictor for comprehension score than models 2 or 3 (AIC = 643.3, 651.2, and 660.5, respectively; difference = 0.0, 7.9, and 17.1). In model 1, only education showed correlation with comprehension score (effect size = 1.40, $p < 0.05$). Further work is needed to confirm this exploratory result.

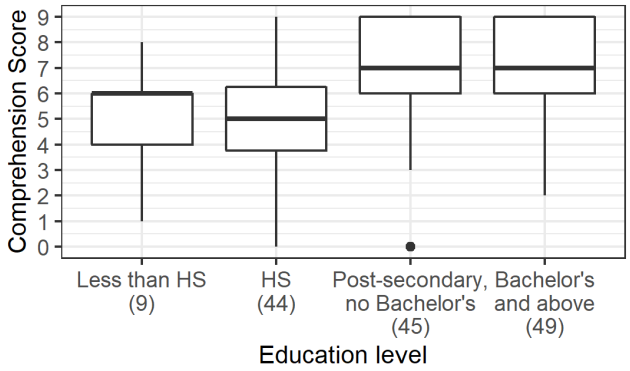



FIGURE G.5: Comprehension score grouped by education level. Higher education level was associated with higher comprehension scores.

### G.2.3 Disagreement with the Rule is Associated with Higher Comprehension Scores

Participants were asked for their opinion on the presented rule in another free response question (Q15). These responses were then qualitatively coded to capture participant sentiment towards the rule as one of five categories: **agree**, **depends**, **disagree**, **not understood**, or **none** (as discussed in §14.4.1.2).

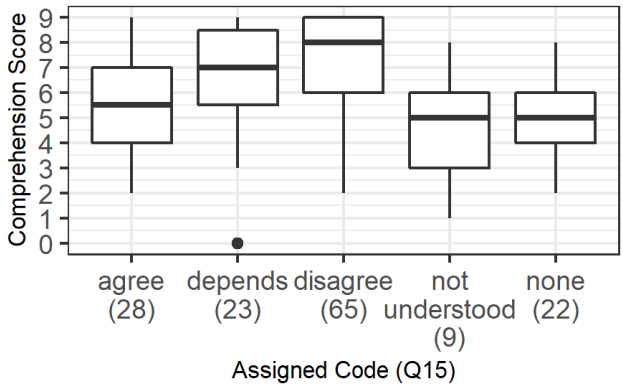



FIGURE G.6: Comprehension score grouped by code assigned to Q15 response. Participants who exhibited negative sentiment toward the rule responses tended to perform better.

This question was added based on the cognitive interviews (see Appendix G.1.1), where perception seemed to influence compliance. The results of coding Q15 can be seen in Fig. G.6. Participants who expressed disagreement with the rule performed better than those who expressed agreement, did not understand the rule, or provided no response to the question (K-W test, post-hoc M-WU). Note that this result should not be interpreted as an overall finding on the appropriateness of demographic parity. Instead we anticipate the perceptions of appropriateness of any fairness definition will be highly context-dependent.

### G.2.4 Non-Compliance is Associated with Lack of Understanding

We were interested in understanding why some participants failed to adhere to the rule, as measured by their self-report of rule usage in Q14. After labeling participants as either “non-compliant" (NC, $n = 57$) or “compliant" (C, $n = 89$), we conducted a series of $\chi^2$ tests to investigate this phenomenon.

Non-compliant participants were less likely to self-report high understanding of the rule in Q13 (see Fig. G.7). Moreover, non-compliance also appears to be associated with a reduced ability to correctly explain the rule in Q12 (see Fig. G.8). Further, negative participant sentiment towards the rule (Q15) also appears to be associated with greater compliance (see Fig. G.9). Thus, non-compliant participants appear to behave this way because they do not *understand* the rule, rather than because they do not *like* it.

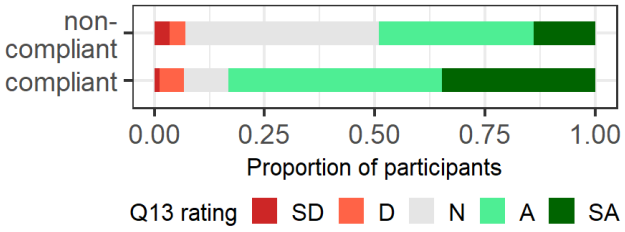



FIGURE G.7: Self-report of understanding (Q13) split by compliance (Q14). NC participants tend to report less confidence in their ability to apply the rule. SD = strongly disagree, D = disagree, N = neither agree nor disagree, A = agree, SA = strongly agree.

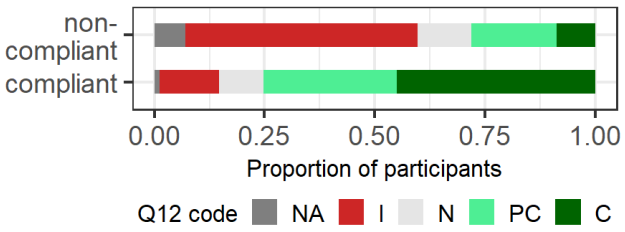



FIGURE G.8: Correctness of rule explanation (Q12) split by compliance (Q14). NC participants tend to be less able to explain the presented rule in their own words. NA = none, I = incorrect, N = neither, PC = partially correct, C = correct.

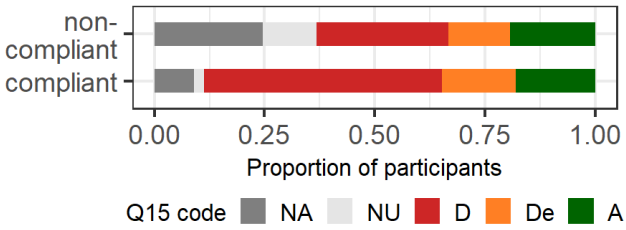



FIGURE G.9: Participant agreement with rule (Q15) split by compliance (Q14). NC participants tend to harbor less negative sentiment towards the rule. NA = none, NU = not understood, D = disagree, De = depends, A = agree.

### G.2.5 Decision Scenarios

For Study-1 we designed three decision-making scenarios to test whether the perceived importance or realism of a particular scenario influenced comprehension score. They are as follows:

- **Art Project (AP):** distributing awards for art projects to primary school students,
- **Employee Awards (EA):** distributing employee awards at a sales company, and
- **Hiring (HR):** distributing job offers to applicants.

In each scenario the students/employees/applicants are partitioned into two groups (parents' occupation for the first scenario, and binary gender for the other two scenarios). We use a between-subjects design: participants are randomly partitioned into three conditions, one for each scenario (AP, EA, or HR). For each condition we define the *fairness rule* in the context of the decision-making scenario (see Appendix G.4 for the full surveys).

Next we describe our main conclusion related to the different decision-making scenarios in Study-1: the scenario does not influence comprehension score.

#### G.2.5.1 Scenario does not Influence Comprehension Scores (RQ4)

We were concerned that less important and/or realistic scenarios would cause participants to take the survey less seriously, and therefore perform more poorly. To test this, participants were randomly assigned to a scenario, resulting in the following distribution: AP = 41, EA = 49, HR = 57.

A K-W test revealed no differences between scenarios in terms of comprehension score (mean comprehension scores: AP = 6.0, EA = 6.74, HR = 5.86). However, differences did exist between scenarios in terms of importance (assessed in Q2), measured in hours of effort deemed necessary to make the relevant decision (K-W, $p < 0.001$). Post-hoc M-WU revealed that participants believed making a decision in the AP scenario merited fewer hours of effort (mean = 3.15hrs) than in the EA (13.52hrs, $p < 0.001$) or HR (15.23hrs, $p < 0.001$) scenarios (corrected $\alpha = 0.05/3 = 0.017$). See Fig. G.10 for distributions of responses.

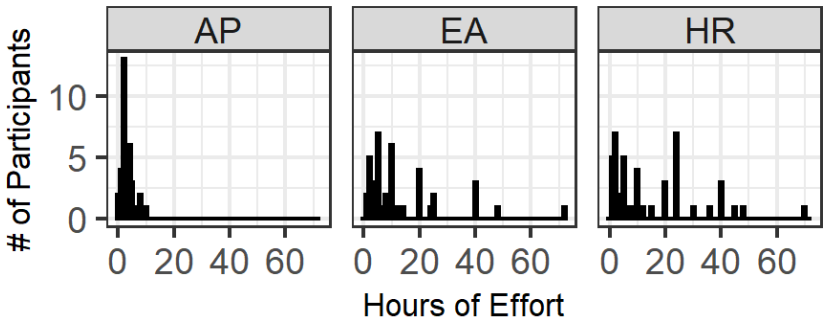



FIGURE G.10: Importance of a scenario by proxy of hours of effort necessary to make a decision in each scenario. AP merited less hours of effort than both EA and HR.

Of note, it is possible that perceived realism, assessed in Q1 on a five-point Likert scale, was also influenced by scenario (K-W, $p = 0.051$), but we may need larger sample sizes to confirm this. Regardless, while the nature of a scenario does influence participant perception in terms of importance and (possibly) realism, it does not appear to influence comprehension (at least for the scenarios we chose). For this

reason, we chose to test a single scenario (HR) in Study-2.

## G.3 Study-2: Detailed Results

### G.3.1 Model Selection

In §14.4.2.2 we assessed eleven linear regression models for predicting comprehension scores. The best fit model, determined by model selection via AIC, included only education (edu) and fairness definition (def) as regressors. The results of model selection are below in Table G.1.

TABLE G.1: Models tested in §14.4.2.2, sorted by best to least fit. The first model in the table (edu + def) is the model of best fit. dAIC = difference from model with lowest AIC value.

| **Model regressors** | **AIC** | **dAIC** |
|---|---|---|
| edu + def | -80.4 | 0 |
| edu | -72.8 | 7.6 |
| gender + edu | -70.3 | 10.1 |
| age + edu | -63.7 | 16.7 |
| gender + age + edu | -61.1 | 19.2 |
| gender + age + eth + edu + def | -61.1 | 19.2 |
| def | -60.8 | 19.6 |
| gender + age + eth + edu | -55.5 | 24.9 |
| gender + age + def | -46.4 | 34 |
| gender + age + eth + def | -41.6 | 38.8 |
| gender + age + eth | -37.2 | 43.2 |

### G.3.2 Non-Compliance

In §14.4.2.4 we sought to further investigate the findings of Study-1 with regards to compliance (Q14). To do so, we labeled those who responded (in Study-2) with either having used their own personal notions of fairness ($n = 26$) or some combination of their personal notions and the rule ($n = 148$) as “non-compliant" (NC), with the remaining $n = 174$ labeled as “compliant" (C). One participant who did not

provide a response was excluded from this analysis, conducted using KW and $\chi^2$ tests.

Non-compliant participants were less likely to self-report high understanding of the rule in Q13 (KW test, $p < 0.001$, see Fig. G.11). Moreover, non-compliance also appears to be associated with a reduced ability to correctly explain the rule in Q12 ($\chi^2$ test, $p < 0.001$, see Fig. G.12). This fits with the overall strong relationship we observed among comprehension scores, ability to explain the rule, and compliance.

Further, greater dislike towards the rule (Q15) also appears to be associated with greater compliance (KW test, $p < 0.05$, see Fig. G.13). However, there was no relationship between disagreement towards the rule (Q16) and compliance (see Fig. G.14).

These results largely corroborate the notion that non-compliant participants appear to behave this way because they do not *understand* the rule, rather than because they do not *like* it.

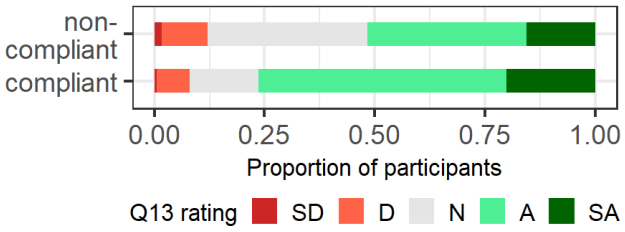



FIGURE G.11: Self-report of understanding (Q13) split by compliance (Q14). NC participants tend to report less confidence in their ability to apply the rule. SD = strongly disagree, D = disagree, N = neither agree nor disagree, A = agree, SA = strongly agree.

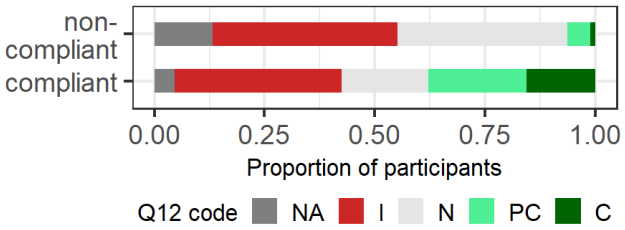



FIGURE G.12: Correctness of rule explanation (Q12) split by compliance (Q14). NC participants tend to be less able to explain the presented rule in their own words. NA = none, I = incorrect, N = neither, PC = partially correct, C = correct.

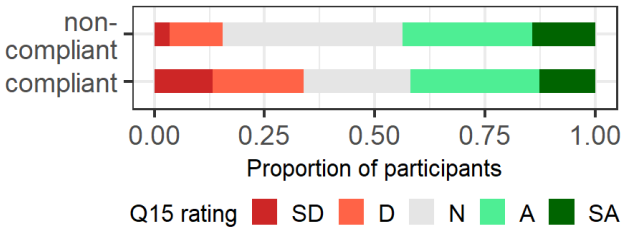



FIGURE G.13: Participant liking for rule (Q15) split by compliance (Q14). NC participants tend to dislike the rule less than C participants. SD = strongly disagree, D = disagree, N = neither agree nor disagree, A = agree, SA = strongly agree.

## G.4 Surveys

### G.4.1 Study-1 Survey

Each of the surveys are split into four main sections. The first section is the consent form which can be found in Appendix G.5. The second section describes the scenario and asks questions about the given scenario (§G.4.1.1). The third section describes the fairness metric, defined as the rule, used (in this case it is demographic parity) and asks specific questions about the metric (§G.4.1.2). Finally the last section asks for demographic information (§G.4.3).

#### G.4.1.1 Scenario descriptions and questions

The following is shown to each participant:

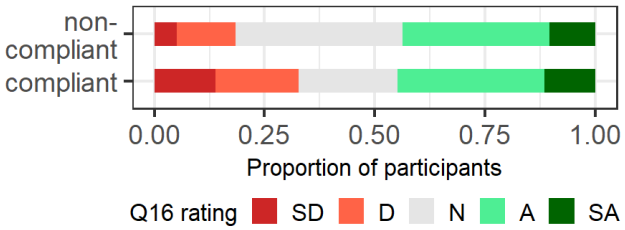



FIGURE G.14: Participant agreement with rule (Q16) split by compliance (Q14). No differences were found between NC and C participants. SD = strongly disagree, D = disagree, N = neither agree nor disagree, A = agree, SA = strongly agree.

It is very important that you read each question carefully and think about your answers. The success of our research relies on our respondents being thoughtful and taking this task seriously.

☐ I have read the above instructions carefully.

We then introduce one of three different decision making scenarios, described below, followed by two questions. Words that vary across scenario in the questions are shown as <art project, employee awards, hiring>.

**Art project** A fourth grade teacher is reviewing 20 student art projects. They will award lollipops to the top 4 students who put the most effort into their projects. The teacher knows that some of the students have artists as parents, who might have helped their children with their art project. The teacher's goal is to give out lollipops only based on the amount of effort that the student *themselves* put into their projects.

The teacher uses the following criteria to decide who should get a lollipop:

- Elaborateness of each project.
- Creativity of each project.

About 50% of the students have artists as parents, and 50% do not.

In the past, students with artists as parents typically put more effort into their projects.

In this group of students there is a wide range of project quality (as measured by elaborateness and creativity). However, this range of quality is about the same between students with artists as parents and those without.

The teacher wants to make sure that they award lollipops in a fair way, no matter whether the students' parents are artists or not.

**Employee awards** A manager at a sales company is deciding which of their 100 employees should receive each of 10 mid-year awards. The manager's goal is to give awards to employees who *will* have high net sales at the end of the year.

The manager uses the following criteria to decide who should get an award:

- Recent performance reviews
- Mid-year net sales
- Number of years on the job

About 50% of the employees are men, and 50% are women.

In the past, men have achieved higher end-of-year net sales than women.

In this group of employees, there is a wide range of qualifications (as measured by performance reviews, mid-year net sales, and number of years on the job). However, this range of qualifications is about the same between male and female employees.

The manager wants to make sure that this awards process is fair to the employees, no matter their gender.

**Hiring** A hiring manager at a new sales company is reviewing 100 new job applications. Each applicant has submitted a resume, and has had an interview. The manager will send job offers to 10 out of the 100 applicants. Their goal is to make offers to applicants who will have high net sales after a year on the job.

The manager will use the following to decide which applicants should receive job offers:

- Interview scores
- Quality of recommendation letters
- Number of years of prior experience in the field

About 50% of the applicants are men, and 50% are women.

In the past, men have achieved higher net sales than women, after one year on the job.

In this applicant pool there is a wide range of applicant quality (as measured by interview scores, recommendation letters, and years of prior experience in the field). However, the range of quality is about the same for both male and female applicants.

The hiring manager wants to make sure that this hiring process is fair to applicants, no matter their gender.

**Questions**

1. To what extent do you agree with the following statement: a scenario similar to the one described above might occur in real life.
    - Strongly agree
    - Agree
    - Neither agree nor disagree
    - Disagree
    - Strongly Disagree
2. How much effort should the <teacher, manager, hiring manager> put in to make sure this decision is fair? [short answer - number of hours]

#### G.4.1.2 Rule descriptions and questions

Unless otherwise noted the rule description is shown above each of the questions for reference. Correct answers are noted in red.

**Art project** The teacher uses the following award rule to distribute lollipops: *The fraction of students who receive lollipops that have artist parents should equal the fraction of students in the class that have artist parents. Similarly, the fraction of students who receive lollipops that do not have artist parents should equal the fraction of students in the class that do not have artist parents.*

Example 1: If 10 out of the 20 students in the class have artist parents, then 2 out of the 4 lollipops would be awarded to students with artist parents (and the remaining 2 would be awarded to students without artist parents).

Example 2: If 5 out of the 20 students in the class have artist parents, then 1 out of the 4 lollipops would be awarded to students with artist parents (and the remaining 3 would be awarded to students without artist parents).

In the next section, we will ask you some questions about the information you have just read. Please note that this is not a test of your abilities. We want to measure the quality of the description you read, not your ability to take tests or answer questions.

**Please note that we ask you to apply and use ONLY the above award rule when answering the following questions. You will have an opportunity to state your opinions and feelings on the rule later in the survey.**

3. Suppose a different teacher is considering awarding lollipops to the whole 4th grade. There are 100 students with artist parents, and 200 students without artist parents. The teacher decides to award 10 lollipops to students with artist

parents. **Assuming the teacher is required to use the award rule above**, how many students without artist parents need to receive lollipops?

(a) 10

(b) 20

(c) 40

(d) 50

4. **Assuming the teacher is required to use the award rule above**, in which of these cases can a teacher award more lollipops to students without artist parents than to students with artist parents?

   (a) When the students without artist parents have higher-quality projects (i.e., more elaborate and more creative) than those with artist parents.

   (b) When there are more students without artist parents than those with artist parents.

   (c) When students without artist parents have more creative projects than those with artist parents.

   (d) This cannot happen under the award rule.

5. **Assuming the teacher is required to use the award rule above**, is the following statement TRUE OR FALSE: Even if a student with artist parents has a project that is of the same quality (i.e., equally elaborate and equally creative) as another project by a student without artist parents, they can be treated differently (ie., only one of the students might get a lollipop).

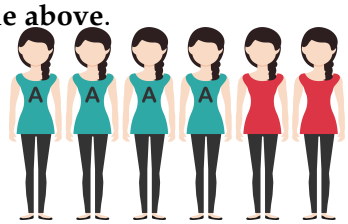
6. **Assuming the teacher is required to use the award rule above**, is the following statement TRUE OR FALSE: If all students without artist parents have

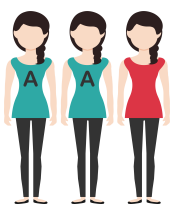
low-quality projects (i.e., low elaborateness and low creativity), but the teacher awards lollipops to some of them, then any lollipops awarded to students with artist parents must be awarded to those who have low-quality projects.

7. **Assuming the teacher is required to use the award rule above**, is the following statement TRUE OR FALSE: Suppose the teacher is distributing 10 lollipops amongst a pool of students that includes students with and without artist parents. Even if all students with artist parents have low-quality (i.e., low elaborateness and low creativity) projects, some of them must still receive lollipops.

8. **Assuming the teacher is required to use the award rule above**, is the following statement TRUE OR FALSE: This award rule always allows the teacher to award lollipops exclusively to the students who have the highest quality (i.e., most elaborate and most creative) projects.

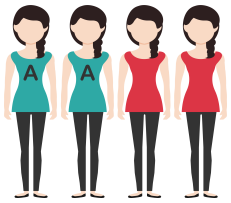
In the two examples above there are 20 students. Consider a different scenario, with **6 students – 4 with artist parents and 2 without, as illustrated below**. The next three questions each give a potential outcome for all six students (i.e., which of the 6 students receive awards). Please indicate which of the outcomes follow **the award rule above**.




9. Alternative scenario 1: 

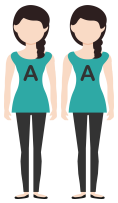

   Does this distribution of awards obey the **award rule**? Yes

10. Alternative scenario 2:




Does this distribution of awards obey the **award rule**? No

11. Alternative scenario 3:




Does this distribution of awards obey the **award rule**? No

12. In your own words, explain the **award rule**. [short answer] (The rule is not shown above this question)

13. To what extent do you agree with the following statement: I am confident I know how to **apply the award rule described above**?

    - Strongly agree
    - Agree
    - Neither agree nor disagree
    - Disagree
    - Strongly Disagree

14. Please select the choice that best describes your experience: When I answered the previous questions...

    (a) I applied the provided award rule only.

    (b) I used my own ideas of what the correct award decision should be rather than the provided award rule.

(c) I used a combination of the provided award rule and my own ideas of what the correct award decision should be.

15. What is your opinion on the award rule? Please explain why. [short answer]

16. Suppose that you are the teacher whose job it is to distribute lollipops to students based on the criteria listed above (i.e., elaborateness of each project, creativity of each project). How would you ensure that this process is fair? [short answer]

17. Was there anything about this survey that was hard to understand or answer? [short answer]

**Employee awards** The manager uses the following award rule to distribute awards: *The fraction of employees who receive awards that are female should equal the fraction of employees that are female. Similarly, fraction of employees who receive awards that are male should equal the fraction of employees that are male.*

Example 1: If there are 50 female employees out of 100, then 5 out of the 10 awards should be awarded to female employees (and the remaining 5 would be made to male employees).

Example 2: If there are 30 female employees out of 100, then 3 out of the 10 awards should be awarded to female employees (and the remaining 7 would be made to male employees).

In the next section, we will ask you some questions about the information you have just read. Please note that this is not a test of your abilities. We want to measure the quality of the description you read, not your ability to take tests or answer questions.

**Please note that we ask you to apply and use ONLY the above award rule when answering the following questions. You will have an opportunity to state your opinions and feelings on the rule later in the survey.**

3. Suppose a different manager is considering employees for a different award. There are 100 male employees and 200 female employees, and they decide to give awards to 10 male employees. **Assuming the manager is required to use the award rule above**, how many female employees do they need to give awards to?

   (a) 10

   (b) 20

   (c) 40

   (d) 50

4. **Assuming the manager is required to use the award rule above**, in which of these cases can a manager give more awards to female employees than to male employees?

   (a) When there are more well-qualified female employees than well-qualified male employees (i.e., more women have better performance reviews, higher mid-year net sales, and more years on the job).

   (b) When there are more female employees than male employees.

   (c) When female employees receive higher performance reviews than male employees.

   (d) This cannot happen under the award rule.

5. **Assuming the manager is required to use the award rule above**, is the following statement TRUE OR FALSE: Even if a male employee's qualifications look

similar to a female employee's (in terms of performance reviews, mid-year net sales, and years on the job), he can be treated differently (i.e., only one of the employees gets an award).

6. **Assuming the manager is required to use the award rule above**, is the following statement TRUE OR FALSE: If all female employees are unqualified (i.e., have low performance reviews, low mid-year net sales, and few years on the job), but you give awards to some of them, then awards given to male employees must be made to unqualified male employees.

7. **Assuming the manager is required to use the award rule above**, is the following statement TRUE OR FALSE: Suppose the manager is distributing 10 awards amongst a pool that includes both male and female employees. Even if all male employees are unqualified for an award (i.e., have low performance reviews, low mid-year net sales, and few years on the job), some of them must still receive awards.

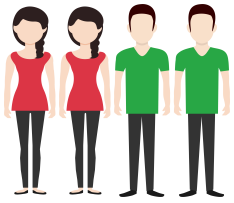
8. **Assuming the manager is required to use the award rule above**, is the following statement TRUE OR FALSE: This award rule always allows the manager to distribute awards exclusively to the most qualified employees (i.e., employees with better performance reviews, high mid-year net sales, and high number of years on the job).

In the two examples above there are 100 employees. Consider a different scenario, with **6 employees– 4 female and 2 male, as illustrated below**. The next three questions each give a potential outcome for all six employees (i.e., which of the 6 employees receive awards). Please indicate which of the outcomes follow **the award rule above**.

9. Alternative scenario 1: 

   Does this distribution of awards obey the **award rule**? Yes

10. Alternative scenario 2:



    Does this distribution of awards obey the **award rule**? No

11. Alternative scenario 3:

    Does this distribution of awards obey the **award rule**? No

12. In your own words, explain the **award rule**. [short answer] (The rule is not shown above this question)

13. To what extent do you agree with the following statement: I am confident I know how to **apply the award rule described above**?

    - Strongly agree
    - Agree

- Neither agree nor disagree
- Disagree
- Strongly Disagree

14. Please select the choice that best describes your experience: When I answered the previous questions...

    (a) I applied the provided award rule only.

    (b) I used my own ideas of what the correct award decision should be rather than the provided award rule.

    (c) I used a combination of the provided award rule and my own ideas of what the correct award decision should be.

15. What is your opinion on the award rule? Please explain why. [short answer]

16. Suppose that you are the manager whose job it is to distribute mid-year awards to employees based on the criteria listed above (i.e., recent performance reviews, mid-year net sales, number of years on the job). How would you ensure that this process is fair? [short answer]

17. Was there anything about this survey that was hard to understand or answer? [short answer]

**Hiring** The hiring manager uses the following hiring rule to send out offers: *The fraction of applicants who receive job offers that are female should equal the fraction of applicants that are female. Similarly, fraction of applicants who receive job offers that are male should equal the fraction of applicants that are male.*

Example 1: If there are 50 female applicants out of the 100 applicants, then 5 out of the 10 offers would be made to female applicants (and the remaining 5 would be made to male applicants).

Example 2: If there are 30 female applicants out of the 100 applicants, then 3 out of the 10 offers would be made to female applicants (and the remaining 7 would be made to male applicants).

In the next section, we will ask you some questions about the information you have just read. Please note that this is not a test of your abilities. We want to measure the quality of the description you read, not your ability to take tests or answer questions.

**Please note that we ask you to apply and use ONLY the above hiring rule when answering the following questions. You will have an opportunity to state your opinions and feelings on the rule later in the survey.**

3. Suppose a different hiring manager is considering applicants for a different job. There are 100 male applicants and 200 female applicants, and they decide to send offers to 10 male applicants. **Assuming the hiring manager is required to use the hiring rule above**, how many female applicants do they need to send offers to?

   (a) 10

   (b) 20

   (c) 40

   (d) 50

4. **Assuming the hiring manager is required to use the hiring rule above**, in which of these cases can a hiring manager make more job offers to female applicants than to male applicants?

(a) When there are more well-qualified female applicants than well-qualified male applicants (i.e., more women have higher interview scores, higher quality recommendation letters, and more years of prior experience in the field).

(b) When there are more female applicants than male applicants.

(c) When female applicants receive better interview scores than male applicants.

(d) This cannot happen under the hiring rule.

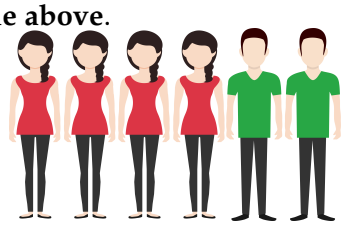
5. **Assuming the hiring manager is required to use the hiring rule above**, is the following statement TRUE OR FALSE: Even if a male applicant's qualifications look similar to a female applicant's (in terms of interview scores, recommendation letters, and years of prior experience in the field), he can be treated differently (i.e., only one of the applicants will receive a job offer).

6. **Assuming the hiring manager is required to use the hiring rule above**, is the following statement TRUE OR FALSE: If all female applicants are unqualified (i.e., have low interview scores, low-quality recommendation letters, and few years of prior experience in the field), but you send job offers to some of them, then any job offers made to male applicants must be made to unqualified male applicants.

7. **Assuming the hiring manager is required to use the hiring rule above**, is the following statement TRUE OR FALSE: Suppose the hiring manager is sending out 10 job offers to a pool that includes male and female applicants. Even if all male applicants are unqualified (i.e., have low interview scores, low-quality recommendation letters, and few years of prior experience in the field), some of them must still receive job offers.

8. **Assuming the hiring manager is required to use the hiring rule above**, is the following statement TRUE OR FALSE: This hiring rule always allows the hiring manager to send offers exclusively to the most qualified applicants (i.e., applicants with high interview scores, high quality recommendation letters, and high number years of prior experience in the field).

In the two examples above there are 100 applicants. Consider a different scenario, with **6 applicants – 4 female and 2 male, as illustrated below**. The next three questions each give a potential outcome for all 6 applicants (i.e., which of the 6 applicants receive job offers). Please indicate which of the outcomes follow **the hiring rule above**.



9. Alternative scenario 1:

   Does this distribution of job offers obey the **hiring rule**? Yes

10. Alternative scenario 2:

    Does this distribution of job offers obey the **hiring rule**? No

11. Alternative scenario 3:

Does this distribution of job offers obey the **hiring rule**? No

12. In your own words, explain the **hiring rule**. [short answer] (The rule is not shown above this question)

13. To what extent do you agree with the following statement: I am confident I know how to **apply the hiring rule described above**?

    - Strongly agree
    - Agree
    - Neither agree nor disagree
    - Disagree
    - Strongly Disagree

14. Please select the choice that best describes your experience: When I answered the previous questions...

    (a) I applied the provided hiring rule only.

    (b) I used my own ideas of what the correct hiring decision should be rather than the provided hiring rule.

    (c) I used a combination of the provided hiring rule and my own ideas of what the correct hiring decision should be.

15. What is your opinion on the hiring rule? Please explain why. [short answer]

16. Suppose that you are the hiring manager whose job it is to send job offers to applicants based on the criteria listed above (i.e., interview scores, quality of

recommendation letters, number of years of prior experience in the field). How would you ensure that this process is fair? [short answer]

17. Was there anything about this survey that was hard to understand or answer? [short answer]

### G.4.2 Study-2: Survey

Each of the surveys are split into four main sections. The first section is the consent form which can be found in Appendix G.5. The second section describes the hiring scenario and asks questions about it (§G.4.2.1). The third section describes the fairness metric, defined as the rule, used (in this case it is demographic parity) and asks specific questions about the metric (§G.4.2.2). Finally the last section asks for demographic information (§G.4.3).

#### G.4.2.1 Scenario description and questions

The following is shown to each participant (note that Step 3 is not shown to participants with the DP definition):

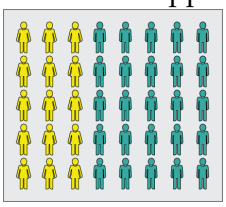
It is very important that you read each question carefully and think about your answers. The success of our research relies on our respondents being thoughtful and taking this task seriously.

☐ I have read the above instructions carefully.

A company, Sales-a-lot, is reviewing their hiring process. They want to hire applicants who are high performing, and they also want to make sure that their hiring process is fair to their applicants, no matter their gender. To do this, Sales-a-lot employs an external firm, Recruit-a-matic, which keeps track of all applicants. This review will take place over one year.

For clarity at each stage of the hiring process we use images to represent the hiring pool.

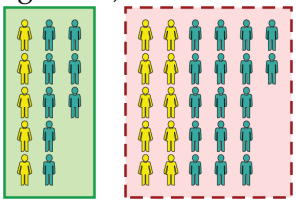
**Step 1: Applicant Pool.** At the beginning of the year, Sales-a-lot reviews all job applicants, and sends job offers to some of them. The initial applicant pool is shown with a gray background. For example, the following image shows an applicant pool with 15 female applicants and 25 male applicants:



**Step 2: Sending Job Offers.** Next, Sales-a-lot sends job offers to some of these applicants, using the following criteria:

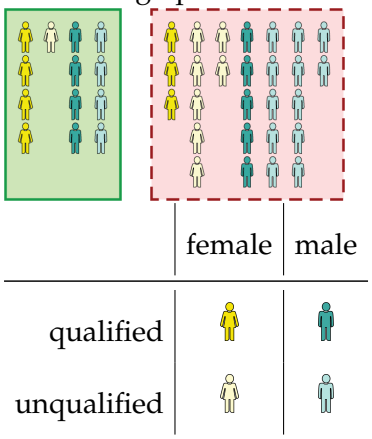
- Interview scores
- Quality of recommendation letters
- Number of years of prior experience in the field

Suppose that Sales-a-lot sends offers to 5 female applicants and 8 male applicants (so 10 female and 17 male applicants didn't receive offers). In the following image, applicants who received a job offer are shown on the left (with a green background) and applicants who didn't receive a job offer are shown on the right, with a red background):



**Step 3: Applicant Evaluation.** For the rest of the year, Recruit-a-matic (the external firm) keeps track of all applicants in the initial pool, whether they received job offers

or not. At the end of the year, Rectruit-a-matic finds out which applicants were high performers, i.e., qualified (shown in dark), and which applicants were low performers, i.e., unqualified (shown in light). For example, the following image shows that most of the high performers received job offers, but some did not.




**Questions**

1. To what extent do you agree with the following statement: a scenario similar to the one described above might occur in real life.

    - Strongly agree
    - Agree
    - Neither agree nor disagree
    - Disagree
    - Strongly disagree

2. How much effort, in hours, should Sales-a-lot put in to make sure these decisions were fair? [short answer - number of hours]

#### G.4.2.2 Rule descriptions and questions

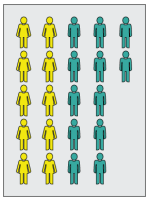
The following sections provide fairness definitions (presented to participants as *rules*) for Demographic Parity, Equal Opportunity (FNR and FPR), and Equalized Odds.

Unless otherwise noted the rule description is shown above each of the questions for reference. Correct answers are noted in red.

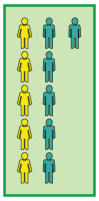
**Demographic Parity.** Recruit-a-matic uses the following rule to determine whether Sales-a-lot's hiring decisions were fair:

*The fraction of male candidates who receive job offers should equal the fraction of female candidates who receive job offers.*

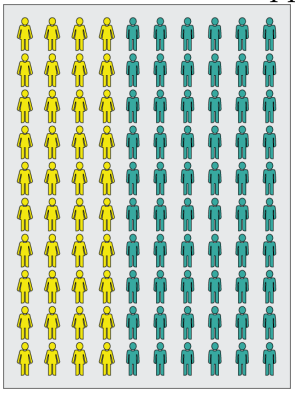
Example 1: Suppose that over the past year, Recruit-a-matic finds that Sales-a-lot received the following applicants (10 female and 12 male).



If Sales-a-lot sent job offers to the following number of applicants (5 female and 6 male), then this would be fair according to the hiring rule (note that there are other possible outcomes that are fair according to the hiring rule).

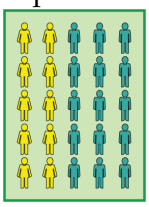


Example 2: Suppose that over the past year, Recruit-a-matic finds that Sales-a-lot reviewed a total of 100 applicants as follows (40 female and 60 male).



If Sales-a-lot sent job offers to the following number of applicants (10 female and

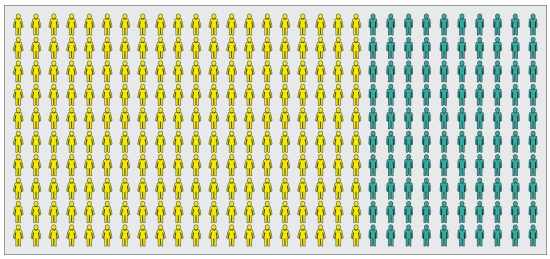
15 male), then this would be fair according to the hiring rule (note that there are other possible outcomes that are fair according to the hiring rule).



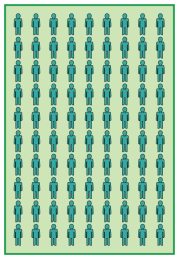
In the next section, we will ask you some questions about the information you have just read. Please note that this is not a test of your abilities. We want to measure the quality of the description you read, not your ability to take tests or answer questions.

**Please note that we ask you to apply and use ONLY the above hiring rule when answering the following questions. You will have an opportunity to state your opinions and feelings on the rule later in the survey.**

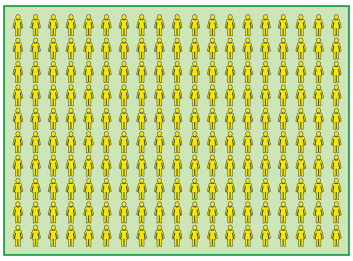
3. Suppose a different company considered applicants for a different job. There were 200 female applicants and 100 male applicants,

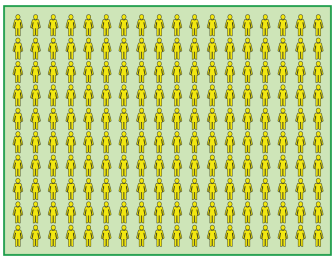


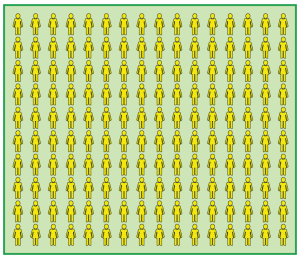
and they did send job offers to 90 male applicants.

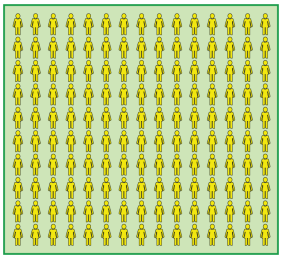


Assuming that Recruit-a-matic reviews their decisions using the hiring rule above, how many female applicants should have received job offers?

(a) 190 

(b) 180 

(c) 160 

(d) 150 

4. Assuming Recruit-a-matic reviews decisions using the hiring rule above, in which of these cases could Sales-a-lot have accepted more qualified female applicants than qualified male applicants?

   (a) When there are more qualified female applicants than qualified male applicants (i.e., more women had low net sales at the end of the year).

   (b) When there are more female applicants than male applicants.

   (c) When female applicants receive worse interview scores than male applicants.

   (d) This cannot happen under the hiring rule.

5. Consider one male applicant and one female applicant, both of whom are similarly qualified for the job (they achieve about the same net sales at the end of their first year). Is the following statement TRUE OR FALSE: The hiring rule above allows Sales-a-lot to make a job offer to one of these applicants and not the other.

6. Consider a situation where all female applicants were unqualified (they all achieve low net sales at the end of their first year), but some of them received job offers. Is the following statement TRUE OR FALSE: The hiring rule above requires that some job offers made to male applicants must have been made to unqualified male applicants.

7. Suppose Sales-a-lot received 100 male and 100 female applicants, and eventually made 10 job offers. Is the following statement TRUE OR FALSE: The hiring rule above requires that even if all male applicants were unqualified (they all achieve low net sales at the end of their first year), some of the unqualified males must have received job offers.

8. Is the following statement TRUE OR FALSE: The hiring rule above always allows Sales-a-lot to send job offers only to the most qualified applicants (those who achieve high net sales at the end of their first year).

Consider a different scenario than the two examples above, with 6 applicants – 4 female and 2 male, as illustrated below. The next three questions each give a different potential outcome for all 6 applicants (i.e., which of the 6 applicants do receive job offers). Please indicate which of the outcomes follow the hiring rule above.

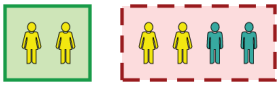
9. Sales-a-lot makes the following hiring decisions.

Do these decisions obey the hiring rule? Yes

10. Sales-a-lot makes the following hiring decisions.

Do these decisions obey the hiring rule? No

11. Sales-a-lot makes the following hiring decisions.



Do these decisions obey the hiring rule? No

12. In your own words, explain the hiring rule. [short answer] [The rule is not shown above this question]

13. To what extent do you agree with the following statement: I am confident I know how to apply the hiring rule described above?

    - Strongly agree
    - Agree
    - Neither agree nor disagree
    - Disagree
    - Strongly Disagree

14. Please select the choice that best describes your experience: When I answered the previous questions...

    (a) I applied the provided hiring rule only.

(b) I used a combination of the provided hiring rule and my own ideas of what the correct hiring rule should be.

(c) I used only my own ideas of what the correct hiring decision should be rather than the provided hiring rule.

15. To what extent do you agree with the following statement: I like the hiring rule?

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly Disagree

16. To what extent do you agree with the following statement: I agree with the hiring rule?

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly Disagree

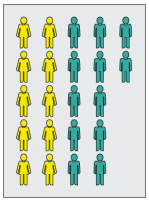
17. Please explain your opinion on the hiring rule. [short answer]

18. Was there anything about this survey that was hard to understand or answer? [short answer]

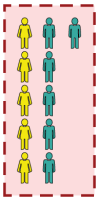
**Equal Opportunity - FNR.** Recruit-a-matic uses the following rule to determine whether Sales-a-lot's hiring decisions were fair:

*The fraction of qualified male candidates who do not receive job offers should equal the fraction of qualified female candidates who do not receive job offers.*

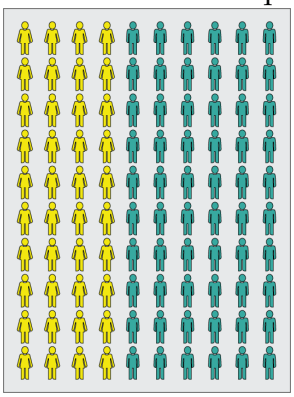
Example 1: Suppose that over the past year, Recruit-a-matic finds that Sales-a-lot received the following qualified applicants (10 female and 12 male).



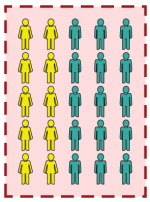
If Sales-a-lot did not send job offers to the following number of qualified applicants (5 female and 6 male), then this would be fair according to the hiring rule (note that there are other possible outcomes that are fair according to the hiring rule).



Example 2: Suppose that over the past year, Recruit-a-matic finds that Sales-a-lot reviewed a total of 100 qualified applicants as follows (40 female and 60 male).



If Sales-a-lot did not send job offers to the following number of qualified applicants (10 female and 15 male), then this would be fair according to the hiring rule (note that there are other possible outcomes that are fair according to the hiring rule).

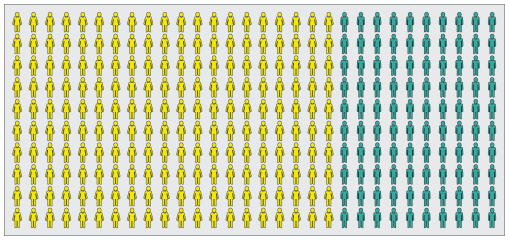


Note that in the above examples the remaining qualified applicants received job offers, but are not displayed here.

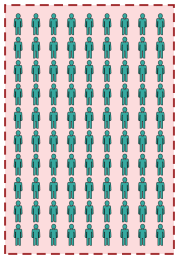
In the next section, we will ask you some questions about the information you have just read. Please note that this is not a test of your abilities. We want to measure the quality of the description you read, not your ability to take tests or answer questions.

**Please note that we ask you to apply and use ONLY the above hiring rule when answering the following questions. You will have an opportunity to state your opinions and feelings on the rule later in the survey.**

3. Suppose a different company considered applicants for a different job. There were 200 qualified female applicants and 100 qualified male applicants,

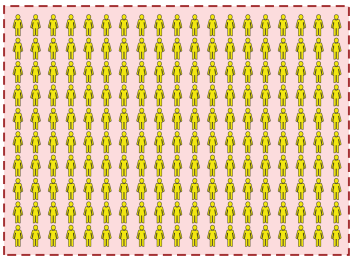


and they did not send job offers to 90 qualified male applicants.

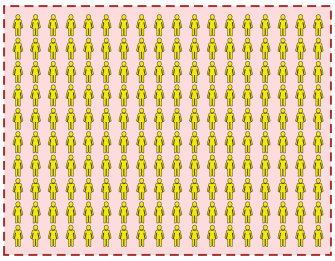


Assuming that Recruit-a-matic reviews their decisions using the hiring rule above, how many qualified female applicants should not have received job offers?

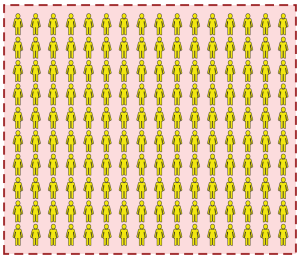
(a) 190



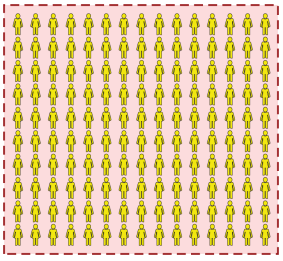
(b) 180



(c) 160



(d) 150



4. Assuming Recruit-a-matic reviews decisions using the hiring rule above, in which of these cases could Sales-a-lot have rejected more qualified female applicants than qualified male applicants?

   (a) When there are more qualified female applicants than qualified male applicants (i.e., more women had low net sales at the end of the year).

   (b) When there are more female applicants than male applicants.

   (c) When female applicants receive worse interview scores than male applicants.

   (d) This cannot happen under the hiring rule.

5. Consider one male applicant and one female applicant, both of whom are similarly qualified for the job (they achieve about the same net sales at the end of their first year). Is the following statement TRUE OR FALSE: The hiring rule above allows Sales-a-lot to make a job offer to one of these applicants and not the other.

6. Consider a situation where all female applicants were unqualified (they all achieve low net sales at the end of their first year), but some of them received job offers. Is the following statement TRUE OR FALSE: The hiring rule above requires that some job offers made to male applicants must have been made to unqualified male applicants.

7. Suppose Sales-a-lot received 100 male and 100 female applicants, and eventually made 10 job offers. Is the following statement TRUE OR FALSE: The hiring rule above requires that even if all male applicants were unqualified (they all achieve low net sales at the end of their first year), some of the unqualified males must have received job offers.

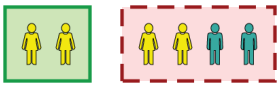
8. Is the following statement TRUE OR FALSE: The hiring rule above always allows Sales-a-lot to send job offers only to the most qualified applicants (those who achieve high net sales at the end of their first year).

Consider a different scenario than the two examples above, with 6 qualified applicants – 4 female and 2 male, as illustrated below. The next three questions each give a different potential outcome for all 6 qualified applicants (i.e., which of the 6 applicants do not receive job offers). Please indicate which of the outcomes follow the hiring rule above.

9. Sales-a-lot makes the following hiring decisions.

Do these decisions obey the hiring rule? Yes

10. Sales-a-lot makes the following hiring decisions.



Do these decisions obey the hiring rule? No

11. Sales-a-lot makes the following hiring decisions.

Do these decisions obey the hiring rule? No

12. In your own words, explain the hiring rule. [short answer] [The rule is not shown above this question]

13. To what extent do you agree with the following statement: I am confident I know how to apply the hiring rule described above?

    - Strongly agree
    - Agree
    - Neither agree nor disagree
    - Disagree
    - Strongly Disagree

14. Please select the choice that best describes your experience: When I answered the previous questions...

    (a) I applied the provided hiring rule only.

(b) I used a combination of the provided hiring rule and my own ideas of what the correct hiring rule should be.

(c) I used only my own ideas of what the correct hiring decision should be rather than the provided hiring rule.

15. To what extent do you agree with the following statement: I like the hiring rule?

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly Disagree

16. To what extent do you agree with the following statement: I agree with the hiring rule?

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly Disagree

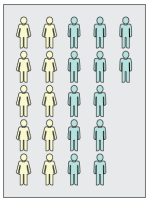
17. Please explain your opinion on the hiring rule. [short answer]

18. Was there anything about this survey that was hard to understand or answer? [short answer]

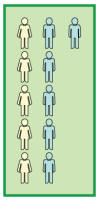
**Equal Opportunity - FPR.** Recruit-a-matic uses the following rule to determine whether Sales-a-lot's hiring decisions were fair:

*The fraction of unqualified male candidates who receive job offers should equal the fraction of unqualified female candidates who receive job offers.*

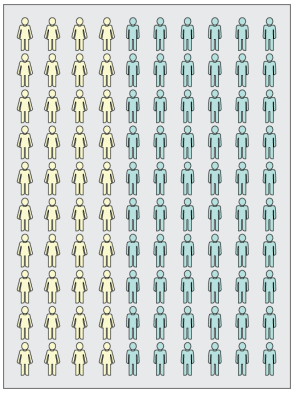
Example 1: Suppose that over the past year, Recruit-a-matic finds that Sales-a-lot received the following unqualified applicants (10 female and 12 male).



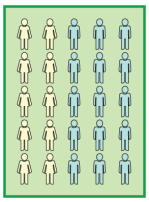
If Sales-a-lot sent job offers to the following number of unqualified applicants (5 female and 6 male), then this would be fair according to the hiring rule (note that there are other possible outcomes that are fair according to the hiring rule).



Example 2: Suppose that over the past year, Recruit-a-matic finds that Sales-a-lot reviewed a total of 100 unqualified applicants as follows (40 female and 60 male).



If Sales-a-lot sent job offers to the following number of unqualified applicants (10 female and 15 male), then this would be fair according to the hiring rule (note that there are other possible outcomes that are fair according to the hiring rule).

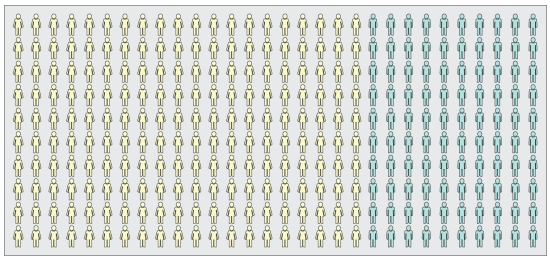


Note that in the above examples the remaining unqualified applicants did not receive job offers, but are not displayed here.

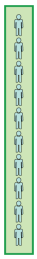
In the next section, we will ask you some questions about the information you have just read. Please note that this is not a test of your abilities. We want to measure the quality of the description you read, not your ability to take tests or answer questions.

**Please note that we ask you to apply and use ONLY the above hiring rule when answering the following questions. You will have an opportunity to state your opinions and feelings on the rule later in the survey.**

3. Suppose a different company considered applicants for a different job. There were 200 unqualified female applicants and 100 unqualified male applicants,

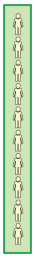


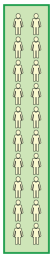
and they did send job offers to 10 unqualified male applicants.

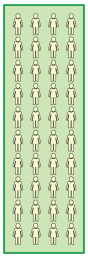


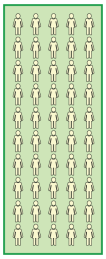
Assuming that Recruit-a-matic reviews their decisions using the hiring rule

above, how many unqualified female applicants should have received job offers?

(a) 10 

(b) 20 

(c) 40 

(d) 50 

4. Assuming Recruit-a-matic reviews decisions using the hiring rule above, in which of these cases could Sales-a-lot have accepted more unqualified female applicants than unqualified male applicants?

   (a) When there are more unqualified female applicants than unqualified male applicants (i.e., more women had low net sales at the end of the year).

   (b) When there are more female applicants than male applicants.

   (c) When female applicants receive worse interview scores than male applicants.

(d) This cannot happen under the hiring rule.

5. Consider one male applicant and one female applicant, both of whom are similarly qualified for the job (they achieve about the same net sales at the end of their first year). Is the following statement TRUE OR FALSE: The hiring rule above allows Sales-a-lot to make a job offer to one of these applicants and not the other.

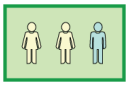
6. Consider a situation where all female applicants were unqualified (they all achieve low net sales at the end of their first year), but some of them received job offers. Is the following statement TRUE OR FALSE: The hiring rule above requires that some job offers made to male applicants must have been made to unqualified male applicants.

7. Suppose Sales-a-lot received 100 male and 100 female applicants, and eventually made 10 job offers. Is the following statement TRUE OR FALSE: The hiring rule above requires that even if all male applicants were unqualified (they all achieve low net sales at the end of their first year), some of the unqualified males must have received job offers.

8. Is the following statement TRUE OR FALSE: The hiring rule above always allows Sales-a-lot to send job offers only to the most qualified applicants (those who achieve high net sales at the end of their first year).

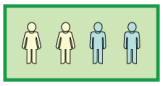
Consider a different scenario than the two examples above, with 6 unqualified applicants – 4 female and 2 male, as illustrated below. The next three questions each give a different potential outcome for all 6 applicants (i.e., which of the 6 applicants receive job offers). Please indicate which of the outcomes follow the hiring rule above.

9. Sales-a-lot makes the following hiring decisions.



Do these decisions obey the hiring rule? Yes

10. Sales-a-lot makes the following hiring decisions.



Do these decisions obey the hiring rule? No

11. Sales-a-lot makes the following hiring decisions.

Do these decisions obey the hiring rule? No

12. In your own words, explain the hiring rule. [short answer] [The rule is not shown above this question]

13. To what extent do you agree with the following statement: I am confident I know how to apply the hiring rule described above?

    - Strongly agree
    - Agree
    - Neither agree nor disagree
    - Disagree
    - Strongly Disagree

14. Please select the choice that best describes your experience: When I answered the previous questions...

(a) I applied the provided hiring rule only.

(b) I used a combination of the provided hiring rule and my own ideas of what the correct hiring rule should be.

(c) I used only my own ideas of what the correct hiring decision should be rather than the provided hiring rule.

15. To what extent do you agree with the following statement: I like the hiring rule?

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly Disagree

16. To what extent do you agree with the following statement: I agree with the hiring rule?

- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly Disagree

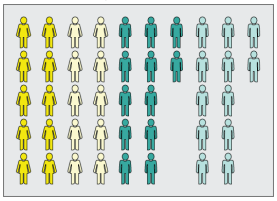
17. Please explain your opinion on the hiring rule. [short answer]

18. Was there anything about this survey that was hard to understand or answer? [short answer]

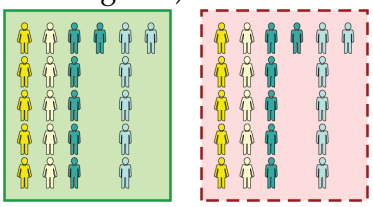
**Equalized Odds.** Recruit-a-matic uses the following rule to determine whether Sales-a-lot's hiring decisions were fair:

*The fraction of qualified male candidates who do not receive job offers should equal the fraction of qualified female candidates who do not receive job offers. Similarly, the fraction of unqualified male candidates who receive job offers should equal the fraction of unqualified female candidates who receive job offers.*

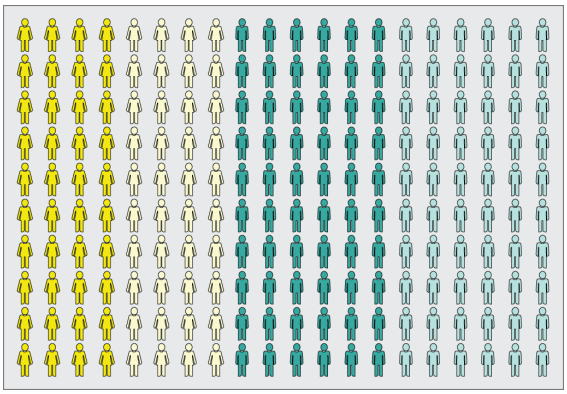
Example 1: Suppose that over the past year, Recruit-a-matic finds that Sales-a-lot received the following qualified applicants (10 female and 12 male) and unqualified applicants (10 female and 12 male).



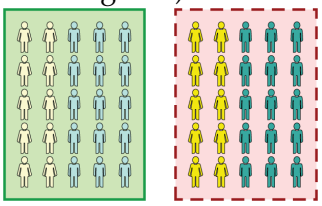
If Sales-a-lot did send offers to the following number of unqualified applicants (left, 5 female and 6 male), and did not send job offers to the following number of qualified applicants (right, 5 female and 6 male), then this would be fair according to the hiring rule (note that there are other possible outcomes that are fair according to the hiring rule).



Example 2: Suppose that over the past year, Recruit-a-lot finds that Sales-a-lot reviewed a total of 100 qualified applicants (40 female and 60 male) and 100 unqualified applicants (40 female and 60 male).



If Sales-a-lot did send offers to the following number of unqualified applicants (left, 10 female and 15 male), and did not send job offers to the following number of qualified applicants (right, 10 female and 15 male), then this would be fair according to the hiring rule (note that there are other possible outcomes that are fair according to the hiring rule).

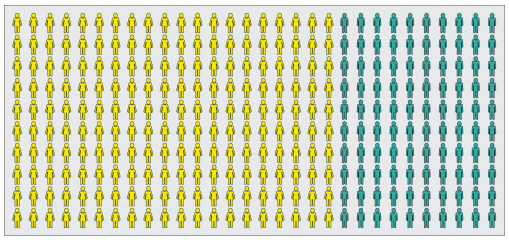


Note that in the above examples the remaining unqualified applicants did not receive job offers, but are not displayed here. Similarly, the remaining qualified applicants received job offers, but are not displayed here.

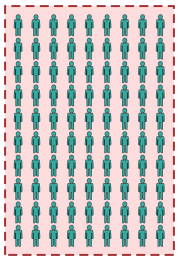
In the next section, we will ask you some questions about the information you have just read. Please note that this is not a test of your abilities. We want to measure the quality of the description you read, not your ability to take tests or answer questions.

**Please note that we ask you to apply and use ONLY the above hiring rule when answering the following questions. You will have an opportunity to state your opinions and feelings on the rule later in the survey.**

3. Suppose a different company considered applicants for a different job. There were 200 qualified female applicants and 100 qualified male applicants,

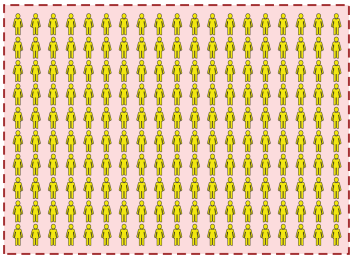


and they did not send job offers to 90 qualified male applicants.

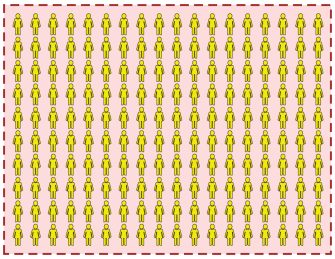


Assuming that Recruit-a-matic reviews their decisions using the hiring rule above, how many qualified female applicants should not have received job offers?

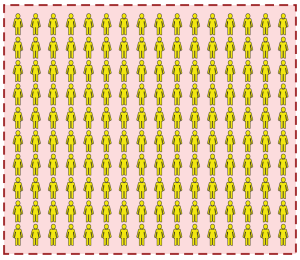
(a) 190

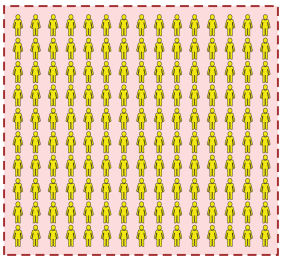


(b) 180



(c) 160



(d) 150 

4. Assuming Recruit-a-matic reviews decisions using the hiring rule above, in which of these cases could Sales-a-lot have accepted more unqualified female applicants than unqualified male applicants?

   (a) When there are more unqualified female applicants than unqualified male applicants (i.e., more women had low net sales at the end of the year).

   (b) When there are more female applicants than male applicants.

   (c) When female applicants receive worse interview scores than male applicants.

   (d) This cannot happen under the hiring rule.

5. Consider one male applicant and one female applicant, both of whom are similarly qualified for the job (they achieve about the same net sales at the end of their first year). Is the following statement TRUE OR FALSE: The hiring rule above allows Sales-a-lot to make a job offer to one of these applicants and not the other.

6. Consider a situation where all female applicants were unqualified (they all achieve low net sales at the end of their first year), but some of them received job offers. Is the following statement TRUE OR FALSE: The hiring rule above requires that some job offers made to male applicants must have been made to unqualified male applicants.

7. Suppose Sales-a-lot received 100 male and 100 female applicants, and eventually made 10 job offers. Is the following statement TRUE OR FALSE: The hiring rule above requires that even if all male applicants were unqualified (they all achieve low net sales at the end of their first year), some of the unqualified males must have received job offers.

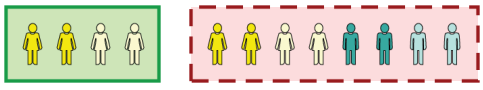
8. Is the following statement TRUE OR FALSE: The hiring rule above always allows Sales-a-lot to send job offers only to the most qualified applicants (those who achieve high net sales at the end of their first year).

Consider a different scenario than the two examples above, with 6 qualified applicants – 4 female and 2 male; and 6 unqualified applicants – 4 female and 2 male. The next three questions each give a different potential outcome for the applicants (i.e., which of the applicants did or did not receive job offers). Please indicate which of the outcomes follow the hiring rule above.

9. Sales-a-lot makes the following hiring decisions.

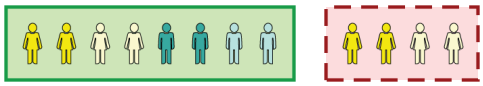
Do these decisions obey the hiring rule? Yes

10. Sales-a-lot makes the following hiring decisions.



Do these decisions obey the hiring rule? No

11. Sales-a-lot makes the following hiring decisions.



Do these decisions obey the hiring rule? No

12. In your own words, explain the hiring rule. [short answer] [The rule is not shown above this question]

13. To what extent do you agree with the following statement: I am confident I know how to apply the hiring rule described above?

    - Strongly agree
    - Agree
    - Neither agree nor disagree
    - Disagree
    - Strongly Disagree

14. Please select the choice that best describes your experience: When I answered the previous questions...

    (a) I applied the provided hiring rule only.

    (b) I used a combination of the provided hiring rule and my own ideas of what the correct hiring rule should be.

    (c) I used only my own ideas of what the correct hiring decision should be rather than the provided hiring rule.

15. To what extent do you agree with the following statement: I like the hiring rule?

    - Strongly agree
    - Agree
    - Neither agree nor disagree
    - Disagree
    - Strongly Disagree

16. To what extent do you agree with the following statement: I agree with the hiring rule?

    - Strongly agree
    - Agree
    - Neither agree nor disagree
    - Disagree
    - Strongly Disagree

17. Please explain your opinion on the hiring rule. [short answer]

18. Was there anything about this survey that was hard to understand or answer? [short answer]

### G.4.3 Demographic Information

1. Please specify the gender with which you most closely identify:

    - Male
    - Female
    - Other
    - Prefer not to answer

2. Please specify your year of birth

3. Please specify your ethnicity (you may select more than one):

    - White
    - Hispanic or Latinx
    - Black or African American

- American Indian or Alaska Native
- Asian, Native Hawaiian, or Pacific Islander
- Other

4. Please specify the highest degree or level of school you have completed:

   - Some high school credit, no diploma or equivalent
   - High school graduate, diploma or the equivalent (for example: GED)
   - Some college credit, no degree
   - Trade/technical/vocational training
   - Associate's degree
   - Bachelor's degree
   - Master's degree
   - Professional or doctoral degree (JD, MD, PhD)

5. How much experience do you have in each of the following areas? (1 - no experience, 2 - limited experience, 3 - significant experience, 4 - expert)

   (a) Human resources (making hiring decisions)

   (b) Management (of employees)

   (c) Education (teaching)

   (d) IT infrastructure/systems administration

   (e) Computer science/programming

   (f) Machine learning/data science

**We will maintain privacy of the information you have provided here. Your information will only be used for data analysis purposes.**

## G.5 Consent

### G.5.1 Online Survey Consent Form

#### G.5.1.1 Project Title

Fairness Evaluation and Comprehension

#### G.5.1.2 Purpose of the Study

This research is being conducted by Michelle Mazurek at the University of Maryland, College Park. We are inviting you to participate in this research project because you are above 18. The purpose of this research project is to understand lay comprehension of different fairness metrics.

#### G.5.1.3 Procedures

The procedures will start with reading a brief description of a decision-making scenario. You will then be asked to answer some comprehension questions about the scenario. The questions will look like the following: What are the pros and cons of the notion of fairness described above?

Finally, you will be asked some demographics questions. The entire survey will take approximately 20 minutes or less.

#### G.5.1.4 Potential Risks and Discomforts

There are several questions to answer over the course of this study, so you may find yourself growing tired towards the end. Outside of this, there are minimal risks to participating in this research study. All data collected in this study will be maintained securely (see Confidentiality section) and will be deleted at the conclusion of the study.

However, if at any time you feel that you wish to terminate your participation for any reason, you are permitted to do so.

#### G.5.1.5 Potential Benefits

There are no direct benefits from participating in this research. We hope that, in the future, other people might benefit from this study through improved understanding of fairness metrics and their applications.

#### G.5.1.6 Confidentiality

Any potential loss of confidentiality will be minimized by storing all data (including information such as MTurk IDs and demographics) will be stored securely (a) in a password-protected computer located at the University of Maryland, College Park or (b) using a trusted third party (Qualtrics). Personally identifiable information that is collected (MTurk IDs, IP addresses, cookies) will be deleted upon study completion. All other data gathered will be stored for three years post study completion, after which it will be erased. The only persons that will have access to the data are the Principle Investigator and the Co-Investigators.

If we write a report or article about this research project, your identity will be protected to the maximum extent possible. Your information may be shared with representatives of the University of Maryland, College Park or governmental authorities if you or someone else is in danger or if we are required to do so by law.

#### G.5.1.7 Compensation

You will receive $3. You will be responsible for any taxes assessed on the compensation.

If you will earn $100 or more as a research participant in this study, you must provide your name, address and SSN to receive compensation.

If you do not earn over $100 only your name and address will be collected to receive compensation.

#### G.5.1.8 Right to Withdraw and Questions

Your participation in this research is completely voluntary. You may choose not to take part at all. If you decide to participate in this research, you may stop participating at any time. If you decide not to participate in this study or if you stop participating at any time, you will not be penalized or lose any benefits to which you otherwise qualify.

If you decide to stop taking part in the study, if you have questions, concerns, or complaints, or if you need to report an injury related to the research, please contact the investigator:

Michelle Mazurek

5236 Iribe Center,

University of Maryland, College Park 20742

mmazurek@cs.umd.edu

(301) 405-6463

#### G.5.1.9 Participant Rights

If you have questions about your rights as a research participant or wish to report a research-related injury, please contact:

University of Maryland College Park

Institutional Review Board Office

1204 Marie Mount Hall

College Park, Maryland, 20742

E-mail: irb@umd.edu

Telephone: 301-405-0678

For more information regarding participant rights, please visit:

https://research.umd.edu/irb-research-participants

This research has been reviewed according to the University of Maryland, College Park IRB procedures for research involving human subjects.

#### G.5.1.10 Statement of Consent

By agreeing below you indicate that you are at least 18 years of age; you have read this consent form or have had it read to you; your questions have been answered to your satisfaction and you voluntarily agree to participate in this research study. Please ensure you have made a copy of the above consent form for your records.

Pease ensure you have made a copy of the above consent form for your records. A copy of this consent form can be found here [link to digital copy].

☐ I am age 18 or older

☐ I have read this consent form

☐ I voluntarily agree to participate in this research study

### G.5.2 Cognitive Interview Consent Form

#### G.5.2.1 Project Title

Fairness Cognitive Interview

#### G.5.2.2 Purpose of the Study

This research is being conducted by Michelle Mazurek at the University of Maryland, College Park. We are inviting you to participate in this research project because you are above the age of 18, and fluent in English. The purpose of this research project is to understand lay comprehension of different fairness metrics.

#### G.5.2.3 Procedures

The procedure involves completing an interview. The full procedure will be approximately 1 hour in duration.

During the interview you will be audio recorded, if you agree to be recorded. You will be asked to first read a brief description of a decision-making scenario. You will then be asked to fill out a survey about the scenario. While answering questions you will be asked verbal questions related to how you reached your answer in the survey.

Sample survey question: Is the following statement true or false? This hiring rule allows the hiring manager to send offers exclusively to the most qualified applicants.

Sample interview question: How did you reach your answer to that survey question?

#### G.5.2.4 Potential Risks and Discomforts

There are several questions to answer over the course of this study, so you may find yourself growing tired towards the end. Outside of this, there are minimal risks to participating in this research study. All data collected in this study will be maintained securely (see Confidentiality section) and will be deleted at the conclusion of the study.

However, if at any time you feel that you wish to terminate your participation for any reason, you are permitted to do so.

#### G.5.2.5 Potential Benefits

There are no direct benefits from participating in this research. We hope that, in the future, other people might benefit from this study through improved understanding of fairness metrics and their applications.

#### G.5.2.6 Confidentiality

Any potential loss of confidentiality will be minimized by storing all data (including information such as demographics) securely (a) in a password protected computer located at the University of Maryland, College Park or (b) using a trusted third party (Qualtrics). Personally identifiable information that is collected will be deleted upon study completion. All other data gathered will be stored for three years post study completion, after which it will be erased. The only persons that will have access to the data are the principle Investigator and the Co-Investigators.

If we write a report or article about this research project, your identity will be protected to the maximum extent possible. Your information may be shared with representatives of the University of Maryland, College Park or governmental authorities if you or someone else is in danger or if we are required to do so by law.

#### G.5.2.7 Compensation

You will receive $30. You will be responsible for any taxes assessed on the compensation.

If you will earn $100 or more as a research participant in this study, you must provide your name, address and SSN to receive compensation.

If you do not earn over $100 only your name and address will be collected to receive compensation.

#### G.5.2.8 Right to Withdraw and Questions

Your participation in this research is completely voluntary. You may choose not to take part at all. If you decide to participate in this research, you may stop participating at any time. If you decide not to participate in this study or if you stop participating at any time, you will not be penalized or lose any benefits to which you otherwise qualify.

If you decide to stop taking part in the study, if you have questions, concerns, or complaints, or if you need to report an injury related to the research, please contact the investigator:

Michelle Mazurek

5236 Iribe Center,

University of Maryland, College Park 20742

mmazurek@cs.umd.edu

(301) 405-6463

#### G.5.2.9 Participant Rights

If you have questions about your rights as a research participant or wish to report a research-related injury, please contact:

University of Maryland College Park

Institutional Review Board Office

1204 Marie Mount Hall

College Park, Maryland, 20742

E-mail: irb@umd.edu

Telephone: 301-405-0678

For more information regarding participant rights, please visit:

https://research.umd.edu/irb-research-participants

This research has been reviewed according to the University of Maryland, College Park IRB procedures for research involving human subjects.

#### G.5.2.10 Statement of Consent

Your signature indicates that you are at least 18 years of age; you have read this consent form or have had it read to you; your questions have been answered to your satisfaction and you voluntarily agree to participate in this research study. You will receive a copy of this signed consent form.

Please initial all that apply (you may choose any number of these statements):

☐ I agree to be audio recorded

☐ I agree to allow researchers to use my audio recording in research publications and presentations.

☐ I do not agree to be audio recorded

If you agree to participate, please sign your name below.

# Bibliography

[1] The Health Insurance Portability and Accountability Act of 1996 (HIPAA). Online at http://www.cms.hhs.gov/hipaa/, 1996. URL http://www.cms.hhs.gov/hipaa/.

[2] California Consumer Privacy Act of 2018 (CCPA). Online at https://leginfo.legislature.ca.gov/faces/codes_displayText.xhtml?division=3.&part=4.&lawCode=CIV&title=1.81.5, 2018. URL https://leginfo.legislature.ca.gov/faces/codes_displayText.xhtml?division=3.&part=4.&lawCode=CIV&title=1.81.5.

[3] Rabeeh Ayaz Abbasi, Onaiza Maqbool, Mubashar Mushtaq, Naif R Aljohani, Ali Daud, Jalal S Alowibdi, and Basit Shahzad. Saving lives using social media: Analysis of the role of twitter for personal blood donation requests and dissemination. *Telematics and Informatics*, 35(4):892–912, 2018.

[4] Gregory D. Abowd, Anind K. Dey, Peter J. Brown, Nigel Davies, Mark Smith, and Pete Steggles. Towards a better understanding of context and context-awareness. In *Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing*, HUC '99, page 304–307, Berlin, Heidelberg, 1999. Springer-Verlag. ISBN 3540665501.

[5] 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.

[6] David J. Abraham, Avrim Blum, and Tuomas Sandholm. Clearing algorithms for barter exchange markets: Enabling nationwide kidney exchanges. In *Proceedings of the 8th ACM Conference on Electronic Commerce*, EC '07, page 295–304, New York, NY, USA, 2007. Association for Computing Machinery. ISBN 9781595936530. doi: 10.1145/1250910.1250954. URL https://doi.org/10.1145/1250910.1250954.

[7] 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.

[8] Nikhil Agarwal, Itai Ashlagi, Michael A Rees, Paulo J Somaini, and Daniel C Waldinger. Equilibrium allocations under alternative waitlist designs: Evidence from deceased donor kidneys. Working Paper 25607, National Bureau of Economic Research, February 2019. URL http://www.nber.org/papers/w25607.

[9] H Akaike. A new look at the statistical model identification. In *IEEE Transactions on Automatic Control*, volume 19, page 716–723, 1974. doi: 10.1109/TAC.1974.1100705.

[10] Turki Alanzi and Batool Alsaeed. Use of social media in the blood donation process in saudi arabia. *Journal of Blood Medicine*, 10:417, 2019.

[11] Colin Allen, Iva Smit, and Wendell Wallach. Artificial morality: Top-down, bottom-up, and hybrid approaches. *Ethics and Information Technology*, 7(3):149–155, 2005. doi: 10.1007/s10676-006-0004-4. URL https://doi.org/10.1007/

s10676-006-0004-4.

[12] David Alvarez-Melis and Tommi S Jaakkola. Towards robust interpretability with self-explaining neural networks. *arXiv preprint arXiv:1806.07538*, 2018.

[13] 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.

[14] O. Amir and J. Levav. Choice construction versus preference construction: The instability of preferences learned in context. *Journal of Marketing Research*, 45 (2):145–158, 2008.

[15] Ross Anderson, Itai Ashlagi, David Gamarnik, and Yash Kanoria. A dynamic model of barter exchange. In *Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1925–1933, 2015.

[16] 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.

[17] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica, May*, 23:2016, 2016.

[18] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias, 2016. URL https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing.

[19] Richard P Anstee. A polynomial algorithm for b-matchings: an alternative approach. *Information Processing Letters*, 24(3):153–157, 1987.

[20] Anna Markella Antoniadi, Yuhan Du, Yasmine Guendouz, Lan Wei, Claudia Mazo, Brett A. Becker, and Catherine Mooney. Current challenges and future opportunities for xai in machine learning-based clinical decision support systems: A systematic review. *Applied Sciences*, 11(11), 2021. ISSN 2076-3417. doi: 10.3390/app11115088. URL https://www.mdpi.com/2076-3417/11/11/5088.

[21] Kenneth J Arrow. *Social choice and individual values*, volume 12. Yale university press, 2012.

[22] Vijay Arya, Rachel K. E. Bellamy, Pin-Yu Chen, Amit Dhurandhar, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Q. Vera Liao, Ronny Luss, Aleksandra Mojsilović, Sami Mourad, Pablo Pedemonte, Ramya Raghavendra, John Richards, Prasanna Sattigeri, Karthikeyan Shanmugam, Moninder Singh, Kush R. Varshney, Dennis Wei, and Yunfeng Zhang. One explanation does not fit all: A toolkit and taxonomy of ai explainability techniques, 2019.

[23] Itai Ashlagi and Alvin E. Roth. Individual rationality and participation in large scale, multi-hospital kidney exchange. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 321–322, 2011.

[24] Itai Ashlagi and Alvin E Roth. Free riding and participation in large scale, multi-hospital kidney exchange. *Theoretical Economics*, 9(3):817–863, 2014.

[25] Itai Ashlagi, David Gamarnik, Michael Rees, and Alvin E. Roth. The need for (long) chains in kidney exchange. NBER Working Paper No. 18202, July 2012.

[26] Itai Ashlagi, Patrick Jaillet, and Vahideh H. Manshadi. Kidney exchange in dynamic sparse heterogenous pools. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 25–26, 2013.

[27] Itai Ashlagi, Felix Fischer, Ian A. Kash, and Ariel D. Procaccia. Mix and match: A strategyproof mechanism for multi-hospital kidney exchange. *Games and Economic Behavior*, 91:284–296, 2015. ISSN 0899-8256. doi: https://doi.org/10.1016/j.geb.2013.05.008. URL https://www.sciencedirect.com/science/article/pii/S089982561300081X.

[28] Olivier Aubert, Peter P. Reese, Benoit Audry, Yassine Bouatou, Marc Raynaud, Denis Viglietti, Christophe Legendre, Denis Glotz, Jean-Phillipe Empana, Xavier Jouven, Carmen Lefaucheur, Christian Jacquelinet, and Alexandre Loupy. Disparities in Acceptance of Deceased Donor Kidneys Between the United States and France and Estimated Effects of Increased US Acceptance. *JAMA Internal Medicine*, 179(10):1365–1374, 10 2019. ISSN 2168-6106. doi: 10.1001/jamainternmed.2019.2322. URL https://doi.org/10.1001/jamainternmed.2019.2322.

[29] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[30] Pranjal Awasthi and Tuomas Sandholm. Online stochastic optimization in the large: Application to kidney exchange. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 405–411, 2009.

[31] David A. Axelrod, Mark A. Schnitzler, Huiling Xiao, William Irish, Elizabeth Tuttle-Newhall, Su-Hsin Chang, Bertram L. Kasiske, Tarek Alhamad, and Krista L. Lentine. An economic assessment of contemporary kidney transplant practice. *American Journal of Transplantation*, 18(5):1168–1176, 2018. doi: https://doi.org/10.1111/ajt.14702. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/ajt.14702.

[32] Haris Aziz. Strategyproof multi-item exchange under single-minded dichotomous preferences. *Autonomous Agents and Multi-Agent Systems*, 34(1): 3, 2019. doi: 10.1007/s10458-019-09426-w. URL https://doi.org/10.1007/s10458-019-09426-w.

[33] Haris Aziz, Aris Filos-Ratsikas, Jiashu Chen, Simon Mackenzie, and Nicholas Mattei. Egalitarianism of random assignment mechanisms. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2016.

[34] Egon Balas. The prize collecting traveling salesman problem. *Networks*, 19(6): 621–636, 1989.

[35] Nikhil Bansal, Anupam Gupta, Jian Li, Julián Mestre, Viswanath Nagarajan, and Atri Rudra. When LP is the cure for your matching woes: Improved bounds for stochastic matchings. *Algorithmica*, 63(4):733–762, 2012.

[36] Salvador Barberà, Walter Bossert, and Prasanta K. Pattanaik. *Ranking Sets of Objects*, pages 893–977. Springer US, Boston, MA, 2004. ISBN 978-1-4020-7964-1. doi: 10.1007/978-1-4020-7964-1_4. URL https://doi.org/10.1007/978-1-4020-7964-1_4.

[37] Solon Barocas and Andrew D Selbst. Big data's disparate impact. *Calif. L. Rev.*, 104:671, 2016.

[38] Valentin Bartier, Bart Smeulders, Yves Crama, and Frits CR Spieksma. Recourse in kidney exchange programs, 2019. Working paper.

[39] Carmen Batanero, Egan J Chernoff, Joachim Engel, Hollylynne S Lee, and Ernesto Sánchez. Research on teaching and learning probability. In *Research on teaching and learning probability*, pages 1–33. Springer, Cham, 2016.

[40] Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, A Mojsilović, et al. Ai fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, 63(4/5): 4–1, 2019.

[41] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust Optimization*. Princeton University Press, 2009.

[42] Stan Benjamens, Pranavsingh Dhunnoo, and Bertalan Meskó. The state of artificial intelligence-based fda-approved medical devices and algorithms: an online database. *npj Digital Medicine*, 3(1):118, 2020. doi: 10.1038/s41746-020-00324-0. URL https://doi.org/10.1038/s41746-020-00324-0.

[43] Dimitris Bertsimas and Melvyn Sim. The price of robustness. *Operations Research*, 52(1):35–53, 2004.

[44] Dimitris Bertsimas, David B Brown, and Constantine Caramanis. Theory and applications of robust optimization. *SIAM Review*, 53(3):464–501, 2011.

[45] Dimitris Bertsimas, Vivek F Farias, and Nikolaos Trichakis. The price of fairness. *Operations Research*, 59(1):17–31, 2011.

[46] Hoda Bidkhori, John Dickerson, Duncan McElfresh, and Ke Ren. Kidney exchange with inhomogeneous edge existence uncertainty. In Jonas Peters and David Sontag, editors, *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 124 of *Proceedings of Machine Learning Research*, pages 161–170. PMLR, 03–06 Aug 2020.

[47] Reuben Binns. Fairness in machine learning: Lessons from political philosophy. *Proceedings of Machine Learning Research*, 81:1–11, 2017.

[48] Reuben Binns, Max Van Kleek, Michael Veale, Ulrik Lyngs, Jun Zhao, and Nigel Shadbolt. 'It's reducing a human being to a percentage': Perceptions of justice in algorithmic decisions. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 377. ACM, 2018.

[49] Péter Biró, David F Manlove, and Romeo Rizzi. Maximum weight cycle packing in directed graphs, with application to kidney exchange programs. *Discrete Mathematics, Algorithms and Applications*, 1(04):499–517, 2009.

[50] Péter Biró, Lisa Burnapp, Bernadette Haase, Aline Hemke, Rachel Johnson, Joris van de Klundert, and David Manlove. Kidney exchange practices in Europe, 2017. First Handbook of the COST Action CA15210: European Network for Collaboration on Kidney Exchange Programmes.

[51] Péter Biró, Bernadette Haase-Kromwijk, Tommy Andersson, Eyjólfur Ingi Ásgeirsson, Tatiana Baltesová, Ioannis Boletis, Catarina Bolotinha, Gregor Bond, Georg Böhmig, Lisa Burnapp, Katarína Cechlárová, Paola Di Ciaccio, Jiri Fronek, Karine Hadaya, Aline Hemke, Christian Jacquelinet, Rachel Johnson, Rafal Kieszek, Dirk R Kuypers, Ruthanne Leishman, Marie-Alice Macher, David Manlove, Georgia Menoudakou, Mikko Salonen, Bart Smeulders, Vito Sparacino, Frits C R Spieksma, María Oliva Valentín, Nic Wilson, and Joris van der Klundert. Building kidney exchange programmes in european overview of exchange practice and activities. *Transplantation*, 103(7):1514–1522, Jul 2019. ISSN 1534-6080 (Electronic); 0041-1337 (Print); 0041-1337 (Linking). doi: 10.1097/TP.0000000000002432.

[52] Simon Blackburn. Dilemmas: Dithering, plumping, and grief. In H. E. Mason, editor, *Moral Dilemmas and Moral Theory*, page 127. Oxford University Press, 1996.

[53] Avrim Blum, Jeffrey Jackson, Tuomas Sandholm, and Martin Zinkevich. Preference elicitation and query learning. *Journal of Machine Learning Research*, 5 (Jun):649–667, 2004.

[54] Avrim Blum, Anupam Gupta, Ariel D. Procaccia, and Ankit Sharma. Harnessing the power of two crossmatches. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 123–140, 2013.

[55] 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.

[56] 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. *Operations Research*, 2020. Earlier version appeared in the ACM Conference on Economics and Computation (EC), 2015.

[57] Susanne Bødker and Morten Kyng. Participatory design that matters—facing the big issues. *ACM Trans. Comput.-Hum. Interact.*, 25(1), February 2018. ISSN 1073-0516. doi: 10.1145/3152421. URL https://doi.org/10.1145/3152421.

[58] Jean-François Bonnefon, Azim Shariff, and Iyad Rahwan. The social dilemma of autonomous vehicles. *Science*, 352(6293):1573–1576, 2016.

[59] Sylvain Bouveret, Ulle Endriss, Jérôme Lang, et al. Fair division under ordinal preferences: Computing envy-free allocations of indivisible goods. In *ECAI*, pages 387–392, 2010.

[60] Bruno Bouzy. Associating shallow and selective global tree search with Monte Carlo for 9× 9 Go. In *International Conference on Computers and Games*, pages 67–80. Springer, 2004.

[61] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[62] R. I. Brafman, C. Domshlak, S. E. Shimony, and Y. Silver. Preferences over sets. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2*, AAAI'06, page 1101–1106. AAAI Press, 2006. ISBN 9781577352815.

[63] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D Procaccia. *Handbook of computational social choice*. Cambridge University Press, 2016.

[64] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D Procaccia. *Handbook of computational social choice*. Cambridge University Press, 2016.

[65] Gerhard Brewka, Miroslaw Truszczynski, and Stefan Woltran. Representing preferences among sets. *Proceedings of the AAAI Conference on Artificial Intelligence*, 24(1), Jul. 2010. URL https://ojs.aaai.org/index.php/AAAI/article/view/7584.

[66] Peter J Brown, John D Bovey, and Xian Chen. Context-aware applications: from the laboratory to the marketplace. *IEEE personal communications*, 4(5): 58–64, 1997.

[67] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In Sorelle A. Friedler

and Christo Wilson, editors, *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, volume 81 of *Proceedings of Machine Learning Research*, pages 77–91, New York, NY, USA, 23–24 Feb 2018. PMLR. URL http://proceedings.mlr.press/v81/buolamwini18a.html.

[68] U.S. Census Bureau. 2017 us census demographics, 2017. URL https://data.census.gov/cedsci.

[69] Nadia Burkart and Marco F Huber. A survey on the explainability of supervised machine learning. *Journal of Artificial Intelligence Research*, 70:245–317, 2021.

[70] Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356 (6334):183–186, 2017.

[71] Colin Camerer and Martin Weber. Recent developments in modeling preferences: Uncertainty and ambiguity. *Journal of Risk and Uncertainty*, 5(4):325–370, 1992. doi: 10.1007/BF00122575. URL https://doi.org/10.1007/BF00122575.

[72] Ioannis Caragiannis, Christos Kaklamanis, Panagiotis Kanellopoulos, and Maria Kyropoulou. The efficiency of fair division. International Workshop on Internet and Network Economics (WINE), 2009.

[73] Anna Bárbara Carneiro-Proietti, Ester C Sabino, Divaldo Sampaio, Fernando A Proietti, Thelma T Gonçalez, Cláudia DL Oliveira, João E Ferreira, Jing Liu, Brian Custer, George B Schreiber, et al. Demographic profile of blood donors at three major brazilian blood centers: results from the international reds-ii study, 2007 to 2008. *Transfusion*, 50(4):918–925, 2010.

[74] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1721–1730, 2015.

[75] Margarida Carvalho, Xenia Klimentova, Kristiaan Glorie, Ana Viana, and Miguel Constantino. Robust models for the kidney exchange problem. *INFORMS Journal on Computing*, 2020.

[76] Lok Chan, Kenzie Doyle, Duncan McElfresh, Vincent Conitzer, John P. Dickerson, Jana Schaich Borg, and Walter Sinnott-Armstrong. Artificial artificial intelligence: Measuring influence of ai 'assessments' on moral decision-making. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, AIES '20, page 214–220, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450371100. doi: 10.1145/3375627.3375870. URL https://doi.org/10.1145/3375627.3375870.

[77] Ruth Chang. The possibility of parity. *Ethics*, 112(4):659–688, July 2002.

[78] Kathleen Chell, Tanya E Davison, Barbara Masser, and Kyle Jensen. A systematic review of incentives in blood donation. *Transfusion*, 58(1):242–254, 2018.

[79] Chaofan Chen, Oscar Li, Chaofan Tao, Alina Jade Barnett, Jonathan Su, and Cynthia Rudin. This looks like that: deep learning for interpretable image recognition. *arXiv preprint arXiv:1806.10574*, 2018.

[80] Robert S Chen, Brendan Lucier, Yaron Singer, and Vasilis Syrgkanis. Robust optimization for non-convex objectives. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, pages 4708–4717, 2017.

[81] Yanhua Chen, Yijiang Li, John D. Kalbfleisch, Yan Zhou, Alan Leichtman, and Peter X.-K. Song. Graph-based optimization algorithm and software on kidney exchanges. *IEEE Transactions on Biomedical Engineering*, 59:1985–1991, 2012.

[82] Zhi-Zhong Chen, Ruka Tanahashi, and Lusheng Wang. An improved randomized approximation algorithm for maximum triangle packing. *Discrete Applied Mathematics*, 2009.

[83] Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.

[84] Alexandra Chouldechova and Aaron Roth. The frontiers of fairness in machine learning. *arXiv preprint arXiv:1810.08810*, 2018.

[85] Kirpal S. Chugh and Vivekanand Jha. Problems and outcomes of living unrelated donor transplants in the developing countries. *Kidney International*, 57:S131–S135, 2021/07/28 2000. doi: 10.1046/j.1523-1755.2000.07421.x. URL https://doi.org/10.1046/j.1523-1755.2000.07421.x.

[86] Mark Cieliebak, Stephan Eidenbenz, Aris Pagourtzis, and Konrad Schlude. On the complexity of variations of equal sum subsets. *Nord. J. Comput.*, 14(3): 151–172, 2008.

[87] Cint. Cint. URL https://www.cint.com/.

[88] Wolfram Conen and Tuomas Sandholm. Preference elicitation in combinatorial auctions. In *Proceedings of the 3rd ACM conference on Electronic Commerce*, pages 256–259. ACM, 2001.

[89] Vincent Conitzer, Walter Sinnott-Armstrong, Jana Schaich Borg, Yuan Deng, and Max Kramer. Moral decision making frameworks for artificial intelligence. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, pages 4831–4835, 2017.

[90] 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.

[91] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[92] N. Cowan, H. A. Gritsch, N. Nassiri, J. Sinacore, and J. Veale. Broken chains and reneging: A review of 1748 kidney paired donation transplants. *American Journal of Transplantation*, 17(9):2451–2457, 2017. doi: https://doi.org/10.1111/ajt.14343. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/ajt.14343.

[93] Ashley C Craig, Ellen Garbarino, Stephanie A Heger, and Robert Slonim. Waiting to give: stated and revealed preferences. *Management Science*, 63(11):3672–3690, 2017.

[94] Jon Crowcroft and Philippe Oechslin. Differentiated end-to-end internet services using a weighted proportional fair sharing tcp. *ACM SIGCOMM Computer Communication Review*, 28(3):53–69, 1998.

[95] David Danks and Alex John London. Algorithmic bias in autonomous systems. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, page 4691–4697. AAAI Press, 2017. ISBN 9780999241103.

[96] Jeffrey Dastin. Amazon scraps secret ai recruiting tool that showed bias against women, 2018. URL https://www.reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idUSKCN1MK08G.

[97] Amit Datta, Michael Carl Tschantz, and Anupam Datta. Automated experiments on ad privacy settings. *Proceedings on Privacy Enhancing Technologies*, 2015(1):92–112, 2015.

[98] Brett Day, Ian J Bateman, Richard T Carson, Diane Dupont, Jordan J Louviere, Sanae Morimoto, Riccardo Scarpa, and Paul Wang. Ordering effects and choice set awareness in repeat-response stated preference studies. *Journal of environmental economics and management*, 63(1):73–91, 2012.

[99] JR DeShazo and German Fermo. Designing choice sets for stated preference methods: the effects of complexity on choice consistency. *Journal of Environmental Economics and management*, 44(1):123–143, 2002.

[100] Guy Dewsbury, Mark Rouncefield, Karen Clarke, and Ian Sommerville. Depending on digital design: extending inclusivity. *Housing Studies*, 19(5):811–825, 2004. doi: 10.1080/0267303042000249224. URL https://doi.org/10.1080/0267303042000249224.

[101] Anind K Dey. Context-aware computing: The cyberdesk project. In *Proceedings of the AAAI 1998 Spring Symposium on Intelligent Environments*, pages 51–54, 1998.

[102] Anind K Dey. Understanding and using context. *Personal and ubiquitous computing*, 5(1):4–7, 2001.

[103] John P Dickerson. A unified approach to dynamic matching and barter exchange. Technical report, Doctoral Dissertation, Carnegie Mellon University, Pittsburgh, PA, 2016.

[104] John P. Dickerson and Tuomas Sandholm. FutureMatch: Combining human value judgments and machine learning to match in dynamic environments. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 622–628, 2015.

[105] John P. Dickerson, Ariel D. Procaccia, and Tuomas Sandholm. Dynamic matching via weighted myopia with application to kidney exchange. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 1340–1346, 2012.

[106] John P. Dickerson, Ariel D. Procaccia, and Tuomas Sandholm. Optimizing kidney exchange with transplant chains: Theory and reality. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 711–718, 2012.

[107] John P. Dickerson, Ariel D. Procaccia, and Tuomas Sandholm. Failure-aware kidney exchange. In *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pages 323–340, 2013.

[108] John P. Dickerson, Ariel D. Procaccia, and Tuomas Sandholm. Price of fairness in kidney exchange. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 1013–1020, 2014.

[109] 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.

[110] John P. Dickerson, Ariel D. Procaccia, and Tuomas Sandholm. Failure-aware kidney exchange. *Management Science*, 2018. To appear; earlier version appeared at EC-13.

[111] John P Dickerson, Karthik A Sankararaman, Aravind Srinivasan, and Pan Xu. Allocation problems in ride-sharing platforms: Online matching with offline reusable resources. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[112] John P Dickerson, Ariel D Procaccia, and Tuomas Sandholm. Failure-aware kidney exchange. *Management Science*, 65(4):1768–1791, 2019. Earlier version appeared in the ACM Conference on Economics and Computation (EC), 2013.

[113] Mary Dillon, Fabricio Oliveira, and Babak Abbasi. A two-stage stochastic programming model for inventory management in the blood supply chain. *International Journal of Production Economics*, 187:27–41, 2017.

[114] Yichuan Ding, Dongdong Ge, Simai He, and Christopher T. Ryan. A non-asymptotic approach to analyzing kidney exchange graphs. *Operations Research*, 2018. To appear; earlier version appeared at EC-15.

[115] Alan Donagan. Consistency in rationalist moral systems. *Journal of Philosophy*, 81(6):291–309, 1984. doi: jphil198481650.

[116] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning, 2017.

[117] John A Doucette, Kate Larson, and Robin Cohen. Conventional machine learning for social choice. In *AAAI*, pages 858–864, 2015.

[118] Upol Ehsan, Q Vera Liao, Michael Muller, Mark O Riedl, and Justin D Weisz. Expanding explainability: Towards social transparency in ai systems. *arXiv preprint arXiv:2101.04719*, 2021.

[119] Upol Ehsan, Q. Vera Liao, Michael Muller, Mark O. Riedl, and Justin D. Weisz. *Expanding Explainability: Towards Social Transparency in AI Systems*. Association for Computing Machinery, New York, NY, USA, 2021. ISBN 9781450380966. URL https://doi.org/10.1145/3411764.3445188.

[120] Hadi El-Amine, Ebru K Bish, and Douglas R Bish. Robust postdonation blood screening under prevalence rate uncertainty. *Operations Research*, 66(1):1–17, 2018.

[121] Glyn Elwyn, Dominick Frosch, Richard Thomson, Natalie Joseph-Williams, Amy Lloyd, Paul Kinnersley, Emma Cording, Dave Tomson, Carole Dodd, Stephen Rollnick, et al. Shared decision making: a model for clinical practice. *Journal of general internal medicine*, 27(10):1361–1367, 2012.

[122] Ulle Endriss, Maria Silvia Pini, Francesca Rossi, and K Brent Venable. Preference aggregation over restricted ballot languages: Sincerity and strategy-proofness. In *Twenty-First International Joint Conference on Artificial Intelligence*, 2009.

[123] Paul Erdös and Tibor Gallai. On maximal paths and circuits of graphs. *Acta Mathematica Hungarica*, 10(3-4):337–356, 1959.

[124] BS Everitt and A Skrondal. *The Cambridge Dictionary of Statistics*. Cambridge University Press, 4th edition, 2010.

[125] Wenyi Fang, Aris Filos-Ratsikas, Søren Kristoffer Stiil Frederiksen, Pingzhong Tang, and Song Zuo. Randomized assignments for barter exchanges: Fairness vs. efficiency. In *International Conference on Algorithmic Decision Theory (ADT)*, 2015.

[126] Golnoosh Farnadi, Behrouz Babaki, and Margarida Carvalho. Fairness in kidney exchange programs through optimal solutions enumeration. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

[127] Federal Trade Comission. Using artificial intelligence and algorithms. FTC Business Blog, April 2020. URL https://www.ftc.gov/news-events/blogs/business-blog/2020/04/using-artificial-intelligence-algorithms.

[128] Federal Trade Comission. Aiming for truth, fairness, and equity in your company's use of ai. FTC Business Blog, April 2021. URL https://www.ftc.gov/news-events/blogs/business-blog/2021/04/aiming-truth-fairness-equity-your-companys-use-ai.

[129] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 259–268. ACM, 2015.

[130] Paolo Ferrari, Willem Weimar, Rachel J. Johnson, Wai H. Lim, and Kathryn J. Tinckam. Kidney paired donation: principles, protocols and programs. *Nephrology Dialysis Transplantation*, 30(8):1276–1285, 10 2014. ISSN 0931-0509. doi: 10.1093/ndt/gfu309. URL https://doi.org/10.1093/ndt/gfu309.

[131] Rachel Freedman, Jana Schaich Borg, Walter Sinnott-Armstrong, John P Dickerson, and Vincent Conitzer. Adapting a kidney exchange algorithm to align with human values. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[132] Rachel Freedman, J Schaich Borg, Walter Sinnott-Armstrong, J Dickerson, and Vincent Conitzer. Adapting a kidney exchange algorithm to align with human values. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2018.

[133] Rachel Freedman, Jana Schaich Borg, Walter Sinnott-Armstrong, John P Dickerson, and Vincent Conitzer. Adapting a kidney exchange algorithm to align with human values. *Artificial Intelligence*, page 103261, 2020.

[134] Rachel Freedman, Jana Schaich 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.

[135] Yoav Freund, H Sebastian Seung, Eli Shamir, and Naftali Tishby. Selective sampling using the query by committee algorithm. *Machine learning*, 28(2-3): 133–168, 1997.

[136] Carl Benedikt Frey and Michael A Osborne. The future of employment: How susceptible are jobs to computerisation? *Technological forecasting and social change*, 114:254–280, 2017.

[137] Adrian Furnham, Katherine Simmons, and Alastair McClelland. Decisions concerning the allocation of scarce medical resources. *Journal of Social Behavior and Personality*, 15(2):185, 2000.

[138] Adrian Furnham, Kathryn Thomson, and Alastair McClelland. The allocation of scarce medical resources across medical conditions. *Psychology and Psychotherapy: Theory, Research and Practice*, 75(2):189–203, 2002.

[139] Amelia Gangemi and Francesco Mancini. Moral choices: the influence of the do not play god principle. In *Proceedings of the 35th Annual Meeting of the Cognitive Science Society, Cooperative Minds: Social Interaction and Group Dynamics*,

pages 2973–2977. Cognitive Science Society, Austin, TX, 2013.

[140] Sommer Gentry, Dorry Segev, and R. A. Montgomery. A comparison of populations served by kidney paired donation and list paired donation. *American Journal of Transplantation*, 5(8):1914–1921, 2005.

[141] Georgios Gerasimou. Indecisiveness, undesirability and overload revealed through rational choice deferral. *The Economic Journal*, 128(614):2450–2479, 2018.

[142] Nina Gerber, Paul Gerber, and Melanie Volkamer. Explaining the privacy paradox: A systematic review of literature investigating privacy attitude and behavior. *Computers & Security*, 77:226–261, 2018. doi: 10.1016/j.cose.2018.04.002. URL https://app.dimensions.ai/details/publication/pub.1103193978.

[143] Ahad J Ghods. Current status of organ transplant in islamic countries. *Exp Clin Transplant*, 13 Suppl 1:13–17, Apr 2015. ISSN 2146-8427 (Electronic); 1304-0855 (Linking).

[144] Gerd Gigerenzer and Adrian Edwards. Simple tools for understanding risks: from innumeracy to insight. *Bmj*, 327(7417):741–744, 2003.

[145] Gerd Gigerenzer, Wolfgang Gaissmaier, Elke Kurz-Milcke, Lisa M Schwartz, and Steven Woloshin. Helping doctors and patients make sense of health statistics. *Psychological science in the public interest*, 8(2):53–96, 2007.

[146] Kristiaan Glorie. *Clearing barter exchange markets: Kidney exchange and beyond*. PhD dissertation, Erasmus University Rotterdam, 2014.

[147] Kristiaan M. Glorie. Estimating the probability of positive crossmatch after negative virtual crossmatch. Technical report, Erasmus School of Economics, 2012.

[148] 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.

[149] Gaston Godin, Paschal Sheeran, Mark Conner, Marc Germain, Danielle Blondeau, Camille Gagné, Dominique Beaulieu, and Herminé Naccache. Factors explaining the intention to give blood among the general population. *Vox sanguinis*, 89(3):140–149, 2005.

[150] Gaston Godin, Mark Conner, Paschal Sheeran, Ariane Bélanger-Gravel, and Marc Germain. Determinants of repeated blood donation among new and experienced blood donors. *Transfusion*, 47(9):1607–1615, 2007.

[151] Nina Grgic-Hlaca, Elissa M Redmiles, Krishna P Gummadi, and Adrian Weller. Human perceptions of fairness in algorithmic decision making: A case study of criminal risk prediction. In *Proceedings of the 2018 World Wide Web Conference*, pages 903–912. International World Wide Web Conferences Steering Committee, 2018.

[152] Yue Guan. When voluntary donations meet the state monopoly: Understanding blood shortages in china. *The China Quarterly*, 236:1111–1130, 2018.

[153] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42, 2018.

[154] Jonathan Guo and Bin Li. The application of medical artificial intelligence technology in rural areas of developing countries. *Health Equity*, 2(1):174–181, 2018. doi: 10.1089/heq.2018.0037. URL https://doi.org/10.1089/heq.2018.0037. PMID: 30283865.

[155] Gurobi Optimization, Inc. Gurobi optimizer reference manual, 2018. URL http://www.gurobi.com.

[156] Chen Hajaj, John P. Dickerson, Avinatan Hassidim, Tuomas Sandholm, and David Sarne. Strategy-proof and efficient kidney exchange using a credit mechanism. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 921–928, 2015.

[157] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *NeurIPS*, pages 3315–3323, 2016.

[158] R. M. Hare. *Moral Thinking: Its Levels, Method, and Point*. Oxford: Oxford University Press, 1981.

[159] Johannes AM Harmsen, RMD Bernsen, Ludwien Meeuwesen, Duane Pinto, and MA Bruijnzeels. Assessment of mutual understanding of physician patient encounters: Development and validation of a mutual understanding scale (mus) in a multicultural general practice setting. *Patient Education and Counseling*, 59(2):171–181, 2005.

[160] Margaret C Harrell and Melissa A Bradley. Data collection methods. semi-structured interviews and focus groups. Technical report, Rand National Defense Research Inst santa monica ca, 2009.

[161] Galen Harrison, Julia Hanson, Christine Jacinto, Julio Ramirez, and Blase Ur. An empirical study on the perceived fairness of realistic, imperfect machine

learning models. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 392–402, 2020.

[162] A. Hart, J. M. Smith, M. A. Skeans, S. K. Gustafson, A. R. Wilk, S. Castro, J. Foutz, J. L. Wainright, J. J. Snyder, B. L. Kasiske, and A. K. Israni. Optn/srtr 2018 annual data report: Kidney. *American Journal of Transplantation*, 20(s1):20–130, 2020. doi: https://doi.org/10.1111/ajt.15672. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/ajt.15672.

[163] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[164] P. J. Held, F. McCormick, A. Ojo, and J. P. Roberts. A cost-benefit analysis of government compensation of kidney donors. *American Journal of Transplantation*, 16(3):877–885, 2016. doi: https://doi.org/10.1111/ajt.13490. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/ajt.13490.

[165] Macey L. Henderson. Social media in the identification of living kidney donors: Platforms, tools, and strategies. *Current Transplantation Reports*, 5 (1):19–26, 2018. doi: 10.1007/s40472-018-0179-8. URL https://doi.org/10.1007/s40472-018-0179-8.

[166] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

[167] Robin M Hogarth and Emre Soyer. Providing information for decision making: Contrasting description and simulation. *Journal of Applied Research in Memory and Cognition*, 4(3):221–228, 2015.

[168] John N Hooker and H Paul Williams. Combining equity and utilitarianism in a mathematical programming model. *Management Science*, 58(9):1682–1693, 2012.

[169] Joel Huber, Dick R Wittink, John A Fiedler, and Richard Miller. The effectiveness of alternative preference elicitation procedures in predicting choice. *Journal of Marketing Research*, 30(1):105–114, 1993.

[170] Johan Huysmans, Karel Dejaeger, Christophe Mues, Jan Vanthienen, and Bart Baesens. An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models. *Decis. Support Syst.*, 51(1):141–154, April 2011. ISSN 0167-9236. doi: 10.1016/j.dss.2010.12.003. URL http://dx.doi.org/10.1016/j.dss.2010.12.003.

[171] Arsh K. Jain, Peter Blake, Peter Cordy, and Amit X. Garg. Global trends in rates of peritoneal dialysis. *Journal of the American Society of Nephrology*, 23(3): 533–544, 2012. ISSN 1046-6673. doi: 10.1681/ASN.2011060607. URL https://jasn.asnjournals.org/content/23/3/533.

[172] Sérgio Jesus, Catarina Belém, Vladimir Balayan, João Bento, Pedro Saleiro, Pedro Bizarro, and João Gama. How can i choose an explainer? an application-grounded evaluation of post-hoc explanations. In *ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, 2021.

[173] Jeya Vikranth Jeyakumar, Joseph Noor, Yu-Hsi Cheng, Luis Garcia, and Mani Srivastava. How can i explain this to you? an empirical study of deep neural network explanation methods. In *Advances in Neural Information Processing Systems*, volume 33, pages 4211–4222, 2020. URL https://proceedings.neurips.cc/paper/2020/file/

2c29d89cc56cdb191c60db2f0bae796b-Paper.pdf.

[174] Stefan Johansson, Jan Gulliksen, and Catharina Gustavsson. Disability digital divide: the use of the internet, smartphones, computers and tablets among people with disabilities in sweden. *Universal Access in the Information Society*, 20 (1):105–120, 2021. doi: 10.1007/s10209-020-00714-x. URL https://doi.org/10.1007/s10209-020-00714-x.

[175] Anson Kahng, Min Kyung Lee, Ritesh Noothigattu, Ariel Procaccia, and Christos-Alexandros Psomas. Statistical foundations of virtual democracy. In *International Conference on Machine Learning*, pages 3173–3182, 2019.

[176] R. M. Karp, U. V. Vazirani, and V. V. Vazirani. An optimal algorithm for on-line bipartite matching. In *Proceedings of the Annual Symposium on Theory of Computing (STOC)*, pages 352–358, 1990.

[177] Bilal Kartal, Ernesto Nunes, Julio Godoy, and Maria Gini. Monte Carlo tree search with branch and bound for multi-robot task allocation. In *The IJCAI-16 workshop on autonomous mobile service robots*, volume 33, 2016.

[178] Korina Katsaliaki and Sally C Brailsford. Using simulation to improve the blood supply chain. *Journal of the Operational Research Society*, 58(2):219–227, 2007.

[179] Harmanpreet Kaur, Harsha Nori, Samuel Jenkins, Rich Caruana, Hanna Wallach, and Jennifer Wortman Vaughan. Interpreting interpretability: Understanding data scientists' use of interpretability tools for machine learning. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2020.

[180] Kidney Paired Donation Work Group. OPTN KPD pilot program cumulative match report (CMR) for KPD match runs: Oct 27, 2010 – Apr 15, 2013, 2013.

[181] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2668–2677, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/kim18d.html.

[182] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. In Christos H. Papadimitriou, editor, *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 43:1–43:23. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-029-3. doi: 10.4230/LIPIcs.ITCS.2017.43. URL http://drops.dagstuhl.de/opus/volltexte/2017/8156.

[183] Anton J Kleywegt, Alexander Shapiro, and Tito Homem-de Mello. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization*, 12(2):479–502, 2002.

[184] 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.

[185] Xenia Klimentova, Ana Viana, João Pedro Pedroso, and Nicolau Santos. Fairness models for multi-agent kidney exchange programmes. *Omega*, page

102333, 2020.

[186] Levente Kocsis and Csaba Szepesvári. Bandit based Monte-Carlo planning. In *European conference on machine learning*, pages 282–293. Springer, 2006.

[187] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, pages 4066–4076, 2017.

[188] Isaac Lage, Andrew Slavin Ross, Been Kim, Samuel J Gershman, and Finale Doshi-Velez. Human-in-the-loop interpretability prior. *arXiv preprint arXiv:1805.11571*, 2018.

[189] Isaac Lage, Emily Chen, Jeffrey He, Menaka Narayanan, Been Kim, Samuel J Gershman, and Finale Doshi-Velez. Human evaluation of models built for interpretability. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 7, pages 59–67, 2019.

[190] Himabindu Lakkaraju, Stephen H Bach, and Jure Leskovec. Interpretable decision sets: A joint framework for description and prediction. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1675–1684, 2016.

[191] Himabindu Lakkaraju, Nino Arsov, and Osbert Bastani. Robust and stable black box explanations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5628–5638. PMLR, 13–18 Jul 2020. URL http://proceedings.mlr.press/v119/lakkaraju20a.html.

[192] Jérôme Lang, Maria Silvia Pini, Francesca Rossi, Domenico Salvagnin, Kristen Brent Venable, and Toby Walsh. Winner determination in voting trees with

incomplete preferences and weighted votes. *Autonomous Agents and Multi-Agent Systems*, 25(1):130–157, 2012. doi: 10.1007/s10458-011-9171-8. URL https://doi.org/10.1007/s10458-011-9171-8.

[193] Heidi Ledford. Millions of black people affected by racial bias in health-care algorithms. *Nature*, 574:608+, 2021/8/3/ 2019. URL https://link.gale.com/apps/doc/A639205074/AONE?u=anon~c93a028c&sid=googleScholar&xid=dfd1c8c7.

[194] Min Kyung Lee. Understanding perception of algorithmic decisions: Fairness, trust, and emotion in response to algorithmic management. *Big Data & Society*, 5(1):2053951718756684, 2018.

[195] Min Kyung Lee and Su Baykal. Algorithmic mediation in group decisions: Fairness perceptions of algorithmically mediated vs. discussion-based social division. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, CSCW '17, pages 1035–1048, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4335-0. doi: 10.1145/2998181.2998230. URL http://doi.acm.org/10.1145/2998181.2998230.

[196] Min Kyung Lee, Anuraag Jain, Hae Jin Cha, Shashank Ojha, and Daniel Kusbit. Procedural justice in algorithmic fairness: Leveraging transparency and outcome control for fair algorithmic mediation. In *Proc. ACM Hum.-Comput. Interact., 3, CSCW*, page Article 182, New York, NY, USA, 2019. ACM. URL https://doi.org/10.1145/3359284.

[197] Min Kyung Lee, Daniel Kusbit, Anson Kahng, Ji Tae Kim, Xinran Yuan, Allissa Chan, Daniel See, Ritesh Noothigattu, Siheon Lee, Alexandros Psomas, and Ariel D. Procaccia. Webuildai: Participatory framework for algorithmic

governance. *Proc. ACM Hum.-Comput. Interact.*, 3(CSCW), November 2019. doi: 10.1145/3359283. URL https://doi.org/10.1145/3359283.

[198] 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).

[199] Benjamin Letham, Cynthia Rudin, Tyler H McCormick, David Madigan, et al. Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. *Annals of Applied Statistics*, 9(3):1350–1371, 2015.

[200] Jian Li, Yicheng Liu, Lingxiao Huang, and Pingzhong Tang. Egalitarian pairwise kidney exchange: Fast algorithms via linear programming and parametric flow. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 445–452, 2014.

[201] Shengwu Li. Ethics and market design. *Oxford Review of Economic Policy*, Forthcoming.

[202] Zhuoshu Li, Kelsey Lieberman, William Macke, Sofia Carrillo, Chien-Ju Ho, Jason Wellen, and Sanmay Das. Incorporating compatible pairs in kidney exchange: A dynamic weighted matching model. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 349–367. ACM, 2019.

[203] Zachary C Lipton. The mythos of model interpretability. *Communications of the ACM*, 61(10):36–43, 2018.

[204] Yicheng Liu, Pingzhong Tang, and Wenyi Fang. Internally stable matchings and exchanges. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 1433–1439, 2014.

[205] László Lovász. On covering of graphs. In *Theory of Graphs (Proc. Colloq., Tihany, 1966)*, pages 231–236. Academic Press New York, 1968.

[206] Meghna Lowalekar, Pradeep Varakantham, and Patrick Jaillet. Online spatio-temporal matching in stochastic and dynamic domains. *Artificial Intelligence*, 261:71–112, 2018.

[207] Joy Lu, Dokyun (DK) Lee, Tae Wan Kim, and David Danks. Good explanation for algorithmic transparency. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, AIES '20, page 93, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450371100. doi: 10.1145/3375627.3375821. URL https://doi.org/10.1145/3375627.3375821.

[208] Mary Frances Luce. Choosing to avoid: Coping with negatively Emotion-Laden consumer decisions. *Journal of Consumer Research*, 24(4):409–433, March 1998.

[209] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[210] David Manlove and Gregg O'Malley. Paired and altruistic kidney donation in the UK: Algorithms and experimentation. *ACM Journal of Experimental Algorithmics*, 19(1), 2015.

[211] Allan B Massie, Joseph Leanza, Lara M Fahmy, Eric KH Chow, Niraj M Desai, Xun Luo, Elizabeth A King, Mary G Bowring, and Dorry L Segev. A risk index for living donor kidney transplantation. *American Journal of Transplantation*, 16 (7):2077–2084, 2016.

[212] Nicholas Mattei, Abdallah Saffidine, and Toby Walsh. Mechanisms for online organ matching. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

[213] Nicholas Mattei, Abdallah Saffidine, and Toby Walsh. Fairness in deceased organ matching. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 236–242, 2018.

[214] Duncan McElfresh, Michael Curry, Tuomas Sandholm, and John Dickerson. Improving policy-constrained kidney exchange via pre-screening. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 2674–2685. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/1bda4c789c38754f639a376716c5859f-Paper.pdf.

[215] Duncan C. McElfresh and John P. Dickerson. Balancing lexicographic fairness and a utilitarian objective with application to kidney exchange. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 1161–1168. AAAI Press, 2018. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16192.

[216] Duncan C McElfresh, Hoda Bidkhori, and John P Dickerson. Scalable robust kidney exchange. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):1077–1084, Jul. 2019. doi: 10.1609/aaai.v33i01.33011077. URL https://ojs.aaai.org/index.php/AAAI/article/view/3899.

[217] Duncan C. McElfresh, Christian Kroer, Sergey Pupyrev, Eric Sodomka, Karthik Abinav Sankararaman, Zack Chauvin, Neil Dexter, and John P. Dickerson. Matching algorithms for blood donation. In *Proceedings of the 21st ACM Conference on Economics and Computation*, EC '20, page 463–464, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379755. doi: 10.1145/3391403.3399458. URL https://doi.org/10.1145/3391403.3399458.

[218] Duncan C. McElfresh, Lok Chan, Kenzie Doyle, Walter Sinnott-Armstrong, Vincent Conitzer, Jana Schaich Borg, and John P. Dickerson. Indecision modeling. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(7):5975–5983, May 2021. URL https://ojs.aaai.org/index.php/AAAI/article/view/16746.

[219] Duncan C McElfresh, Ke Ren, John P Dickerson, and Hoda Bidkhori. Distributionally robust cycle and chain packing with application to organ exchange. *Proceedings of the winter simulation conference*, 2021.

[220] Alasdair McIntyre. Moral dilemmas. *Philosophy and Phenomenological Research*, 50(n/a):367–382, 1990. doi: 10.2307/2108048.

[221] Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. Adwords and generalized online matching. *Journal of the ACM*, 54(5), 2007.

[222] Martin Mevissen, Emanuele Ragnoli, and Jia Yuan Yu. Data-driven distributionally robust polynomial optimization. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, pages 37–45, 2013.

[223] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019. ISSN 0004-3702. doi: https:

//doi.org/10.1016/j.artint.2018.07.007. URL https://www.sciencedirect.com/science/article/pii/S0004370218305988.

[224] Tim Miller, Piers Howe, and Liz Sonenberg. Explainable ai: Beware of inmates running the asylum or: How i learnt to stop worrying and love the social and behavioural sciences. *arXiv preprint arXiv:1712.00547*, 2017.

[225] Daniel Mochon. Single-option aversion. *Journal of Consumer Research*, 40(3): 555–566, 2013.

[226] Marco Molinaro and R. Ravi. Kidney exchanges and the query-commit problem. Manuscript, 2013.

[227] Christian Morath, Martin Zeier, Bernd Döhler, Gerhard Opelz, and Caner Süsal. Abo-incompatible kidney transplantation. *Frontiers in Immunology*, 8:234, 2017. ISSN 1664-3224. doi: 10.3389/fimmu.2017.00234. URL https://www.frontiersin.org/article/10.3389/fimmu.2017.00234.

[228] R Noothigattu, S Gaikwad, E Awad, S Dsouza, I Rahwan, P Ravikumar, and AD Procaccia. A voting-based system for ethical decision making. *AAAI 2018*, 2018.

[229] Ritesh Noothigattu, Snehalkumar 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.

[230] JC Nunnally. *Psychometric Theory*. McGraw-Hill, 2nd edition, 1978.

[231] Ziad Obermeyer, Brian Powers, Christine Vogeli, and Sendhil Mullainathan. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453, 2019. ISSN 0036-8075. doi: 10.1126/science.aax2342. URL https://science.sciencemag.org/content/366/6464/447.

[232] Annette M O'Connor. Validation of a decisional conflict scale. *Medical decision making*, 15(1):25–30, 1995.

[233] Carina Oedingen, Tim Bartling, and Christian Krauth. Public, medical professionals' and patients' preferences for the allocation of donor organs for transplantation: study protocol for discrete choice experiments. *BMJ open*, 8(10): e026040, 2018.

[234] Osonde A. Osoba and William Welser IV. *An Intelligence in Our Image: The Risks of Bias and Errors in Artificial Intelligence*. RAND Corporation, Santa Monica, CA, 2017. doi: 10.7249/RR1744.

[235] Andres F Osorio, Sally C Brailsford, and Honora K Smith. A structured review of quantitative models in the blood supply chain: a taxonomic framework for decision-making. *International Journal of Production Research*, 53(24):7191–7212, 2015.

[236] Andres F Osorio, Sally C Brailsford, Honora K Smith, Sonia P Forero-Matiz, and Bernardo A Camacho-Rodríguez. Simulation-optimization model for production planning in the blood supply chain. *Health Care Management Science*, 20(4):548–564, 2017.

[237] Sofia Ouhbi, José Luis Fernández-Alemán, Ambrosio Toval, Ali Idri, and José Rivera Pozo. Free blood donation mobile applications. *Journal of medical systems*, 39(5):52, 2015.

[238] D. Oyserman. Identity-based motivation. In Scott & S. Kosslyn, editor, *Emerging Trends in the Social and Behavioral Sciences*. John Wiley and Sons, Hoboken, NJ, 2015.

[239] Frank A Pasquale. Toward a fourth law of robotics: Preserving attribution, responsibility, and explainability in an algorithmic society. *Ohio State Law Journal*, 78, 2017.

[240] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[241] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[242] Robert B Peng, Haena Lee, Zheng T Ke, and Milda R Saunders. Racial disparities in kidney transplant waitlist appearance in chicago: Is it race or place?

*Clin Transplant*, 32(5):e13195, May 2018. ISSN 1399-0012 (Electronic); 0902-0063 (Print); 0902-0063 (Linking). doi: 10.1111/ctr.13195.

[243] Marek Petrik and Dharmashankar Subramanian. RAAM: The benefits of robustness in approximating aggregated MDPs in reinforcement learning. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, pages 1979–1987, 2014.

[244] Maria Silvia Pini, Francesca Rossi, Kristen Brent Venable, and Toby Walsh. Aggregating Partially Ordered Preferences. *Journal of Logic and Computation*, 19(3):475–502, 04 2008. ISSN 0955-792X. doi: 10.1093/logcom/exn012. URL https://doi.org/10.1093/logcom/exn012.

[245] Maria Silvia Pini, Francesca Rossi, Kristen Brent Venable, and Toby Walsh. Incompleteness and incomparability in preference aggregation: Complexity results. *Artificial Intelligence*, 175(7-8):1272–1289, 2011.

[246] Angelisa C Plane, Elissa M Redmiles, Michelle L Mazurek, and Michael Carl Tschantz. Exploring user perceptions of discrimination in online targeted advertising. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 935–951, 2017.

[247] Benjamin Plaut, John P. Dickerson, and Tuomas Sandholm. Fast optimal clearing of capped-chain barter exchanges. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 601–607, 2016.

[248] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. In *Advances in Neural Information Processing Systems*, pages 5680–5689, 2017.

[249] Michael Poss. Robust combinatorial optimization with variable cost uncertainty. *European Journal of Operational Research*, 237(3):836–845, 2014.

[250] Forough Poursabzi-Sangdeh, Daniel G Goldstein, Jake M Hofman, Jennifer Wortman Wortman Vaughan, and Hanna Wallach. Manipulating and measuring model interpretability. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–52, 2021.

[251] Gregory P Prastacos and Eric Brodheim. Pbds: a decision support system for regional blood management. *Management Science*, 26(5):451–463, 1980.

[252] Organ Procurement and Transplantation Network. Organ procurement and transplantation network (optn) policies, 2021. accessed from https://optn.transplant.hrsa.gov/media/1200/optn_policies.pdf on 7/28/2021.

[253] Organ Procurement and Transplantation Network. National organ data, 2021. data retrieved from OPTN, https://optn.transplant.hrsa.gov/data/view-data-reports/national-data/.

[254] László Pyber. Covering the edges of a connected graph by paths. *Journal of Combinatorial Theory, Series B*, 66(1):152–159, 1996.

[255] Peter Railton. Pluralism, determinacy, and dilemma. *Ethics*, 102(4):720–742, 1992. doi: 10.1086/293445.

[256] Bashir Rastegarpanah, Mark Crovella, and Krishna P. Gummadi. Fair inputs and fair outputs: The incompatibility of fairness in privacy and accuracy. In *Adjunct Publication of the 28th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '20 Adjunct, page 260–267, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379502. doi: 10.1145/3386392.3399568. URL https://doi.org/10.1145/3386392.3399568.

[257] J. Rawls. *A Theory of Justice*. Harvard University Press, 1971.

[258] Elissa M Redmiles, Sean Kross, and Michelle L Mazurek. How well do my results generalize? comparing security and privacy survey results from mturk, web, and telephone samples. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 1326–1343. IEEE, 2019.

[259] Michael Rees, Jonathan Kopke, Ronald Pelletier, Dorry Segev, Matthew Rutter, Alfredo Fabrega, Jeffrey Rogers, Oleh Pankewycz, Janet Hiller, Alvin Roth, Tuomas Sandholm, Utku Ünver, and Robert Montgomery. A nonsimultaneous, extended, altruistic-donor chain. *New England Journal of Medicine*, 360 (11):1096–1101, 2009.

[260] Pascale Reich, Paula Roberts, Nancy Laabs, Artina Chinn, Patrick McEvoy, Nora Hirschler, and Edward L Murphy. A randomized trial of blood donor recruitment strategies. *Transfusion*, 46(7):1090–1096, 2006.

[261] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939778. URL https://doi.org/10.1145/2939672.2939778.

[262] P.R. Rich, M.H. Van Loon, J. Dunlosky, and M.S. Zaragoza. Belief in corrective feedback for common misconceptions: Implications for knowledge revision. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 82:85–97, 2017.

[263] Nicholas Roberts, Spencer James, Meghan Delaney, and Christina Fitzmaurice. The global need and availability of blood products: a modelling study. *The Lancet Haematology*, 6(12):e606–e615, 2019.

[264] A Robinson and R Thomson. Variability in patient preferences for participating in medical decision making: implication for the use of decision support tools. *Qual Health Care*, 10 Suppl 1(Suppl 1):i34–8, Sep 2001. ISSN 0963-8172 (Print); 0963-8172 (Linking). doi: 10.1136/qhc.0100034..

[265] R Tyrrell Rockafellar, Stanislav Uryasev, et al. Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42, 2000.

[266] Andrew Slavin Ross, Michael C Hughes, and Finale Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations. *arXiv preprint arXiv:1703.03717*, 2017.

[267] Alvin Roth, Tayfun Sönmez, and Utku Ünver. Kidney exchange. *Quarterly Journal of Economics*, 119(2):457–488, 2004.

[268] Alvin Roth, Tayfun Sönmez, and Utku Ünver. A kidney exchange clearinghouse in New England. *American Economic Review*, 95(2):376–380, 2005.

[269] Alvin Roth, Tayfun Sönmez, and Utku Ünver. Pairwise kidney exchange. *Journal of Economic Theory*, 125(2):151–188, 2005.

[270] Alvin Roth, Tayfun Sönmez, and Utku Ünver. Efficient kidney exchange: Coincidence of wants in a market with compatibility-based preferences. *American Economic Review*, 97:828–851, 2007.

[271] Nick S Ryan, Jason Pascoe, and David R Morse. Enhanced reality fieldwork: the context-aware archaeological assistant. In *Computer applications in archaeology*. Tempus Reparatum, 1998.

[272] Debjani Saha, Candice Schumann, Duncan Mcelfresh, John Dickerson, Michelle Mazurek, and Michael Tschantz. Measuring non-expert comprehension of machine learning fairness metrics. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 8377–8387. PMLR, 13–18 Jul 2020. URL http://proceedings.mlr.press/v119/saha20c.html.

[273] Susan L. Saidman, Alvin Roth, Tayfun Sönmez, Utku Ünver, and Frank Delmonico. Increasing the opportunity of live kidney donation by matching for two and three way exchanges. *Transplantation*, 81(5):773–782, 2006.

[274] Milda R Saunders, Haena Lee, G Caleb Alexander, Hyo Jung Tak, J Richard Jr Thistlethwaite, and Lainie Friedman Ross. Racial disparities in reaching the renal transplant waitlist: is geography as important as race? *Clin Transplant*, 29(6):531–538, Jun 2015. ISSN 1399-0012 (Electronic); 0902-0063 (Print); 0902-0063 (Linking). doi: 10.1111/ctr.12547.

[275] Nripsuta Ani Saxena, Karen Huang, Evan DeFilippis, Goran Radanovic, David C Parkes, and Yang Liu. How do fairness definitions fare? Testing public attitudes towards three algorithmic definitions of fairness in loan allocations. *Artificial Intelligence*, 283:103238, 2020.

[276] Elke S Schaeffner, Jyotsna Mehta, and Wolfgang C Winkelmayer. Educational level as a determinant of access to and outcomes after kidney transplantation in the united states. *Am J Kidney Dis*, 51(5):811–818, May 2008. ISSN 1523-6838 (Electronic); 0272-6386 (Linking). doi: 10.1053/j.ajkd.2008.01.019.

[277] Leslie P Scheunemann and Douglas B White. The ethics and reality of rationing in medicine. *Chest*, 140(6):1625–1632, 2011.

[278] Johanes Schneider and Joshua Handali. Personalized explanation in machine learning: A conceptualization. In *In Proceedings of the 27th European Conference on Information Systems (ECIS)*, 2019. ISBN 978-1-7336325-0-8.

[279] Jesse D. Schold, Laura D. Buccini, David A. Goldfarb, Stuart M. Flechner, Emilio D. Poggio, and Ashwini R. Sehgal. Association between kidney transplant center performance and the survival benefit of transplantation versus dialysis. *Clinical Journal of the American Society of Nephrology*, 9(10):1773–1780, 2014. ISSN 1555-9041. doi: 10.2215/CJN.02380314. URL https://cjasn.asnjournals.org/content/9/10/1773.

[280] SEDAC. The gridded population of the world (gpw) data (version 4), developed by the center for international earth science information network (ciesin), columbia university and were obtained from the nasa socioeconomic data and applications center (sedac). http://sedac.ciesin.columbia.edu/data/collection/gpw-v4, 2020. Accessed: 7-10-2020.

[281] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[282] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[283] Alexander Shapiro and Tito Homem-de Mello. A simulation-based approach

to two-stage stochastic programming with recourse. *Mathematical Programming*, 81(3):301–325, 1998.

[284] Lloyd S Shapley and Martin Shubik. Game theory in economics, chapter 4:" preferences and utility"; presently. Technical report, Report R-904/4-NSF, RAND Corporation, Santa Monica, 1974.

[285] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[286] Walter Sinnott-Armstrong. *Moral Dilemmas*. Blackwell, 1988.

[287] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[288] Il'ya Meerovich Sobol'. On the distribution of points in a cube and the approximate evaluation of integrals. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 7(4):784–802, 1967.

[289] B Nilsson Sojka and Peter Sojka. The blood donation experience: self-reported motives and obstacles for donating blood. *Vox sanguinis*, 94(1):56–63, 2008.

[290] Kacper Sokol and Peter Flach. Explainability fact sheets: a framework for systematic assessment of explainable approaches. In *ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pages 56–67, 2020.

[291] Daniel A. Solomon, Nicole Rabidou, Sanjay Kulkarni, Richard Formica, and Liana Fraenkel. Accepting a donor kidney: an evaluation of patients' and transplant surgeons' priorities. *Clinical Transplantation*, 25(5):786–793,

2011. doi: https://doi.org/10.1111/j.1399-0012.2010.01342.x. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1399-0012.2010.01342.x.

[292] Tayfun Sönmez and M Utku Ünver. Market design for kidney exchange. *The Handbook of Market Design*, pages 93–137, 2013.

[293] Hossein Azari Soufiani, David C Parkes, and Lirong Xia. Preference elicitation for general random utility models. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, pages 596–605, 2013.

[294] J.T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. In *ICLR (workshop track)*, 2015. URL http://lmb.informatik.uni-freiburg.de/Publications/2015/DB15a.

[295] Megha Srivastava, Hoda Heidari, and Andreas Krause. Mathematical notions vs. human perception of fairness: A descriptive approach to fairness for machine learning. *CoRR*, abs/1902.04783, 2019. URL http://arxiv.org/abs/1902.04783.

[296] H Steihaus. The problem of fair division. *Econometrica*, 16:101–104, 1948.

[297] Ariane Sümnig, Martin Feig, Andreas Greinacher, and Thomas Thiele. The role of social media for blood donor motivation and recruitment. *Transfusion*, 58(10):2257–2259, 2018.

[298] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3319–3328. PMLR, 06–11 Aug 2017. URL http://proceedings.mlr.press/v70/sundararajan17a.html.

[299] Harini Suresh, Steven R Gomez, Kevin K Nam, and Arvind Satyanarayan. Beyond expertise and roles: A framework to characterize the stakeholders of interpretable machine learning and their needs. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–16, 2021.

[300] Efrain Talamantes, Keith C. Norris, Carol M. Mangione, Gerardo Moreno, Amy D. Waterman, John D. Peipert, Suphamai Bunnapradist, and Edmund Huang. Linguistic isolation and access to the active kidney transplant waiting list in the united states. *Clinical Journal of the American Society of Nephrology*, 12(3):483–492, 2017. ISSN 1555-9041. doi: 10.2215/CJN.07150716. URL https://cjasn.asnjournals.org/content/12/3/483.

[301] WHO Transplantation Taskforce. Position statement on the proposal for a global kidney exchange, 2018. URL https://www.who.int/transplantation/donation/GKE-statement.pdf.

[302] Amos Tversky and Eldar Shafir. Choice under conflict: The dynamics of deferred decision. *Psychological science*, 3(6):358–361, 1992.

[303] P. A. Ubel, J. Baron, and D. A. Asch. Social responsibility, personal responsibility, and prognosis in public judgments about transplant allocation. *Bioethics*, 1999.

[304] Kristen Underhill. Price and prejudice: An empirical test of financial incentives, altruism, and racial bias. *The Journal of Legal Studies*, 48(2):245–274, 2019. doi: 10.1086/707010. URL https://doi.org/10.1086/707010.

[305] UNOS. United Network for Organ Sharing (UNOS). http://www.unos.org/.

[306] UNOS. Revising kidney paired donation pilot program priority points, 2015. OPTN/UNOS Public Comment Proposal.

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

[308] Berk Ustun, Alexander Spangher, and Yang Liu. Actionable recourse in linear classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 10–19, 2019.

[309] Anne Van Dongen, Robert Ruiter, Charles Abraham, and Ingrid Veldhuizen. Predicting blood donation maintenance: the importance of planning future donations. *Transfusion*, 54(3pt2):821–827, 2014.

[310] John Vines, Rachel Clarke, Peter Wright, John McCarthy, and Patrick Olivier. *Configuring Participation: On How We Involve People in Design*, page 429–438. Association for Computing Machinery, New York, NY, USA, 2013. ISBN 9781450318990. URL https://doi.org/10.1145/2470654.2470716.

[311] Paul Voigt and Axel von dem Bussche. *The EU General Data Protection Regulation (GDPR): A Practical Guide*. Springer Publishing Company, Incorporated, 1st edition, 2017. ISBN 3319579584.

[312] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

[313] Brian Wahl, Aline Cossy-Gantner, Stefan Germann, and Nina R Schwalbe. Artificial intelligence (ai) and global health: how can ai contribute to health in resource-poor settings? *BMJ Global Health*, 3(4), 2018. doi: 10.1136/bmjgh-2018-000798. URL https://gh.bmj.com/content/3/4/e000798.

[314] 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.

[315] Xing Wang, Niels Agatz, and Alan Erera. Stable matching for dynamic ride-sharing systems. *Transportation Science*, 52(4):850–867, 2018.

[316] Helena Webb, Ansgar Koene, Menisha Patel, and Elvira Perez Vallejos. Multi-stakeholder dialogue for policy recommendations on algorithmic fairness. In *Proceedings of the 9th International Conference on Social Media and Society*, SM-Society '18, page 395–399, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450363341. doi: 10.1145/3217804.3217952. URL https://doi.org/10.1145/3217804.3217952.

[317] Marieke GM Weernink, Sarah IM Janus, Janine A Van Til, Dennis W Raisch, Jeannette G Van Manen, and Maarten J IJzerman. A systematic review to identify the use of preference elicitation methods in healthcare decision making. *Pharmaceutical medicine*, 28(4):175–185, 2014.

[318] James Wexler, Mahima Pushkarna, Tolga Bolukbasi, Martin Wattenberg, Fernanda Viégas, and Jimbo Wilson. The what-if tool: Interactive probing of machine learning models. *IEEE transactions on visualization and computer graphics*, 26(1):56–65, 2019.

[319] World Health Organization WHO. Blood safety and availability, 2017. URL https://www.who.int/news-room/fact-sheets/detail/blood-safety-and-availability. [Online; accessed 22-July-2004].

[320] Jenna Wiens, W. Nicholson Price, and Michael W. Sjoding. Diagnosing bias in data-driven algorithms for healthcare. *Nature Medicine*, 26(1):25–26, 2020. doi: 10.1038/s41591-019-0726-6. URL https://doi.org/10.1038/s41591-019-0726-6.

[321] Allison Woodruff, Sarah E Fox, Steven Rousso-Schindler, and Jeffrey Warshaw. A qualitative exploration of perceptions of algorithmic fairness. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 656. ACM, 2018.

[322] Huan Xu, Constantine Caramanis, and Shie Mannor. Robust regression and LASSO. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NIPS)*, pages 1801–1808, 2009.

[323] Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Suggala, David I Inouye, and Pradeep K Ravikumar. On the (in)fidelity and sensitivity of explanations. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper/2019/file/a7471fdc77b3435276507cc8f2dc2569-Paper.pdf.

[324] Özgür Yılmaz. Kidney exchange: An egalitarian mechanism. *Journal of Economic Theory*, 146(2):592–618, 2011.

[325] Shan Yuan, Shelley Chang, Kasie Uyeno, Gay Almquist, and Shirong Wang. Blood donation mobile applications: are donors ready? *Transfusion*, 56(3): 614–621, 2016.

[326] Andrea A. Zachary and Mary S. Leffell. Hla mismatching strategies for solid organ transplantation – a balancing act. *Frontiers in Immunology*, 7:

575, 2016. ISSN 1664-3224. doi: 10.3389/fimmu.2016.00575. URL https://www.frontiersin.org/article/10.3389/fimmu.2016.00575.

[327] Behzad Zahiri and Mir Saman Pishvaee. Blood supply chain network design considering blood group compatibility under uncertainty. *International Journal of Production Research*, 55(7):2013–2033, 2017.

[328] Dan Zakay. " to choose or not to choose": On choice strategy in face of a single alternative. *The American journal of psychology*, pages 373–389, 1984.

[329] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[330] Hanrui Zhang and Vincent Conitzer. A pac framework for aggregating agents' judgments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2237–2244, 2019.

[331] Ye Zhang, Ulf-G Gerdtham, Helena Rydell, and Johan Jarl. Socioeconomic inequalities in the kidney transplantation process: A registry-based study in sweden. *Transplantation direct*, 4(2):e346–e346, 02 2018. doi: 10.1097/TXD.0000000000000764. URL https://pubmed.ncbi.nlm.nih.gov/29464207.

[332] 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.