Matching Algorithms for Blood Donation

Duncan McElfresh[1] , **Christian Kroer**[2] , **Sergey Pupyrev**[2] , **Eric Sodomka**[2] , **Neil Dexter**[2] , **Zack Chauvin**[2] , **John P Dickerson**[1]

[1]University of Maryland, College Park
[2]Facebook

Abstract

Managing perishable inventory, such as blood stock awaiting use by patients in need, has been a topic of research for decades. Yet, most research focuses on the effective use of this scarce resource across the blood supply chain, and assumes the supply of blood itself can be impacted only via coarse policy levers. In this paper, empowered by the widespread geographic coverage of social networks, we choose instead to model the first stage of the full blood supply chain—that is, the supply of blood itself—as a matching market. Here, potential blood donors are matched to donation centers by way of a central recommendation engine; that engine can at some cost prompt (e.g., via push notification) individual donors to donate to a preferred center or centers. Potential donors may have other constraints (e.g., maximum allowable frequency of prompts) or preferences (e.g., geographic, social) that should be taken into account. We develop policies for matching potential blood donors to donation centers under these constraints and preferences, and under simple models of demand shocks to the system. We provide preliminary experimental results in simulation using real data from a large, worldwide social network, and show that our system provides lift relative to the status quo allocation methods.

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 [WHO, 2017]. The case remains, though, that many patients do not have timely access to blood, especially in times of need. Thus, the World Health Organization (WHO) recommends that the blood supply chain—collection, testing, processing, storage, and distribution—be managed at a national level [WHO, 2017].

Optimization-based approaches to management of the blood supply chain have a rich history in the operations research and healthcare management literature. A recent paper due to [Osorio *et al.*, 2015] overviews over 100 publications in this space since the 1960s. The supply chain is roughly split into collection, testing & processing, storage & inventory, and distribution [Osorio *et al.*, 2017]. Substantial research effort has gone into each of those segments [Katsaliaki and Brailsford, 2007; Zahiri and Pishvaee, 2017; Dillon *et al.*, 2017]. Yet, we note that most optimization-based research in the initial *collection* stage of the blood supply chain has focused on *prediction* of blood supply (e.g., during a crisis). Given the ubiquity of social networks, in this work, we dovetail with that research by focusing instead on the *creation* of new blood supply via automated social prompts, subject to the expressed preferences and constraints of potential donors and the overall donation system.

In this paper, we begin by formalizing a basic model of blood donation as an online matching problem (§2); here, we draw on intuition from the literature investigating online bipartite matching under various arrival distribution assumptions, per-vertex and per-side budget constraints, and so on [Goel and Mehta, 2008; Mehta, 2012]. Recent work in online matching has moved into more complex markets (e.g., crowdsourcing [Ho and Vaughan, 2012], rideshare [Dickerson *et al.*, 2018], and kidney exchange [Ashlagi *et al.*, 2013; Anderson *et al.*, 2017]), and has shown that—even in the face of theoretical intractability—sophisticated matching techniques can lead to improvements in economic efficiency. We aim to show a similar result here: we design stochastic matching policies and compare them against an (unrealistic) omniscient optimal policy and a (realistic) myopic greedy policy (§3). In early-stage, exploratory experiments using real data from a large social network, we show that our stochastic matching policy outperforms the status quo and performs well when compared to the omniscient policy (§4).

[1]Some examples follow. China maintains state control of its donation centers, which take a mix of captive-, quota-, and voluntary-based donations [Guan, 2018]. The US mixes state- and private-run donation that is primarily sourced via voluntary donations [Osorio *et al.*, 2015]. Brazil has seen a recent shift from remunerated to non-remunerated (aka voluntary) donation at its initially state-run, and now Federally-run, centers [Carneiro-Proietti *et al.*, 2010].

2 Blood donations matching model

We represent a blood donation problem as a weighted bipartite *donation graph* $G = (U, V, E)$, with donors $u \in U$ and recipients $v \in V$.[2] Each vertex has a set of *attributes* (e.g., blood type, geographical location, and so on); donor and recipient 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 $e = (u, v) \in E$. If an edge $e = (u, v)$ exists, then donor u can be *notified* to donate to v. Edge weights w_e represent the utility (in our early-stage work, the *likelihood*), of this donation.

Note on Edges: In this work we assume that compatibility and edge weights are determined *only* by the geographic distance between donor and recipient. Let $d(u, v)$ be that distance between u and v; the existence of edge $e = (u, v)$, and edge weight w_e, depend only on $d(u, v)$. Let $E_{:v}$ and $E_{u:}$ be the set of edges into recipient vertex v and out of donor vertex u, respectively.

Time Dependence: The donation graph G is inherently *dynamic*, in that some edges may only be in G temporarily.[3] We discretize time into days $t \in \mathcal{T} \equiv \{1, \dots T\}$, where $T < \infty$ is a finite-time horizon. Let $E(t)$ be the available edge set at time t; $E_{:v}(t)$ and $E_{u:}(t)$ are defined similarly. Furthermore, as a direct consequence of the above, we assume that changes to the edge set $E(t)$ are due only to dynamic *demand*.

Demand: We consider two types of recipients: *facilities* $S \subseteq V$, such as blood banks and hospitals, and *events* $D \subseteq V$, such as blood drives or emergency requests. Facilities $s \in S$ are available during *all* time steps, and edges into these recipients are always available (i.e., $E_{:s}(t) = E_{:s}$).

Events $d \in D$ arrive in an online manner, and are available only during a fixed subset of time steps, denoted by $T(d) \subset \mathcal{T}$. For time steps outside of this duration, $t' \notin T(d)$ the edge set is empty – i.e., $E_{:d}(t') \equiv \emptyset$. In a non-omniscient model we may need to reserve some supply for events; the motivation for this is that events typically have higher edge weights, and thus they may be worth "saving" supply for, even if they only occur stochastically.

Supply: Each donor may be notified once every K days (we use $K = 14$); this restriction is self-imposed, out of respect for the donors, and to avoid notification fatigue. Weight w_e is an estimate of the probability that sending this notification (i.e. prompting donor u to donate at demand site v) will result in a donation. Medical and legal restrictions require that donors donate *at most* once every 4-6 weeks; however we do not include this constraint in our model at this stage.

Objective In this initial study we focus on two objectives: (a) maximizing the number of donations (i.e., total edge weight, henceforth *efficiency*), while (b) evenly distributing donations among recipients; in Section 3.1 we discuss these objectives in greater depth. In general there are many additional objectives in blood donation, such as ensuring that donations do not exceed storage capacity, respecting donor preferences, and so on. Our model can be easily to extended to include these objectives, though we leave these extensions to future work.

Next we discuss three different solution *classes* for this problem.

[2] We use the terms "donors" and "recipients" as shorthand for *prospective* donors and recipients. The social network 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.

[3] 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.

3 Solution Classes

We consider three solution classes, which have access to different amounts of information about the particular donation scenario at hand. First we consider two extreme policies: omniscient (assuming *all* knowledge of future events) and myopic (assuming *no* knowledge of future events).

3.1 Omniscient Policy

We first consider an overly optimistic setting where we know the exact set of events that occur at each time step, as well as all supplies and demands; in other words we have a fully-specified graph G. This policy illustrates the *best possible* behavior of any allocation policy, and serves as an upper-bound on performance.

The omniscient policy is expressible as an integer linear program (ILP) – using decision variables x_{et} which are 1 if edge e is matched at time step t and 0 otherwise. Recall that our primary objective is to maximize overall efficiency; this is easily represented in the objective as $\sum_{t\in\mathcal{T}} \sum_{e\in E(t)} w_e x_{et}$. Our secondary objective is to ensure that notifications are allocated *equitably* between recipients. We leave the particular notion of equity up to the notification planner. However we assume they provide us with two pieces of information: 1) a function $q(\mathbf{x})$ that measure some aspect of *equity* of notifications $\mathbf{x}$, and 2) a parameter $\lambda \in [0, 1]$ that controls the relative importance of equity, compared to efficiency. The full omniscient policy is expressed in Problem 1.

$$\begin{aligned} \max \quad & (1-\lambda) \sum_{t\in\mathcal{T}} \sum_{e\in E(t)} w_e x_{et} + \lambda\, q(\mathbf{x}) \\ \text{s.t.} \quad & \sum_{t'=t-K}^{t} \sum_{e\in E_{u:}(t')} x_{et'} \le 1 \quad \forall u \in U, t \in \mathcal{T} \\ & x_{et} \in \{0, 1\} \qquad\qquad\qquad\quad \forall e \in E, t \in \mathcal{T} \end{aligned} \tag{1}$$

In Section 4 we provide an example function $q(\mathbf{x})$, and demonstrate how parameter λ balances the two objectives of equity and efficiency.

3.2 Hardness

In this section we show that in general, solving the omniscient policy problem is NP-complete even in the case where there is no equitability, and we thus only care about maximizing the edge weights. The reduction is from 3SAT. In 3SAT we are given Boolean variables V' and clauses C, where each clause consists of three literals such that at least one of them must be true. The goal is to find an assignment of values to the variables such that each clause is satisfied.

Theorem 1. *Solving the omniscient policy problem is NP-hard, even in the case of no equitability function.*

Proof. Let there be $2|C|+2$ timesteps. Set $K=2$.

For each Boolean variable v_i', we make 2 donors T_i, F_i. Now for each i,t pair we make an "assignment-forcing" recipient that is only open on day t, and only T_i and F_i can donate to it, with an edge weight of $|C|$, and a capacity upper bound of 1 (no lower bound). In order to capture the $|C| * (2|C|+2)$ edge weight available for v_i we now have to assign T_i/F_i in alternating order and assign their supply to the recipient of that day. Assignment on odd days corresponds to the truth value of the variable (e.g. if T_i donates on odd days then we set v_i' to true).

Now, for each clause c_i (assume some arbitrary ordering of clauses), we will add an additional recipient r_i which is only available on days $2i-1, 2i$, and only the 3 literals corresponding to the clause have an edge leading to the recipient. The edge weight is $|C|+1/2$, and the capacity upper bound is 1 (no lower bound). The idea is that now 1 of the donors corresponding to a literal in c can switch to r_i, and thereby increase the edge weight of the matching by 1, as compared to donating to their assignment-forcing recipient. Note that this does not break the forcing of assignments: In order to capture at least $|C| * (2|C|+2)$ for any variable v_i both T_i and F_i must donate $|C|+1$ times, and since we add two extra days with no corresponding clauses this requires that they donate on alternating days.

Now, any satisfying assignment of the 3SAT formula can be used to construct a solution to the reduction that achieves an objective value of $|V'| * |C| * (2|C|+2) + \frac{1}{2}|C|$: since it will be in alternating order we get $|V'| * |C| * (2|C|+2)$ from the assignment-forcing recipients (or upgrades in some cases), and there will be $|C|$ "upgrades" since we know that at least 1 literal is available to switch to r_i for each clause c_i.

Conversely, any solution to the reduction that achieves $|V'| * |C| * (2|C|+2) + \frac{1}{2}|C|$ must be a solution to the 3SAT instance: every pair T_i, F_i must be in alternating order, otherwise they can achieve at most $((2|C|+2) - \frac{1}{2}) * |C|$ utility, since if they are not in alternating order then they must miss at least 1 day and thus $|C|$ units of edge weight, and they can at most make up for that by satisfying every clause which yields only $\frac{1}{2}|C|$. Since every pair T_i, F_i is in alternating order we have a valid assignment of truth values. Furthermore, the assignment must be such that every clause is satisfied, since otherwise the addition of $\frac{1}{2}|C|$ upgrades would not be achieved.

□

Here we showed only NP-hardness. NP-completeness is easily shown for the case where there is no equitability function. If there is an equitability function q then it depends on whether q is verifiable (for example, it is verifiable if q is MIP representable).

3.3 Min cost flow when the planning horizon $T \leq K$, the notification limit

In this section we assume that $T \leq K$, which means that every donor can be matched at most once. This assumption can be leveraged to make certain variants of our problem substantially more tractable. Furthermore, we assume that we are given upper and lower bounds u_v, l_v on demand for each recipient v. With these assumptions, an optimal assignment for the model can be computed in polynomial time (and efficiently in practice) via *min-cost flow*; this is easily seen by noting that the problem for a single time step is an instance of the *transportation problem*. Multiple timesteps can be handled by replicating each recipient T times, where replica t denotes the recipient receiving donors on the t'th day (this works because $T \leq K$ and thus each donor can be used at most once).

Formally, we construct the network flow problem as follows: each donor $u \in U$ is instantiated with a source node s_u with outgoing flow of 1. Each recipient $v \in V$ is represented by nodes $n_{v1}, \ldots, n_{vT}$, where n_{vt} represents the donors assigned to v at day t (we can of course leave out pairs v,t such that v is unavailable on day t, e.g. for events of limited duration). Each n_{vt} is connected to the single sink node, and the connecting edge has upper and lower bounds u_v, l_v on capacity, and cost 0, we let E_{vs} denote the edges from recipient v to the sink. For each edge in the original graph $e \in E$ we add edges $e_1, \ldots, e_T$, with e_t representing using the edge e on day t; each e_t has cost $-w_e$ and infinite capacity.

A solution to the min-cost flow model can be computed with the following LP

$$\begin{array}{lll} \min & \sum_{t\in\mathcal{T}} \sum_{e\in E} -w_e x_{et} & \\ \text{s.t.} & \sum_{t\in\mathcal{T}: t-K\leq t'\leq t} \sum_{e\in E_{u:}} x_{et'} \leq 1 & \forall u \in U, t \in \mathcal{T} \\ & l_v \leq x_{et} \leq u_v & \forall v \in V, e \in E_{vs}, t \in \mathcal{T} \\ & 0 \leq x_{et} \leq 1 & \forall e \in E, t \in \mathcal{T} \end{array} \tag{2}$$

Since each capacity is integral (or infinite), we know that an optimal integer solution exists, due to the well-known property that basis solutions of the min-cost flow LP are integral in this case (see e.g. [Vohra, 2011]). Since $T \leq K$ we know that each donor can be assigned at most one time, and thus the above construction is valid.

3.4 Greedy (Myopic) Policies

In this section we consider greedy policies, with the myopic objective of maximizing edge weight in only the *current* time step. Our greedy policies use a single parameter for each demand node $v \in V$: the *edge demand* l_v. At each time step, we aim to match l_v edges to vertex v; this is a natural way to "save" donors to meet future demand. Of course, each donor can be matched *at most* once every K time steps, so it may be impossible to meet edge demand at each time step. For this reason, the outcome depends on the *order* in which edges are matched. We discuss two orderings here. In each of these policies, when a donor is matched at time t, all edges incident to this donor are immediately removed from $E(t)$.

Greedy: Sequential At each time step t, we visit each recipient node $v \in V$ *sequentially*, in random order. When visiting vertex v, we match the l_v highest-weight edges in $E_{:v}(t)$. If $|E_{:v}(t)| < l_v$, then all edges in $E_{:v}(t)$ are matched.

Greedy: Round-Robin In this policy, we visit each vertex in a round-robin process, in random order. When visiting vertex v we match only one edge from $E_{:v}(t)$, unless $E_{:v}(t) = \emptyset$ or l_v edges have been matched already. This process repeats until no more recipients can be matched.

Next we propose a more reasonable policy, which makes probabilistic assumptions about future demand.

3.5 Stochastic Matching Policy

This policy uses probabilistic information about future events, and identifies the best edges to use in each time step *in expectation*. We assume a distribution over the frequency and duration of events; in our model, only edges are dynamic – vertices are static. Using these distributions we estimate a probability that each edge will exist in each future time step, p_{et}.

In order to match edges during time step t^*, we construct a set of *known* edges $E^L(t')$, for all $t' \leq t^*$ and a set of *probabilistic* edges $E^P(t')$ for all $t' > t^*$. Note that $E^L(t)$ is simply constructed through observation. We construct $E^P(t)$ by adding *all* edges that occur with nonzero probability, with weight $w_e p_{et}$ – scaled by the probability that edge e will be available. We then solve the ILP for the omniscient policy using $E^L(t)$ and $E^P(t)$, and use the decision variables (i.e. edges) for time step t^* only. Furthermore, we fix the decision variables for all prior time steps (i.e., these decision variables are treated as data). The ILP for this probabilistic policy, for time step t^* is given in Problem 3.

$$\begin{aligned}
\max \quad & (1-\lambda)\Big(\sum_{t\in\mathcal{T}:t\leq t^*}\sum_{e\in E^L(t)} w_e x_{et} \\
& \qquad + \sum_{t\in\mathcal{T}:t>t^*}\sum_{e\in E^P(t)} p_{et} w_e x_{et}\Big) + \lambda\, q(\mathbf{x}) \\
\text{s.t.} \quad & \sum_{t'=t-K}^{t}\sum_{e\in E_{u:}(t')} x_{et'} \leq 1 \quad \forall u \in U, t \in \mathcal{T} : t > t^* \\
& x_{et} \in \{0,1\} \qquad \forall e \in E(t), t \in \mathcal{T} : t > t^*
\end{aligned} \tag{3}$$

We include all prior time steps ($t < t^*$) in this formulation to incorporate all matched edges in the objective term $q(\mathbf{x})$, and to ensure that donors are matched at most once every K time steps.

Next demonstrate the behavior of each of these policies using numerical simulation.

4 Experiments

We test each matching policy described in Section 3 using simulated blood donations scenarios, generated using data from a real blood donations setting. To clearly differentiate these policies, we consider *only* event (dynamic) recipient nodes.

Estimating Random Graph Parameters We simulate random blood donation graphs using distributions that resemble a real donation scenario for a single city; all distributions are estimated from real data. Locations of both donors and recipients are represented by coordinates $(x, y) \in \mathbb{R}^2$; the city center is located at the origin $(0, 0)$. Each donor/recipient location is generated using a random distance $d \geq 0$ and angle $\phi \in [0, 2\pi]$ from the city center. Distances d are drawn from a Gamma distribution with shape $k = 3.28$ and scale $\lambda = 1.87$; directions ϕ are drawn uniformly from $\phi \in [0, 2\pi]$. Donor/recipient locations are defined as $(x, y) = (d \cdot \cos(\phi), d \cdot \sin(\phi))$.

In our experiments the number of events is a fixed parameter (e.g., N events occur within a fixed time span); the duration t of each event follows a simple discrete distribution:

$$\begin{aligned}
&P(t = 1) = 2/3 - \epsilon, \\
&P(t = 2) = 1/3 - \epsilon, \\
&P(t = y) = 2\epsilon/5, \ \text{ for } y = 3, \ldots, 7.
\end{aligned}$$

Generating Random Graphs We generate random donation graphs with the following components:

- 100 donors, with locations drawn from the distribution described above. Donors may be matched at most once every $K = 14$ days.
- 60 events, with frequency and duration drawn from the distributions described above, over a period of 90 days.
- edges for each donor u and event v, if distance $d(u, v)$ is no greater than 6. Edge weight is set to $w(u, v) \equiv 1/(1 + d(u, v))$.

Balancing Equity and Efficiency We define a simple notion of equity between recipient nodes, based on the *edge demand* used by our greedy policies (see Section 3.4). To be equitable, we endeavor to meet demand for each vertex, at each time step. The greedy policies use l_v as the target number of edges to match, as in Section 3.4. A *demand violation* occurs when the amount of blood supplied to a recipient (i.e., a facility or event) is less than the amount of blood demanded. For the omniscient and stochastic policies, we define the equity objective $q(\mathbf{x})$ as the negative sum of all demand violations; that is, we define r_{vt} as the total demand violation for vertex v at time step t, and define $q(\mathbf{x}) \equiv -\sum_{v\in V}\sum_{t\in\mathcal{T}} r_{vt}$.

For these experiments we set demand using a simple heuristic that divides available edges between facilities: $l_v = E_{:v}/(C_v \cdot F)$, where C_v is the number of other recipient nodes that share at least one donor with v (i.e. the number of v's *competitors*). The factor F determines the magnitude of demand; we set $F = 2$ for event nodes, and $F = 4$ for facility nodes.

We define r_{vt} using the following constraints, for each $t \in \mathcal{T}$ and $v \in V$:

$$\begin{aligned}
r_v &\geq l_v - \sum_{e\in E_{:v}} x_{ev} \\
r_v &\geq 0
\end{aligned}$$

The full mixed-integer program (MIP) formulation is given in Problem 4.

$$\begin{aligned}
\max \quad & (1-\lambda)\sum_{t\in\mathcal{T}}\sum_{e\in E(t)} w_e x_{et} - \lambda \sum_{v\in V}\sum_{t\in\mathcal{T}} r_{vt} \\
\text{s.t.} \quad & \sum_{t'=t-K}^{t}\sum_{e\in E_{u:}(t')} x_{et'} \leq 1 \quad \forall u \in U, t \in \mathcal{T} \\
& r_v \geq l_v - \sum_{e\in E_{:v}} x_{et} \qquad \forall v \in V, t \in \mathcal{T} \\
& r_{vt} \geq 0 \qquad \forall v \in V, t \in \mathcal{T} \\
& x_{et} \in \{0,1\} \qquad \forall e \in E, t \in \mathcal{T}
\end{aligned} \tag{4}$$

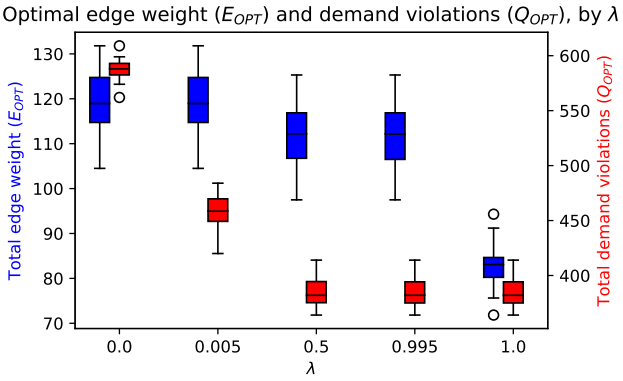

Figure 1: Optimal efficiency E_{OPT} and equity Q_{OPT} for various values of λ, calculated using the omniscient policy.

4.1 Results and Discussion

We generate 30 random graphs, and allocate notifications according to each of the three policies described in Section 3; for the omniscient and stochastic policies, we include the equity objective described above, with $\lambda \in [0, 1]$. For each graph and several values of λ, we calculate two metrics: the omniscient-optimal efficiency $E_{OPT}(\lambda)$ (the sum of edge weights), and the omniscient-optimal equity $Q_{OPT}(\lambda)$ (the sum of violations r_{vt}). Figure 1 shows boxplots of $E_{OPT}(\lambda)$ and $Q_{OPT}(\lambda)$ for each value of λ. These results represent the *best we can do*, with respect to the objective described in Section 3.1. Note that the parameter λ controls the trade-off between efficiency and equity.

The omniscient policy is not implementable in practice, but it serves as an upper-bound on performance; next we compare our more realistic policies to the omniscient policy. We test both greedy policies (sequential and round-robin), and the stochastic matching policy, using all 30 simulated donation graphs. In our implementation, the stochastic policy assumes that events will occur at one of 10 random locations drawn from the true location distribution, and will occur on any given day with probability $p = 0.1$. Both greedy policies attempt to satisfy edge demand l_v, for each facility and event node v, at each time step.

Using the efficiency objective ($E(\lambda)$) and equity objective ($Q(\lambda)$) obtained by each policy on these simulated graphs, we calculate the fractional optimality gap for both objectives, compared to the *best possible* E (with $\lambda = 0$) and Q (with $\lambda = 1$):

$$\%Q_{OPT}(\lambda) \equiv \frac{Q_{OPT}(1) - Q(\lambda)}{Q_{OPT}(1)} \times 100$$

$$\%E_{OPT}(\lambda) \equiv \frac{E_{OPT}(0) - E(\lambda)}{E_{OPT}(0)} \times 100$$

Figure 2 shows boxplots of $\%Q(\lambda)$ and $\%E(\lambda)$ for each policy: omniscient (`OPT`), stochastic (`STO`), greedy-round-robin (`GRD-RR`), and greedy-sequential (`GRD-SEQ`). The top figure shows the optimality gap for total edge weight. When $\lambda = 0$ (i.e., when the objective was to maximize total edge weight), the greedy algorithms on average had about 20%

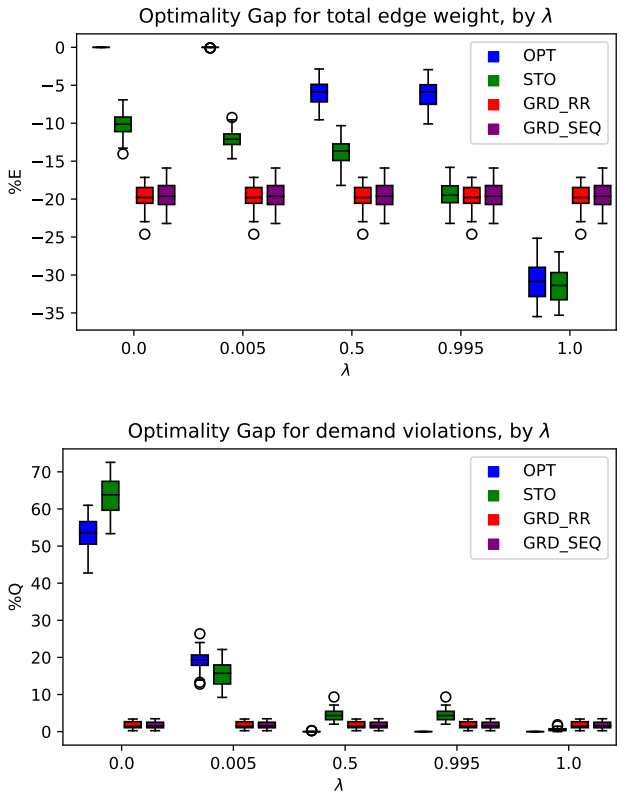

Figure 2: Optimality gap metrics $\%E_{OPT}(\lambda)$ and equity $\%Q_{OPT}(\lambda)$ for each λ.

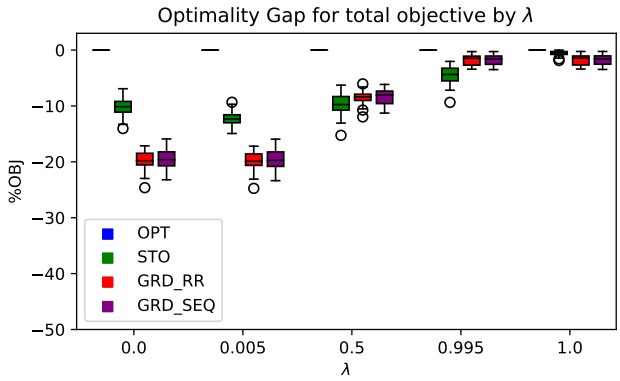

Figure 3: Total objective value for various values of λ.

less edge weight than the maximum total edge weight possible, compared to stochastic, which achieved about 10% less total edge weight than the maximum possible. When $\lambda = 1$ (i.e., when the objective was to maximize equity), the optimal and stochastic algorithms achieved about 30% less total edge weight than the maximum possible. The bottom figure shows the optimality gap for demand violations. When $\lambda = 1$, we see that stochastic algorithm had slightly fewer demand violations than greedy. In all where λ was positive the number of demand violations is small compared to the number of violations that occurred when $\lambda = 0$ (i.e., when equity was ignored).

While Figure 2 decomposes the performance in terms of edge weight and demand violations, that decomposition makes it difficult to understand the extent to which one algorithm outperforms another overall. Figure 3 aims to resolve this issue by looking at the total objective value achieved for different values of λ. That is, if λ is a parameter that indicates how much total value the platform is willing to sacrifice for an additional demand violation, then this figure shows the performance of different algorithms in terms of that objective. We see that the stochastic algorithm outperforms greedy for most values of λ, with the biggest improvement over greedy occurring for values of $\lambda \leq 0.5$. It should be noted that both greedy policies approach optimal as λ approaches 1. That is–in these experiments–greedy policies are effective at minimizing demand violations, but not very effective at maximizing edge weight.

5 Conclusions

We consider the problem of connecting blood donors with demand centers in a time-dependent setting, with uncertain demand. We formalize an objective for this problem, including the objectives of *efficiency* (maximizing the number of donations) and *equity* (i.e., between demand centers). We formulate an omniscient version of this problem as an integer program, and propose two realistic policies – stochastic and greedy – to which we compare the omniscient-optimal objective. Initial results indicate that there is a significant tradeoff between total edge weight (a proxy for efficiency) and total demand violations (a proxy for equity). Figure 1 demonstrates this tradeoff: a policy that *strictly* maximizes edge weight ($\lambda = 0$) also produces an extremely high number of demand violations (600, compared to the nominal 400). Similarly, a policy that *strictly* minimizes demand violations ($\lambda = 1$) also minimizes total edge weight (80, compared to the nominal 120). However for other cases, e.g. $\lambda = 0.5, 0.995$, total edge weight is nearly maximal, while total demand violations are nearly minimal. This indicates that a realistic policy *should* be able to achieve this balance of objectives.

We tested two types of policies: stochastic (which assumes some information about future demand), and greedy (which maximizes edge weight only for the immediate time step. As expected, our stochastic policy performs closer to optimal than either greedy policy.

In future work we intend to formulate a less-conservative stochastic matching policy, test these policies on real data, and analyze the theoretical optimality gap of each policy class.

We also plan to address other notions of equity between demand nodes, which more closely align with the goals of blood donations; for example, minimizing disparity in total edge weight (i.e., expected number of donations), or maximizing the minimum number of donations to each center.

Acknowledgments. Dickerson and McElfresh performed work as consultants for Facebook, who own the resulting intellectual property.

References

[Anderson *et al.*, 2017] Ross Anderson, Itai Ashlagi, David Gamarnik, and Yash Kanoria. Efficient dynamic barter exchange. *Operations Research*, 65(6):1446–1459, 2017.

[Ashlagi *et al.*, 2013] 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.

[Carneiro-Proietti *et al.*, 2010] 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.

[Dickerson *et al.*, 2018] 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.

[Dillon *et al.*, 2017] 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.

[Goel and Mehta, 2008] Gagan Goel and Aranyak Mehta. Online budgeted matching in random input models with applications to adwords. In *Proceedings of ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 982–991, 2008.

[Guan, 2018] Yue Guan. When voluntary donations meet the state monopoly: Understanding blood shortages in china. *The China Quarterly*, 236:1111–1130, 2018.

[Ho and Vaughan, 2012] Chien-Ju Ho and Jennifer Wortman Vaughan. Online task assignment in crowdsourcing markets. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2012.

[Katsaliaki and Brailsford, 2007] 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.

[Mehta, 2012] Aranyak Mehta. Online matching and ad allocation. *Theoretical Computer Science*, 8(4):265–368, 2012.

[Osorio *et al.*, 2015] 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.

[Osorio *et al.*, 2017] 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.

[Vohra, 2011] Rakesh V Vohra. *Mechanism design: a linear programming approach*, volume 47. Cambridge University Press, 2011.

[WHO, 2017] World Health Organization WHO. Blood safety and availability, 2017. [Online; accessed 22-July-2004].

[Zahiri and Pishvaee, 2017] 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.