LEARNING POTENTIALS FOR DYNAMIC MATCHING AND APPLICATION TO HEART TRANSPLANTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Each year, thousands of patients in need of heart transplants face life-threatening wait times due to organ scarcity. While allocation policies aim to maximize population-level outcomes, current approaches often fail to account for the dynamic arrival of organs and the composition of waitlisted candidates, thereby hampering efficiency. The United States is transitioning from rigid, rule-based allocation to more flexible data-driven models. In this paper, we propose a novel framework for non-myopic policy optimization in general online matching relying on *potentials*, a concept originally introduced for kidney exchange. We develop scalable and accurate ways of learning potentials that are higher-dimensional and more expressive than prior approaches. Our approach is a form of *self-supervised imitation learning*: the potentials are trained to mimic an omniscient algorithm that has perfect foresight. We focus on the application of heart transplant allocation and demonstrate, using real historical data, that our policies significantly outperform prior approaches—including the current US *status quo* policy and the proposed *continuous distribution* framework—in optimizing for population-level outcomes. Our analysis and methods come at a pivotal moment in US policy, as the current heart transplant allocation system is under review. We propose a scalable and theoretically grounded path toward more effective organ allocation.

1 INTRODUCTION

Organ transplantation is the preferred treatment for patients with end-stage heart failure. However, the demand for life-saving organs far exceeds the available supply, requiring an efficient and equitable allocation policy (Cameli et al., 2022). The difficulty of heart transplant allocation arises from its dynamic and time-sensitive nature. Heart transplantation depends exclusively on deceased donors. Since hearts have a limited cold ischemia time, they must be allocated and transplanted quickly upon availability. Decision-makers need to irrevocably match arriving organs with waiting patients without knowledge of future arrivals (Figure 1). The system involves a challenging tradeoff: accepting a marginal match for a patient may prevent them from receiving a superior match later on. Conversely, waiting for a better organ risks the patient dying on the waitlist. Organ allocation is arguably the highest-stakes application of online matching.

Following a major revision in 2018, the US heart transplant allocation policy is again transitioning to new solutions. The current system is a rigid, hierarchical tier-based system. The 2018 revision expanded a 3-tier system to a 6-tier one to better quantify the urgency of candidates (Kilic et al., 2021). However, the current system has a major limitation: it fails to leverage granular estimates of pretransplant mortality and post-transplant outcomes, often conflating heterogeneous patients within the same priority status (Shore et al., 2020; Zhang et al., 2024). This results in the system allocating scarce organs to patients with worse outcomes (Cogswell et al., 2020). To overcome these deficiencies, the *Organ Procurement and Transplantation Network (OPTN)* is reviewing the proposed *continuous distribution* framework (OPTN, 2024; Papalexopoulos et al., 2024). Continuous distribution, which is the current allocation scheme for lung transplants in the US, assigns each patient a real-valued score. While continuous distribution replaces rigid tiers with a more flexible score, it assumes that the score is a static weighted sum of patient feature values. This underlying class of policies is overly restrictive. Even with optimized weights, it lacks the expressiveness to consider patient-donor feature interactions and to fully adapt to the complex, non-linear dynamics of arriving

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

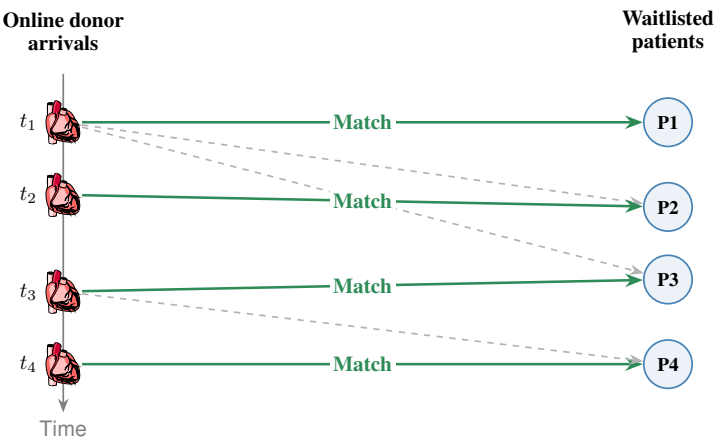


Figure 1: Schematic illustration of online heart transplant allocation.

organs and an evolving waitlist. Designing an allocation policy that maximizes population-level outcomes in this complicated setting is a key challenge.

1.1 Our contributions

In this paper, we propose a computational approach for designing a matching policy and show on real data that it significantly outperforms prior approaches and proposals.

Our methodology builds on the notion of *potentials*, which were introduced for kidney exchange (Dickerson et al., 2012; Dickerson & Sandholm, 2015). Kidney exchange is a different problem than online matching as it involves cycles longer than two (Roth et al., 2004; Abraham et al., 2007) and never-ending altruist donor chains (Rees et al., 2009; Dickerson et al., 2019; 2016). Potentials enable a policy to account for long-term pool composition by offsetting edge weights to reflect far-out utility rather than near-term gains. Those two prior potentials-based papers used *Sequential Model-based Algorithm Configuration (SMAC)* (Hutter et al., 2011) to learn low-dimensional linear potential functions. Preliminary work has shown that simple linear potentials outperform myopic policies in heart allocation (Anagnostides et al., 2025). However, as we will show, simple linear parameterizations are insufficient to capture the full complexity of heart transplant allocations. In general, we show that black-box optimization is insufficient for learning complex, non-linear potentials required for achieving better outcomes in heart transplant allocation.

To bypass the inefficiencies of black-box optimization, we formulate a novel approach for learning potential functions using offline self-supervised imitation learning. We develop an omniscient oracle that identifies optimal allocations in hindsight using full knowledge of donor arrivals over a historical period. Our policy optimization then learns potential functions to imitate the omniscient oracle.

Our framework not only improves the way potentials are learned but also enables the learning of more expressive potentials, such as high-dimensional and non-linear functions via self-supervised learning. These potential functions can reflect complex interactions between patient characteristics, donor features, the arrival distribution, and the waitlist state.

While our approach is applicable to general online matching problems, we focus on the application of heart transplant allocation. We demonstrate using real historical data from the *United Network for Organ Sharing (UNOS)* that our approach considerably outperforms all prior approaches and proposals in optimizing for population-level outcomes, including the current US *status quo* policy and the proposed continuous distribution framework. To our knowledge, our work is the first to thoroughly evaluate the continuous distribution framework from an optimization standpoint. We further demonstrate that our framework can effectively optimize weights within the continuous distribution system, offering an alternative to current committee-assigned or heuristic weight-selection methods. Finally, our best potential-based policy is near-optimal: it yields 95% of the solution quality upper bound that we generate using omniscient matching. We further show that the framework is agnostic to the underlying utility function: when optimizing a fairness-aware objective, our best

potential-based policy still achieves 96.5% of the omniscient upper bound utility, significantly higher than the baseline policies.

We also address data scarcity to reduce overfitting inherent in historical simulations. We introduce a data augmentation pipeline that generates diverse, medically realistic training scenarios from historical trajectories. Our method randomizes the temporal arrival of donors and patients while preserving real clinical features and progressions. We show that the augmentation helps the model learn potentials that capture the underlying matching logic rather than overfitting to historical timestamps.

In summary, we develop a new framework to harness the power of data-driven learning within online matching problems. We apply our framework to develop significantly better policies for heart transplant allocation that account for the long-term consequences of matching scarce resources. Our methods and experiments come at a pivotal moment in US policy, as the current heart transplant allocation system is under review.

2 Our heart transplant allocation model

We model the heart transplant allocation process as a dynamic online bipartite matching problem. In this section, we define the model, including the dynamics, the utility, and the optimization objective.

The allocation process unfolds over a discrete horizon $\mathcal{H} = \{1, \dots, T\}$. At each time step t, a donor organ (online node) arrives. The decision maker must then decide whether to match the organ to a patient (offline node) in the waitlist or discard it. We refer to the sequence of events over $\mathcal{H}$ as a *trajectory*. The system is defined by the following components:

- $\mathcal{D}$: the stream of donors. At time t, a donor $d^{(t)} \in \mathcal{D}$ arrives, characterized by a vector of features that includes blood type, medical data, and location. We use the superscript notation $d^{(t)}$ to denote the donor arrival at time t.
- $\mathcal{P}^{(t)}$: the set of patients waiting for a transplant at time t. Each patient $p^{(t)} \in \mathcal{P}^{(t)}$ is defined by a vector of features including blood type, medical data, and location. The pool is dynamic: new patients arrive, and existing patients may depart (due to death or delisting) or experience medical status updates. The notation $p^{(t)}$ denotes a patient's condition at time t.
- $\mathcal{C} : \mathcal{D} \times \mathcal{P} \to \{0, 1\}$ is a binary constraint function. A match between donor d and patient p is feasible if and only if $\mathcal{C}(d, p) = 1$ (*e.g.*, an edge exists in the bipartite graph). Constraints typically include blood type compatibility and maximum geographic distance.
- $\mathcal{U} : \mathcal{D} \times \mathcal{P} \to \mathbb{R}$: the utility function quantifying the value of a match. In this work, we maximize *population life years gained (PLYG)*, defined formally below. While we focus on PLYG, the framework generalizes to arbitrary utility functions, including those that capture fairness or equity.

In the context of abstract online matching, $\mathcal{D}$ is the set of online nodes, $\mathcal{P}$ is the set of offline nodes (each with capacity 1), $\mathcal{C}$ determines edge existence, and $\mathcal{U}$ determines edge weights. A key constraint in organ allocation is the viability of an organ. When a donor $d^{(t)}$ arrives, the allocation must be made immediately. An organ cannot be stored for future time steps. Therefore, the set of feasible matches for a donor $d^{(t)}$ is restricted to the set of compatible patients currently in the pool, denoted $\mathcal{P}_{d,t}$:

$$\mathcal{P}_{d,t} = \left\{ p^{(t)} \in \mathcal{P}^{(t)} \mid \mathcal{C}\left(d^{(t)}, p^{(t)}\right) = 1 \right\}.$$

A policy must select a patient $p^{(t)} \in \mathcal{P}_{d,t}$ or discard the organ.

PLYG is a utilitarian metric for organ allocation, defined as the expected increase in survival time for a patient receiving a donor organ versus remaining unmatched on the waitlist. This metric is standard in organ allocation research (*e.g.*, Berrevoets et al., 2020; 2021). Maximizing PLYG incentivizes matches that provide the largest marginal gain in survival. PLYG does not simply target the sickest patients (who may have low post-transplant survival) or the healthiest (who have high waitlist survival), but rather takes a population-level efficiency-focused approach. Beyond PLYG, there are also other objectives one can consider for the application, including fairness with respect to different demographic groups. Our framework is agnostic to the underlying utility function. For

example, the edge weights can be adjusted to prioritize marginalized populations, as has been done in kidney exchange research (Dickerson & Sandholm, 2015). We demonstrate this flexibility with a fairness-aware objective in Section J that balances the PLYG with the transplant rate for biologically disadvantaged patients.

Let $X = \left\{\left(d^{(t)}, p^{(t)}\right)\right\}_{t=1}^{T}$ be a sequence of matches made over the horizon $\mathcal{H}$, where t denotes the time of the match. The total utility is the sum of the marginal life years gained:

$$\mathcal{U}(X) = \sum_{\left(d^{(t)}, p^{(t)}\right) \in X} \mathcal{U}\left(d^{(t)}, p^{(t)}\right).$$

In an offline setting, where the full sequence of donors and patient dynamics is known in advance, the optimal allocation X^* is the maximum weight matching:

$$X^* = \arg\max_X \mathcal{U}(X) \text{ subject to } \forall \left(d^{(t)}, p^{(t)}\right) \in X, \mathcal{C}\left(d^{(t)}, p^{(t)}\right) = 1. \tag{1}$$

In the real-world online setting, the policy must select a match at time t without knowledge of future donor arrivals or patient dynamics. We provide an illustrative example that motivates *potentials* in the context of heart transplantation in Section B.

3 Potential-based matching

In online matching, an algorithm must balance immediate gains against future uncertainty. Potential-based policies operate as greedy algorithms that act on learned, modified weights; *potentials* are a data-driven approach for quantifying the future value of the resources being consumed (Dickerson et al., 2012). In the context of heart transplantation, considering the long-term consequences of matching patients on the waitlist carries high stakes. We detail our potential-based policy framework in the context of heart transplantation.

3.1 General policy framework

We define a value function $Q\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}; \theta\right)$ that assigns a scalar value to every viable match between the current donor $d^{(t)}$ and a patient $p^{(t)}$ in the waitlist $\mathcal{P}_{d,t}$. This function, parameterized by θ, augments the weights at each decision step in the matching problem. The weights θ are learned offline, while the online policy selects the patient that maximizes the score:

$$p_{selected} = \arg\max_{p^{(t)} \in \mathcal{P}_{d,t}} Q\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}; \theta\right).$$

The augmented edge weight is defined as a combination of the actual edge weight and a parameterized potential function P_θ:

$$Q\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}; \theta\right) = \underbrace{\mathcal{U}\left(d^{(t)}, p^{(t)}\right)}_{\text{immediate utility}} - \underbrace{P_\theta\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}\right)}_{\text{learned potential}}.$$

The potential function P_θ acts as a penalty for removing a patient from the waitlist (consuming a resource).

The learning process, detailed later in Section 4, seeks to learn parameters θ such that the induced policy maximizes global utility. A patient with low potential signals that matching immediately has high strategic value (*e.g.*, a patient is in critical condition or harder to match). On the other hand, a high potential indicates that maintaining a patient in the pool is desired. Generalized to abstract online matching, the potential function can depend on the offline nodes, the arriving online node, and any associated features.

While the potential-based matching framework generalizes to arbitrary potential functions, in this work, we utilize linear functions and neural networks (*Multi-Layer Perceptrons*) to bracket the

spectrum of interpretability and expressiveness. For linear potential functions, we utilize a feature set that encodes a patient's blood (O, A, B, and AB) and location (one of the nine geographical regions of the mainland United States), capturing two primary drivers of organ scarcity. For the neural networks, in addition to a low-dimensional input that encodes the patient's blood type, we employ a richer, 34-dimensional feature vector that captures both the characteristics of the specific match between the patient and donor and the broader context of the waitlist. We further discuss the details of the potential function architectures, hyperparameters, and feature sets in Section C

4 Learning the Potentials

The objective of learning a potential-based policy is to find the parameter vector θ that maximizes the cumulative utility of an online matching over time. However, directly optimizing θ based on simulator outcomes poses significant computational challenges. The objective function is non-convex and non-differentiable with respect to θ. The underlying system involves discrete, sequential decisions where small changes in weights can lead to discontinuities in the allocation policy.

We explore two distinct optimization paradigms illustrated in Figure 2. The first approach treats the simulator as a black box and optimizes θ using Bayesian optimization, specifically, SMAC. The optimizer iteratively proposes a candidate θ, runs the simulation, observes the total utility, and updates its probabilistic model of the response surface to propose the next θ. Black-box optimization via SMAC is the standard approach utilized in all prior potential-based matching literature (Dickerson et al., 2012; Anagnostides et al., 2025; Dickerson & Sandholm, 2015). SMAC suffers from severe sample inefficiency and has no finite-time convergence or quality guarantees. Each function evaluation requires simulating an entire trajectory. Furthermore, SMAC cannot scale to high-dimensional parameter spaces, such as the weights of a neural network. We use SMAC to benchmark the linear potential functions and show that it scales poorly in higher dimensions (Section K.3). Further discussion of the drawbacks of SMAC are provided in Section D.

The second—our primary novel contribution—is a self-supervised imitation learning framework for optimizing θ. Instead of optimizing the simulation outcomes directly, the policy learns to mimic the decisions of an optimal omniscient algorithm that can make allocations in hindsight using full knowledge of events. In other words, the omniscient algorithm to be mimicked has perfect foresight. For a given donor, we train the potential function to reflect that the omniscient algorithm's selected patient was chosen over all other candidates in the pool. Across a large-scale dataset of decisions, the potentials reflect the logic used by the optimal matches, identifying in what contexts patients should be matched versus maintained in the pool. In the next subsection we describe our imitation learning approach.

4.1 Our Imitation Learning Framework

Our imitation learning framework works using the optimal solutions from an omniscient algorithm, and we learn potential functions via supervised learning such that the induced ranking of patients mimics the optimal allocation. Our imitation learning framework offers significant advantages over black-box optimization. First, it is data-efficient. Obtaining ground-truth labels from the omniscient algorithm is computationally inexpensive and invariant to the dimensionality of the potential function. Second, it improves interpretability. Rather than optimizing a sparse, delayed reward signal, we explicitly train the policy to replicate specific decision patterns from the omniscient algorithm. Finally,

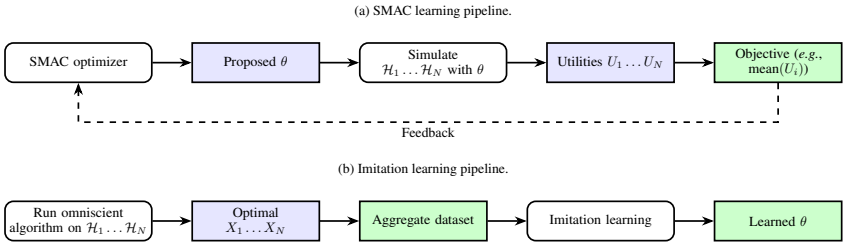


Figure 2: Learning pipelines using a set of trajectories.

the surrogate loss functions are differentiable (and convex in the linear case), which facilitates stable and efficient convergence using gradient-based optimization.

4.1.1 OMNISCIENT ALGORITHM

Supervising imitation learning requires ground-truth labels. We construct an omniscient algorithm that computes the optimal allocation over a fixed horizon, $\mathcal{H}$, in hindsight. The omniscient algorithm relies on perfect knowledge over the horizon, specifically, the complete sequence of donor arrivals and patient progressions. While using an omniscient algorithm to benchmark online algorithms is standard in operations research (*e.g.*, for computing competitive ratios), directly learning from these algorithms is relatively unexplored.

Let $\mathcal{D}$ be the set of donors arriving over a horizon $\mathcal{H}$ and $\mathcal{P}$ be the set of all unique patients who appeared on the waiting list at any time $t \in \mathcal{H}$. We construct a static bipartite graph where the nodes correspond to these patients and donors. An edge exists between a patient p and a donor d (having arrived at time t) if and only if they were compatible at that moment (*i.e.*, $p \in \mathcal{P}_{d,t}$). The weight of the edge is the utility $\mathcal{U}\left(d^{(t)}, p^{(t)}\right)$. The optimal allocation of transplants over $\mathcal{H}$ corresponds to a maximum weighted bipartite matching on this graph.

Any combinatorial algorithm for computing a maximum weight bipartite matching could be used to solve this problem. We make use of integer programming to compute the hindsight optimal matching. We refer to this program as the *omniscient algorithm*. A comprehensive description of the program can be found in Section E. Solving an instance using off-the-shelf solvers (*e.g.*, Gurobi) takes seconds.

We leverage the optimal patient-donor pairs, X^*, to construct a dataset of decisions for the imitation learning pipeline. The optimal utility also serves as a theoretical upper bound for evaluating online policies.

4.1.2 LEARNING POTENTIALS TO IMITATE THE OMNISCIENT ALGORITHM

We compile a dataset of optimal decisions from the omniscient algorithm consisting of tuples $\left(d^{(t)}, p^*, \mathcal{P}^{(t)}\right)$ for each $d^{(t)} \in \mathcal{D}$. The patient p^* is the patient chosen by the omniscient algorithm from the waitlist $\mathcal{P}^{(t)}$ for donor $d^{(t)}$. We omit any donors from the training set that the omniscient algorithm discards. For simplicity of presentation, we describe the approach as if we were learning from a single trajectory, though in practice we aggregate tuples from multiple trajectories.

Recall the potential-based policy

$$Q\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}; \theta\right) = \mathcal{U}\left(d^{(t)}, p^{(t)}\right) - P_\theta\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}\right).$$

To imitate the omniscient algorithm, we require that the score of the optimal patient p^* exceeds that of any other available candidate $p^{(t)}$,

$$Q\left(d^{(t)}, p^*, \mathcal{P}^{(t)}; \theta\right) > Q\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}; \theta\right) \tag{2}$$

for all pairs $\left(p^*, p^{(t)}\right)$ in which $p^{(t)} \in \mathcal{P}_{d,t} \setminus \{p^*\}$. Using this inequality, we present methods for learning potential functions.

Support vector machine (SVM) formulation We first present a method for learning linear potential functions using a *Support Vector Machine (SVM)* formulation. We frame the problem as finding the hyperplane that separates the optimal decision from suboptimal alternatives with a margin. We optimize θ by minimizing the pairwise hinge loss over all donors and compatible patients

$$\mathcal{L}_{\text{SVM}}(\theta) = \sum_{d^{(t)} \in \mathcal{D}} \sum_{p^{(t)} \in \mathcal{P}_{d,t} \setminus \{p^*\}} \max\left\{0, 1 - \left[Q\left(d^{(t)}, p^*, \mathcal{P}^{(t)}; \theta\right) - Q\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}; \theta\right)\right]\right\}. \tag{3}$$

Minimizing this convex objective finds a hyperplane θ that maximally separates the optimal patient from suboptimal alternatives. We provide a formal derivation of the SVM formulation in Section F, which is based on RankSVM (Joachims, 2002).

Pairwise regression We maximize the likelihood that the optimal patient is ranked higher than any individual alternative. This is equivalent to binary cross-entropy on the score differences,

$$\mathcal{L}_{\text{pair}}(\theta) = \sum_{d^{(t)} \in \mathcal{D}} \sum_{p^{(t)} \in \mathcal{P}_{d,t} \setminus \{p^*\}} \log \left[1 + \exp \left(Q\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}; \theta\right) - Q\left(d^{(t)}, p^*, \mathcal{P}^{(t)}; \theta\right) \right) \right]. \quad (4)$$

Listwise regression Alternatively, listwise regression considers the entire list of patients simultaneously. We minimize the *Kullback-Leibler (KL)* divergence between the predicted softmax distribution and the ground-truth distribution

$$\mathcal{L}_{\text{list}}(\theta) = \sum_{d^{(t)} \in \mathcal{D}} -\log \left[\frac{\exp\left(Q\left(d^{(t)}, p^*, \mathcal{P}^{(t)}; \theta\right)\right)}{\sum_{p^{(t)} \in \mathcal{P}_{d,t}} \exp\left(Q\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}; \theta\right)\right)} \right]. \quad (5)$$

In the linear case, pairwise regression is analogous to *logistic regression (LR)* on utility differences, while listwise regression is equivalent to multinomial logistic regression. The regression formulation allows us to extend the imitation learning framework to deep neural networks. Neural networks trade the direct interpretability of linear functions for greater expressiveness. We provide experiments of training with the different loss functions in Section K.4; it turned out that both loss functions are very effective, with the listwise objective performing slightly better in higher-dimensions.

4.2 Generating sound semi-synthetic data for learning

A major challenge in policy learning for organ allocation is data scarcity. The historical registry naturally contains only a single realized trajectory of events. Training strictly on a single historical realization risks overfitting. Furthermore, imitation learning is susceptible to *covariate shift*.

To mitigate this, we employ data augmentation. We construct a diverse set of semi-synthetic trajectories by bootstrapping the historical data. Our method randomizes the arrivals of new patients and donors while preserving the real clinical features and progressions. Only the relative timing and concurrent availability of patient-donor pairs are altered. The augmentation exposes the learner to a wider variety of trajectories of waitlist states and donor arrivals. We maintain that in expectation, the following match the real historical data: the distribution of donor features, the distribution of patient features, and the duration and trajectory of patients' disease progression on the waitlist. More details of our algorithm for generating such semi-synthetic trajectories including pseudocode and a visual example are provided in Section G. We also include analysis to prove that the semi-synthetic trajectory generation preserves the key statistical properties of the underlying population. We show in Section K.2 that training with semi-synthetic data significantly improves the learned policies.

5 Baseline allocation policies for comparison

We evaluate the potential-based policies against three allocation algorithms: the myopic utility maximizer, the current rule-based *status quo* heart allocation policy in the US, and the proposed *composite allocation score (CAS)*, the foundation of the continuous distribution system. In particular, CAS ranks patients using a linear combination of attributes including medical urgency, sensitization (CPRA), waiting time, and blood type compatibility (Papalexopoulos et al., 2024). Following public comment in 2024, the OPTN published preliminary patient features for CAS (OPTN, 2024). However, as of 2026, no weights have been proposed for the heart transplant CAS. To provide the strongest possible comparison, we optimize the weights for the CAS feature set using the potential learning framework from Section 4. In Section K.1 we show that learning weights for CAS via imitation learning substantially outperforms SMAC.

More information on all of these policies can be found in Section H. Table 1 summarizes the information used by the different policies as of material published in 2026. Only our potential-based policy explicitly accounts for the state of the waitlist and the direct utility simultaneously. In fact, any feature can be explicit in the decision logic depending on the potential function.

	Status quo	CAS	Myopic	Potential-based (ours)
Decision logic	Hierarchical rules	Linear score	Greedy utility	Utility - potential
Input features				
Patient status	Explicit	Explicit	Via utility	Explicit & via utility
Blood type	Explicit	Explicit	Via utility	Explicit & via utility
Location	Explicit	Explicit	Via utility	Explicit & via utility
Waiting time	Explicit	Explicit	Via utility	Explicit & via utility
Sensitization	Explicit	Explicit	Via utility	Explicit & via utility
Time on LVAD	Explicit	Explicit	Via utility	Explicit & via utility
Predicted LYG ($\mathcal{U}$)	—	—	Explicit	Explicit
Waitlist state	—	—	—	Explicit

Table 1: Comparison of information used in decision making across policies. "Explicit" indicates the feature is a direct component of the decision rule (*e.g.*, a score component). "Via utility" indicates the feature is used implicitly to calculate the estimated survival $\mathcal{U}$.

6 RESULTS

We use a discrete-event simulator initialized with historical data from UNOS dating back to 1987. The simulator models the complex dynamics of the allocation system, including donor arrivals, patient waitlist additions and removals (due to death or recovery), and the temporal progression of patient covariates (*e.g.*, updating lab results). We rely on Cox proportional hazards models (Cox, 1972) to estimate the life years gained. We employ a chronological split into a training horizon and a test horizon. We use the trajectory of events from the first quarter of 2019—January 1 to March 31—as the seed for training. As described, we augment this period by generating semi-synthetic trajectories for model training. We evaluate the policies on the unseen, real trajectory of the remaining nine months (April 1 to December 31). More details on the simulation environment can be found in Section I.

Main comparison of approaches We compare the potential-based policies against the baselines, and benchmark them also against the omniscient algorithm. We report the population life years gained (PLYG) of each policy in Table 2. For potential-based policies and CAS, which rely on optimization, we select the best-performing variation from the tuning phase across feature sets and loss functions. All ablations are presented in Section K. Table A9 provides the statistical significance of these results.

We observe that CAS offers a substantial improvement over the *status quo* policy, achieving an average of nearly 1,000 additional life years per month. CAS offers the flexibility to assign a granular score coupled with objective-aware weight optimization. However, our myopic baseline achieves a further improvement of ∼900 life years per month over CAS. These findings confirm that current allocation rules, and even proposed scoring systems like CAS, rely on features that are only proxies for survival outcomes.

The potential-based methods consistently further outperform the myopic baseline with statistical significance. Beyond SMAC, the imitation learning framework facilitates higher-utility allocations.

	Baselines			Linear pot.			Non-linear pot.		Upper bound
Month	Status quo	CAS	Myopic	SMAC	LR	SVM	NN (4)	NN (34)	Omniscient
Apr.	1332.56	2236.19	3119.34	3156.49	3123.28	3152.88	**3161.46**	3159.21	3328.70
May	1494.35	2503.84	3401.63	3413.65	3406.34	3423.36	3424.41	**3426.39**	3620.68
Jun.	1335.29	2293.97	3176.31	3195.48	3186.39	3202.31	3195.55	**3204.26**	3383.43
Jul.	1590.27	2470.58	3431.31	3412.50	**3442.54**	3441.79	3428.72	3441.19	3621.28
Aug.	954.02	2607.67	3707.53	3742.73	3708.29	3724.13	**3747.19**	3744.17	3937.68
Sep.	1632.99	2428.52	3380.64	3382.23	3390.62	3388.08	3389.69	**3397.23**	3590.33
Oct.	1772.68	2341.80	3184.58	3203.56	3186.07	**3209.87**	3208.22	3204.37	3377.67
Nov.	1705.20	2103.34	2867.00	2883.59	2864.91	2881.16	2882.68	**2886.60**	3011.73
Dec.	983.93	2198.27	3253.35	3237.31	3249.54	3247.10	3245.60	**3255.66**	3404.93
Mean PLYG	1422.37	2353.80	3280.19	3291.95	3284.22	3296.74	3298.17	**3302.12**	3475.16

Table 2: Monthly and mean PLYG of policies on real trajectories from April to December 2019. LR = logistic regression, SVM = support vector machine, NN (4) = 4-feature neural network, and NN (34) = 34-feature neural network.

Employing neural network potential functions achieves the strongest performance in 7 out of 9 test months, resulting in the highest mean utility. The myopic policy explicitly optimizes the primary objective, so it already captures the majority of the achievable utility. The remaining gap to the omniscient upper bound is therefore intrinsically small. Our learned potentials recover roughly 11% of the remaining gap between myopic matching and the omniscient upper bound. To put this into perspective, the difference is equivalent to dozens of additional successful transplants per year. We also observe a hierarchy in model performance. The neural network potential functions outperform the linear models. Furthermore, the 34-feature neural network outperforms the 4-feature variant. These results suggest that the state of the waitlist provides an important signal that allows the policy to adapt to the allocation dynamics.

Our best potential-based model achieves a competitive ratio of around 95% compared to the theoretical upper bound from omniscient matching, making our approach near-optimal on the real data distribution. It is unknown whether closing the remaining 5% gap is possible without omniscient knowledge. We further show in Section K.5 that the linear potentials are remarkably robust to temporal distribution shifts: potentials trained in 2019 perform nearly identically to those retrained on 2022 data when evaluated in 2022. The more expressive neural network potentials can be sensitive to distribution shift, but periodic retraining accommodates such shifts and recovers strong performance.

Fairness and alternate optimization objectives When optimizing the PLYG, we also analyze important secondary objectives, specifically mortality rates and transplant rates across blood types (Figures A3 and A4), with a complete discussion of these results in Section J. When disaggregated by blood type, significant disparities appear across all policies. Patients with blood type AB experience the highest transplant rates and the highest number of transplants per waiting year. The advantage is most pronounced under CAS, in which AB transplant rates spike and mortality rates drop disproportionately. Blood type O patients consistently face the lowest transplant rates and the highest mortality risks. The disparity is partly structural. Type O patients constitute the largest group on the waitlist but can only receive type O organs, creating more competition for donors. The potential-based policies achieve higher utility while simultaneously sacrificing less fairness. Yet even potential-based policies suffer from systemic disparities in access across different blood types.

Using our methodology, we can learn potentials using an alternate objective function that explicitly optimizes fairness. This is another strength of our approach. We show in Section J that when replacing the PLYG objective with a weighted sum of the (normalized) PLYG and (normalized) transplant rate for biologically disadvantaged patients, our neural network potential-based policy again achieves 96.5% of the utility of the omniscient algorithm and dominates the *status quo* and myopic policies across the two individual objectives. Our imitation learning approach can immediately generalize to such other utility functions.

7 Conclusions

In this work, we presented a computational framework for non-myopic policy optimization for dynamic matching problems based on potentials. Our methodology represents a shift from prior work on optimizing potentials, moving from inefficient black-box optimization to a scalable offline imitation learning approach. Our framework enabled us to train expressive high-dimensional potential functions, a capability that prior optimization methods could not achieve.

We applied our framework to heart transplant allocation. Our policies demonstrated significant improvements in population-level outcomes compared to myopic strategies, current and proposed US allocation systems, and prior potential-based approaches. Our policies are also near-optimal: they achieve 95% of the solution quality upper bound that comes from omniscient matching. Beyond maximizing PLYG, we demonstrated that our framework can optimize alternative utility functions, including fairness-aware objectives, without changing the underlying methodology.

Our analysis and methods are particularly timely as US policymakers review the implementation of continuous distribution for heart transplant allocation. Our findings suggested that the current linear scoring rules are overly restrictive and that substantial gains in population outcomes are possible through the more expressive, data-driven policies we proposed. Ultimately, we provided a scalable foundation for designing allocation mechanisms that make the most efficient use of scarce life-saving resources.

AI USE STATEMENT

In this work, we used generative AI tools for developing the software used in the experiments and editing the manuscript for clarity and consistency. We have not used generative AI tools to generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, assist with translation, clean and reformat datasets, support qualitative and thematic data analysis, or interpret results. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

We rely on historical data from the UNOS registry, which is imperfect. For example, it contains entry errors and missing values. While we implement preprocessing steps to impute missing entries and filter incongruent records, inherent data quality issues remain. In turn, the edge weights in our bipartite graph rely on imperfect predictive models. Determining patient outcomes, such as graft survival after a transplant, is a challenging predictive task. Unobserved confounders may influence actual outcomes, and systemic biases can be translated into the predictions. That being said, our potential learning framework is agnostic to how the weights are set. Therefore, future improvements in survival prediction and bias mitigation can be immediately combined with our algorithms.

We optimized our policies to maximize the population life years gained, one of many possible objectives for the application. However, as we show with the fairness-aware optimization, other objectives could also be optimized for, and we do not necessarily argue that utilitarian maximization is the sole or most appropriate metric for organ allocation.

REPRODUCIBILITY STATEMENT

We present comprehensive details necessary for reproducibility in the technical and code appendix. Hyperparameter settings, software libraries, and compute details can be found in Sections C and I. Algorithm and semi-synthetic data generation implementations are detailed in Sections E, G and H. The UNOS patient registry is not publicly available due to sensitive patient information.

REFERENCES

David J Abraham, Avrim Blum, and Tuomas Sandholm. Clearing algorithms for barter exchange markets: Enabling nationwide kidney exchanges. In *ACM Conference on Electronic Commerce (EC)*, 2007.

Gagan Aggarwal, Gagan Goel, Chinmay Karande, and Aranyak Mehta. Online vertex-weighted bipartite matching and single-bid budgeted allocations. In *Symposium on Discrete Algorithms (SODA)*, 2011.

Mohammad Ali Alomrani, Reza Moravej, and Elias Boutros Khalil. Deep policies for online bipartite matching: A reinforcement learning approach. *Transactions on Machine Learning Research*, 2022.

Ioannis Anagnostides, Zachary Sollie, Arman Kilic, and Tuomas Sandholm. Policy optimization for dynamic heart transplant allocation. *Circulation*, 152(Suppl_3):A4369427–A4369427, 2025.

Ioannis Anagnostides, Itai Zilberstein, Zachary W Sollie, Arman Kilic, and Tuomas Sandholm. Position: Machine learning for heart transplant allocation policy optimization should account for incentives. In *International Conference on Machine Learning (ICML)*, 2026.

Antonios Antoniadis, Themis Gouleakis, Pieter Kleer, and Pavel Kolev. Secretary and online matching problems with machine learned advice. In *Neural Information Processing Systems (NeurIPS)*, 2020.

Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5):469–483, 2009.

Henri Arno, Dennis Frauen, Emil Javurek, Thomas Demeester, and Stefan Feuerriegel. Rank-learner: Orthogonal ranking of treatment effects. In *International Conference on Machine Learning (ICML)*, 2026.

Itai Ashlagi and Alvin E Roth. Free riding and participation in large scale, multi-hospital kidney exchange. *Theoretical Economics*, 9(3):817–863, 2014.

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

Brian Ayers, Tuomas Sandholm, Igor Gosev, Sunil Prasad, and Arman Kilic. Using machine learning to improve survival prediction after heart transplantation. *Journal of Cardiac Surgery*, 36(11): 4113–4120, 2021.

Michael Bain and Claude Sammut. A framework for behavioural cloning. *Machine Intelligence 15*, pp. 103–129, 1995.

Santiago Balseiro, Haihao Lu, and Vahab Mirrokni. Dual mirror descent for online allocation problems. In *International Conference on Machine Learning (ICML)*, 2020.

Santiago Balseiro, Haihao Lu, and Vahab Mirrokni. Regularized online allocation problems: Fairness and beyond. In *International Conference on Machine Learning (ICML)*, 2021.

Jeroen Berrevoets, James Jordon, Ioana Bica, and Mihaela van der Schaar. OrganITE: Optimal transplant donor organ offering using an individual treatment effect. In *Neural Information Processing Systems (NeurIPS)*, 2020.

Jeroen Berrevoets, Ahmed M. Alaa, Zhaozhi Qian, James Jordon, Alexander E. S. Gimson, and Mihaela van der Schaar. Learning queueing policies for organ transplantation allocation using interpretable counterfactual survival analysis. In *International Conference on Machine Learning (ICML)*, 2021.

Dimitris Bertsimas, Vivek F Farias, and Nikolaos Trichakis. Fairness, efficiency, and flexibility in organ allocation for kidney transplantation. *Operations Research*, 61(1):73–87, 2013.

Avrim Blum and Paul Gölz. Incentive-compatible kidney exchange in a slightly semi-random model. In *ACM Conference on Economics and Computation (EC)*, 2021.

Abigail Breitfeld, Alberto Candela, Juan Delfa, Akseli Kangaslahti, Itai Zilberstein, Steve Chien, and David Wettergreen. Learning-based planning for improving science return of Earth observation satellites. In *International Symposium on Artificial Intelligence, Robotics and Automation in Space*, 2024.

Christopher JC Burges. From RankNet to LambdaRank to LambdaMART: An overview. *Learning*, 11(23-581):81, 2010.

Matteo Cameli, Maria Concetta Pastore, Alessandro Campora, Matteo Lisi, and Giulia Elena Mandoli. Donor shortage in heart transplantation: How can we overcome this challenge? *Frontiers in Cardiovascular Medicine*, 9:1001002, 2022.

Zhaohui Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: from pairwise approach to listwise approach. In *International Conference on Machine Learning (ICML)*, 2007.

Rebecca Cogswell, Ranjit John, Jerry D Estep, Sue Duval, Ryan J Tedford, Francis D Pagani, Cindy M Martin, and Mandeep R Mehra. An early investigation of outcomes with the new 2018 donor heart allocation system in the United States. *Journal of Heart and Lung Transplantation*, 39 (1):1–4, 2020.

David R Cox. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, 34(2):187–202, 1972.

B Cummiskey, K Lindblad, A Martinez, J Menteer, H Copeland, and R Daly. Understanding the transplant community's priorities in heart allocation. *American Journal of Transplantation*, 25(8): S582, 2025.

John P. Dickerson and Tuomas Sandholm. FutureMatch: Combining human value judgments and machine learning to match in dynamic environments. In *Conference on Artificial Intelligence (AAAI)*, 2015.

John P. Dickerson, Ariel D. Procaccia, and Tuomas Sandholm. Dynamic matching via weighted myopia with application to kidney exchange. In *Conference on Artificial Intelligence (AAAI)*, 2012.

John P Dickerson, David F Manlove, Benjamin Plaut, Tuomas Sandholm, and James Trimble. Position-indexed formulations for kidney exchange. In *ACM Conference on Economics and Computation (EC)*, 2016.

John P Dickerson, Ariel D Procaccia, and Tuomas Sandholm. Failure-aware kidney exchange. *Management Science*, 65(4):1768–1791, 2019. Early version in 2013 ACM Conference on Electronic Commerce (EC).

Jon Feldman, Nitish Korula, Vahab Mirrokni, Shanmugavelayutham Muthukrishnan, and Martin Pál. Online ad assignment with free disposal. In *International Workshop On Internet And Network Economics (WINE)*. Springer, 2009.

Dennis Frauen, Valentyn Melnychuk, Jonas Schweisthal, Mihaela van der Schaar, and Stefan Feuerriegel. Treatment effect estimation for optimal decision-making. *arXiv preprint arXiv:2505.13092*, 2025.

Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2026. URL https://www.gurobi.com.

Chen Hajaj, John Dickerson, Avinatan Hassidim, Tuomas Sandholm, and David Sarne. Strategy-proof and efficient kidney exchange using a credit mechanism. In *Conference on Artificial Intelligence (AAAI)*, 2015.

Zhiyi Huang, Zhihao Gavin Tang, and David Wajc. Online matching: A brief survey. *ACM SIGecom Exchanges*, 22(1):135–158, 2024.

Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *International Conference on Learning and Intelligent Optimization (LION)*, 2011.

Thorsten Joachims. Optimizing search engines using clickthrough data. In *ACM Conference on Knowledge Discovery and Data Mining (KDD)*, 2002.

Fahad Kamran, Maggie Makar, and Jenna Wiens. Learning to rank for optimal treatment allocation under resource constraints. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2024.

Richard M Karp, Umesh V Vazirani, and Vijay V Vazirani. An optimal algorithm for on-line bipartite matching. In *Symposium on Theory of Computing (STOC)*, 1990.

Süleyman Kerimov, Itai Ashlagi, and Itai Gurvich. Dynamic matching: Characterizing and achieving constant regret. *Management Science*, 70(5):2799–2822, 2024.

Süleyman Kerimov, Itai Ashlagi, and Itai Gurvich. On the optimality of greedy policies in dynamic matching. *Operations Research*, 73(1):560–582, 2025.

Arman Kilic, Michael A Mathier, Gavin W Hickey, Ibrahim Sultan, Victor O Morell, Suresh R Mulukutla, and Mary E Keebler. Evolving trends in adult heart transplant with the 2018 heart allocation policy change. *JAMA cardiology*, 6(2):159–167, 2021.

Changhee Lee, William Zame, Jinsung Yoon, and Mihaela van der Schaar. DeepHit: A deep learning approach to survival analysis with competing risks. In *Conference on Artificial Intelligence (AAAI)*, 2018.

Pengfei Li, Jianyi Yang, and Shaolei Ren. Learning for edge-weighted online bipartite matching with robustness guarantees. In *International Conference on Machine Learning (ICML)*, 2023.

Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. Adwords and generalized online matching. *Journal of the ACM (JACM)*, 54(5):22–es, 2007.

Johan Nilsson, Mattias Ohlsson, Peter Höglund, Björn Ekmehag, Bansi Koul, and Bodil Andersson. The international heart transplant survival algorithm (IHTSA): a new model to improve organ sharing and survival. *PloS one*, 10(3):e0118644, 2015.

OPTN. Update on continuous distribution of hearts. https://www.hrsa.gov/optn/policies-bylaws/public-comment/update-on-continuous-distribution-of-hearts, January 2024.

OPTN. Continuous distribution. https://www.hrsa.gov/optn/policies-bylaws/policy-issues/continuous-distribution, January 2025.

Takayuki Osa, Joni Pajarinen, Gerhard Neumann, J Andrew Bagnell, Pieter Abbeel, and Jan Peters. An algorithmic perspective on imitation learning. *Foundations and Trends® in Robotics*, 7(1-2): 1–179, 2018.

Theodore Papalexopoulos, James Alcorn, Dimitris Bertsimas, Rebecca Goff, Darren Stewart, and Nikolaos Trichakis. Reshaping national organ allocation policy. *Operations Research*, 72(4): 1475–1486, 2024.

David Parkes and Tuomas Sandholm. Optimize-and-dispatch architecture for expressive ad auctions. In *Workshop on Sponsored Search Auctions at the ACM Conference on Electronic Commerce (EC)*, 2005.

Michael A Rees, Jonathan E Kopke, Ronald P Pelletier, Dorry L Segev, Matthew E Rutter, Alfredo J Fabrega, Jeffrey Rogers, Oleh G Pankewycz, Janet Hiller, Alvin E Roth, Tuomas Sandholm, M. Utku Unver, and Robert A Montgomery. A nonsimultaneous, extended, altruistic-donor chain. *New England Journal of Medicine*, 360(11):1096–1101, 2009.

Matthew Richardson, Ewa Dominowska, and Robert Ragno. Predicting clicks: estimating the click-through rate for new ads. In *International Conference on World Wide Web (WWW)*, 2007.

Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.

Alvin E Roth, Tayfun Sönmez, and M Utku Ünver. Kidney exchange. *The Quarterly Journal of Economics*, 119(2):457–488, 2004.

Alexander Schrijver. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer, 2003.

Supriya Shore, Jessica R Golbus, Keith D Aaronson, and Brahmajee K Nallamothu. Changes in the United States adult heart allocation policy: challenges and opportunities. *Circulation: Cardiovascular Quality and Outcomes*, 13(10):e005795, 2020.

Xuanming Su and Stefanos Zenios. Patient choice in kidney allocation: The role of the queueing discipline. *Manufacturing & Service Operations Management*, 6(4):280–301, 2004.

Panos Toulis and David C Parkes. Design and analysis of multi-hospital kidney exchange mechanisms using random graphs. *Games and Economic Behavior*, 91:360–382, 2015.

Toon Vanderschueren, Wouter Verbeke, Felipe Moraes, and Hugo Manuel Proença. Metalearners for ranking treatment effects. *arXiv preprint arXiv:2405.02183*, 2024.

Chao Yu, Jiming Liu, Shamim Nemati, and Guosheng Yin. Reinforcement learning in healthcare: A survey. *ACM Computing Surveys (CSUR)*, 55(1):1–36, 2021.

Kevin C Zhang, Nikhil Narang, Carine Jasseron, Richard Dorent, Kevin A Lazenby, Mark N Belkin, Jonathan Grinstein, Anoop Mayampurath, Matthew M Churpek, Kiran K Khush, and William F Parker. Development and validation of a risk score predicting death without transplant in adult heart transplant candidates. *Journal of the American Medical Association (JAMA)*, 331(6):500–509, 2024.

Itai Zilberstein, Ioannis Anagnostides, Zachary W Sollie, Arman Kilic, and Tuomas Sandholm. Near-optimal dynamic matching via coarsening with application to heart transplantation. In *International Conference on Machine Learning (ICML)*, 2026.

A Additional related work

Organ allocation Research in data-driven organ allocation typically restricts optimization to simplified, interpretable policy classes. Early research proposed using techniques from queueing theory to maximize welfare in the context of kidney allocation (Su & Zenios, 2004). More recently, policies for kidney allocation were proposed based on a point system that ranks patients according to fixed priority criteria (Bertsimas et al., 2013). Similarly, the current *status quo* policy for adult heart transplants in the US utilizes a rigid six-tier hierarchy (Kilic et al., 2021). The system is widely criticized for failing to optimize for pretransplant and post-transplant outcomes (Shore et al., 2020). To address these limitations, the OPTN is reviewing transitioning to the *continuous distribution framework* (Papalexopoulos et al., 2024). Continuous distribution has already been deployed for the allocation of lungs, and is being considered for hearts, kidneys, livers, intestines, and pancreases (OPTN, 2025). That framework assigns a *composite allocation score (CAS)* to each patient, a linear combination of features such as medical urgency and geography. While CAS allows for more granularity than tiers, it ultimately reduces the allocation problem to optimizing a small, fixed set of weights. As of early 2026, these weights remain unfinalized for hearts, yet a preliminary feature set is available (Cummiskey et al., 2025; OPTN, 2024). In addition to evaluating our own potential-based policies, we apply our framework to derive optimized weights for CAS so we can compare our approach against theirs.

Parallel research focuses on improving the estimation of outcomes of possible matches rather than on optimizing the matching policy (Lee et al., 2018; Nilsson et al., 2015; Ayers et al., 2021). The *OrganITE* framework (Berrevoets et al., 2020) prioritizes matches based on estimated individual treatment effects and organ scarcity. Its successor, *OrganSync* (Berrevoets et al., 2021), groups organs into clustered types and employs priority queues for each type, with the goal of obtaining more robust counterfactual survival estimates. In contrast, our work focuses on the underlying matching logic. We utilize a simulator based on real UNOS data that is agnostic to the underlying predictive model of transplant outcomes. This ensures our policy optimization remains compatible as medical prognosis tools evolve.

Finally, it is worth noting that incentive compatibility has been studied in kidney exchange (*e.g.*, Blum & Gölz, 2021; Toulis & Parkes, 2015; Ashlagi & Roth, 2014; Hajaj et al., 2015), but that direction is orthogonal to our paper. Studying the myriad of incentive issues in the allocation of other organs such as hearts is an important direction for future work. Recent work highlights widespread incentive misalignment in the US heart transplant allocation system and asserts that some of these issues could be mitigated with better allocation policies (Anagnostides et al., 2026).

Imitation learning and learning-to-rank Our methods presented in Section 4 formalize potential function optimization as a form of offline imitation learning (Ross & Bagnell, 2010; Bain & Sammut, 1995). We construct an omniscient oracle that optimally solves historical matching problems in hindsight to generate training labels. Most prior work on imitation learning has been studied in the context of *Markov Decision Processes* for continuous control tasks, such as robotics, autonomous driving, or sensing (Argall et al., 2009; Osa et al., 2018; Breitfeld et al., 2024). Its application to discrete online matching is understudied and is a shift from prior work in dynamic allocation, which largely relies on direct policy search from simulation outcomes (Dickerson et al., 2012; Dickerson & Sandholm, 2015; Anagnostides et al., 2025). The success of reinforcement learning in the organ allocation domain is limited by sparse rewards, high variance, and the computational cost of simulating patient trajectories (Yu et al., 2021). By casting the problem as self-supervised learning from an oracle, we bypass these issues, achieving high sample efficiency and enabling the use of expressive, differentiable function approximators such as deep neural networks.

We formulate the potential function training as a form of learning-to-rank problem (Cao et al., 2007; Burges, 2010). In our formulation, the model is trained to minimize the divergence between its predicted rankings and the oracle's optimal node selection. Learning-to-rank is distinct from standard classification, which typically assumes fixed label sets or treats predictions independently. In organ

allocation, the candidate set is dynamic and of variable size. For each donor, the model must identify the optimal patient against a changing pool of alternatives. Learning-to-rank objectives explicitly model these within-list comparisons. Learning-to-rank has proven successful across domains ranging from healthcare resource allocation (Kamran et al., 2024; Vanderschueren et al., 2024; Frauen et al., 2025; Arno et al., 2026) to ad auctions (Richardson et al., 2007). Our work is the first to extend it to optimizing matching policies via potentials for sequential decision-making.

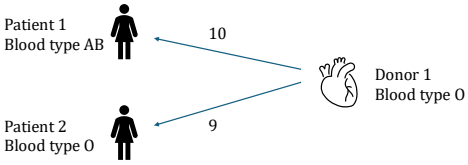
Online matching Online matching has seen great success in domains such as Internet advertising (Huang et al., 2024; Mehta et al., 2007; Richardson et al., 2007). Seminal work in online matching has focused on maximizing competitive ratios under adversarial arrival sequences (Karp et al., 1990). These algorithms provide strong theoretical guarantees, such as the optimal $1 - 1/e$ ratio. However, the assumption of an adversary is ill-suited for organ allocation, where patient and donor arrivals follow discernible biological and temporal patterns. It is worth pointing out that although dynamic matching involves an intrinsic tradeoff between short- and long-term value, there are conditions under which this tension is limited (Kerimov et al., 2024), and carefully designed greedy policies can be near-optimal (Kerimov et al., 2025).

Our work is somewhat more closely aligned with the literature on online *stochastic* matching, where arrivals are assumed to be drawn from known distributions (Feldman et al., 2009; Aggarwal et al., 2011). This includes the edge-weighted model (Feldman et al., 2009), which is analogous to an advertiser gaining specific revenue from a match, as well as the vertex-weighted model (Aggarwal et al., 2011). The edge-weighted stochastic matching model was recently adapted to heart transplant allocation, showing that under certain realistic yet simplifying assumptions, theoretically near-optimal policies are possible (Zilberstein et al., 2026). Data-driven approaches are commonly used to parameterize policies, such as in ad exchanges (Parkes & Sandholm, 2005) and dynamic kidney exchange (Awasthi & Sandholm, 2009; Dickerson et al., 2012; Dickerson & Sandholm, 2015). However, standard stochastic matching algorithms typically assume that the underlying bipartite graph or type distribution is known *ex ante*. These assumptions limit their applicability to the highly dynamic domain of organ allocation, where patients are continuously added to and removed from the waitlist due to medical progression, logistics, or death, creating a state space too complex for standard models.

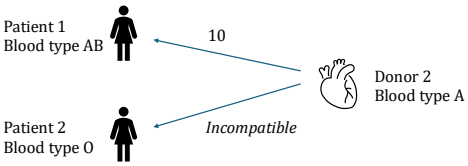
Another prominent direction in online matching leverages online primal-dual frameworks, where dual variables act as shadow prices to guide allocation. For example, dual mirror descent algorithms dynamically update thresholds to maximize efficiency (Balseiro et al., 2020). These approaches were extended to handle fairness constraints in regularized online allocation (Balseiro et al., 2021). Parallel research is the emerging field of learning-augmented algorithms, which integrate machine-learned advice into classical frameworks (Antoniadis et al., 2020; Li et al., 2023). Finally, deep reinforcement learning has been applied to learn matching policies directly from experience (Alomrani et al., 2022). Our work differentiates itself from these approaches by adopting an imitation learning strategy: instead of relying on trial-and-error or online gradient updates, we explicitly learn to mimic a hindsight-optimal oracle to guide the online policy.

B A MOTIVATING EXAMPLE

We motivate the need for non-myopic decision making with a simple example illustrated in Figure A1. Consider a waitlist with two patients: Patient 1 with blood type AB and Patient 2 with blood type O. At $t = 1$, a universal donor (*i.e.*, blood type O) arrives. Matching to Patient 1 yields a slightly higher immediate utility than matching to Patient 2. A myopic policy would prefer Patient 1. However, at $t = 2$, a type A donor arrives. Type A organs are compatible with Patient 1 but incompatible with Patient 2. If Patient 1 was matched at $t = 1$, the type A organ must be discarded, resulting in a total utility of 10. In contrast, an optimal policy recognizes that Patient 1 is easier to match and allocates the type O organ for Patient 2. This sequence yields a utility of 19. In fact, the utility of Patient 2 can be an arbitrarily small positive number, and the optimal matching would still be the same. Preserving scarce resources for hard-to-match patients motivates our use of *potential functions* to quantify the long-term value of patients on the waitlist.



(a) Matching at $t = 1$.



(b) Matching at $t = 2$.

Figure A1: Example motivating non-myopic decision making in heart transplant allocation. The edge weights denote the immediate utility of a patient-donor match. A myopic algorithm would obtain a suboptimal utility of 10 by matching Patient 1 with Donor 1. The optimal decisions are non-myopic: match Patient 2 with Donor 1 and Patient 1 with Donor 2.

C Further details on the potential functions

Linear potential functions The simplest form of potential function assumes that the potential of a patient is a linear combination of features:

$$P_\theta\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}\right) = \theta^T \phi\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}\right),$$

where ϕ is a feature mapping and θ is a learnable weight vector. Linear potentials offer high interpretability, but sacrifice expressiveness. All prior experiments on potential-based matching have relied exclusively on linear formulations (Dickerson et al., 2012; Dickerson & Sandholm, 2015; Anagnostides et al., 2025). We utilize a feature set ϕ that encodes a patient's blood (O, A, B, and AB) and location (one of the nine geographical regions of the mainland United States), capturing two primary drivers of organ scarcity.

Neural network-based potential functions To capture non-linear interactions between patient characteristics and the state of the waitlist, we learn potentials using a neural network, specifically a *Multi-Layer Perceptron (MLP)*. We design the MLP so that if $\theta = \vec{0}$, the potential is zero; this effectively enables regularization to ground the model closer to a myopic policy and better learn subtle deviations. In addition to a low-dimensional neural network that uses the patient's blood type as input, we employ a richer, 34-dimensional feature vector that captures both the characteristics of the specific match between the patient and donor and the broader context of the waitlist. Match features consist of a patient's blood type and location, the donor's blood type, and the predicted life years gained from the match. State features summarize the pool of waitlisted patients and the state of the allocation. These features include the fraction of waitlisted candidates with each blood type, the spatial distribution of the waitlist (fraction of patients in each region), the maximum and mean utility available in the current pool, and the donor arrival time. The input features are normalized using a standard scalar (zero mean, unit variance). Specific architectural details are provided in Table A1.

D Further details on black-box optimization

The black-box optimization approach has been used to optimize potentials in kidney exchange research (Dickerson et al., 2012; Anagnostides et al., 2025; Dickerson & Sandholm, 2015). As was done in that research, we will use SMAC (Hutter et al., 2011) for adjusting the potentials. SMAC treats the entire simulation and utility calculation as a black box function $f(\theta) \to \mathbb{R}$. It uses Bayesian optimization to model f and iteratively selects a candidate θ to evaluate.

While SMAC optimizes the true objective directly, it suffers from severe sample inefficiency and has no finite-time convergence or quality guarantees. Each function evaluation requires simulating an entire trajectory. Furthermore, SMAC cannot scale to high-dimensional parameter spaces, such as the weights of a neural network. We use SMAC to benchmark the linear potential functions and show that it scales poorly in higher dimensions (Section K.3).

Category	Hyperparameter	Value
Architecture	Hidden layers (4 features)	(64, 32)
	Hidden layers (34 features)	(128, 64, 32)
	Activation function	Leaky ReLU
	Leak coefficient	1×10^{-2}
	Output activation	Identity
Training	Optimizer	Adam
	Learning rate	1×10^{-4}
	Batch size	32
	Epochs	25
	Gradient clipping (norm)	1.0
	L2 regularization	1×10^{-3}
	Dropout rate	0.3

Table A1: Hyperparameters for neural network potential functions.

SMAC could also be used to minimize the loss functions of the imitation learning framework. However, SMAC is ill-suited for high-dimensional differentiable problems compared to gradient-based methods.

E Further details on the omniscient algorithm

We formulate the *omniscient algorithm* as an integer linear program with binary variables as follows. Let $x_{d,p} \in \{0, 1\}$ be a binary variable equal to 1 if patient p is matched with donor d, and 0 otherwise. Note that variables exist only for compatible patient-donor pairs.

$$\begin{aligned}
\text{maximize} \quad & \sum_{d \in \mathcal{D}} \sum_{p \in \mathcal{P}_d} \mathcal{U}(d, p) \cdot x_{d,p} \\
\text{subject to} \quad & \sum_{p \in \mathcal{P}_d} x_{d,p} \leq 1, && \forall d \in \mathcal{D} \\
& \sum_{d \in \mathcal{D}_p} x_{d,p} \leq 1, && \forall p \in \mathcal{P} \\
& x_{d,p} \in \{0, 1\} && \forall (d, p) \in \mathcal{E}
\end{aligned}$$

where $\mathcal{P}_d$ is the set of compatible patients for donor d, and $\mathcal{D}_p$ is the set of donors compatible with patient p, and $\mathcal{E}$ is the set of all viable matches.

The constraints enforce that each patient and donor is matched only once. The program outputs the optimal utility (according to Equation (1)) over $\mathcal{H}$ as well as the optimal patient-donor pairs, X^*.

A typical instantiation of the integer program for a month-long horizon contains tens to hundreds of thousands of decision variables (corresponding to patient-donor matches). However, since the problem reduces to finding a maximum weight matching in a bipartite graph, the constraint matrix is *totally unimodular* (Schrijver, 2003). Total unimodularity guarantees that the optimal solution to the linear programming relaxation is integral. We can then relax the integer constraints and solve the problem as a continuous linear program in polynomial time. An instance of this size can be solved optimally in seconds using an off-the-shelf linear program solver (*e.g.*, Gurobi). Combinatorial algorithms for maximum weight bipartite matching, such as the Hungarian algorithm, could alternatively be used to solve this problem.

F Derivation of the SVM formulation

Assuming a linear potential function and enforcing a margin of 1, we get from Inequality 2:

$$\mathcal{U}\left(d^{(t)}, p^*\right) - \theta^T \phi\left(d^{(t)}, p^*, \mathcal{P}^{(t)}\right) > \mathcal{U}\left(d^{(t)}, p^{(t)}\right) - \theta^T \phi\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}\right) + 1.$$

Rearranging terms:

$$\theta^T \left[\phi\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}\right) - \phi\left(d^{(t)}, p^*, \mathcal{P}^{(t)}\right) \right] > 1 - \left[\mathcal{U}\left(d^{(t)}, p^*\right) - \mathcal{U}\left(d^{(t)}, p^{(t)}\right) \right].$$

The target margin for the learnable component θ is dynamically adjusted based on the difference in utility between the optimal and suboptimal patient. The training vectors are the feature differences $\phi\left(d^{(t)}, p^{(t)}, \mathcal{P}^{(t)}\right) - \phi\left(d^{(t)}, p^*, \mathcal{P}^{(t)}\right)$.

G GENERATING SOUND SEMI-SYNTHETIC DATA FOR LEARNING

A major challenge in policy learning for organ allocation is data scarcity. The historical registry naturally contains only a single realized trajectory of events. Training strictly on a single historical realization risks overfitting. Furthermore, imitation learning is susceptible to *covariate shift*: a policy can be inaccurate when it encounters parts of the state space not covered during training Ross & Bagnell (2010).

To mitigate this, we employ data augmentation. We construct a diverse set of semi-synthetic trajectories by bootstrapping the historical data. The augmentation exposes the learner to a wider variety of trajectories of waitlist states and donor arrivals. We develop a method that ensures that the generated trajectories are statistically consistent with historical data in a medically meaningful way but differ in the relative timing of events so that multiple trajectories can be generated for training. Specifically, we want to maintain that in expectation, the following match the real historical data:

- the distribution of donor features,
- the distribution of patient features, and
- the duration and trajectory of patients' disease progression on the waitlist.

Our algorithm for generating such semi-synthetic trajectories is shown in Algorithm 1

ALGORITHM 1: Semi-synthetic trajectory generation

Input: Historical sets $\mathcal{D}_\mathcal{H}, \mathcal{P}_\mathcal{H}$ over horizon $\mathcal{H}$; New horizon $\mathcal{H}'$; Resampling bounds $D_{\min}, D_{\max}, P_{\min}, P_{\max}$.
Output: Semi-synthetic trajectory τ_{syn}

Initialize empty trajectory τ_{syn}
Sample volume $N_D \sim U(D_{\min}, D_{\max})$
Sample N_D donors with replacement from $\mathcal{D}_\mathcal{H}$
for *each sampled donor* d **do**
  Assign new arrival time $t \sim U(1, |\mathcal{H}'|)$
  Add $d^{(t)}$ to τ_{syn}
end
Sample volume $N_P \sim U(P_{\min}, P_{\max})$
Sample N_P patients with replacement from $\mathcal{P}_\mathcal{H}$
for *each sampled patient* p **do**
  if $p \in \mathcal{P}_0$ *(Waitlist initial set)* **then**
    Shift start time by offset $\delta \sim U(-k, k)$ up to $t = 0$
  else
    Assign new arrival time $t \sim U(1, |\mathcal{H}'|)$
  end
  Relativize all patient events (medical progression, death) to new start time
  Add p to τ_{syn}
end
return τ_{syn}

From a single historical horizon $\mathcal{H}$, we generate a large set of semi-synthetic trajectories over sub-horizons $\mathcal{H}'$. For instance, if $\mathcal{H}$ represents a three-month-long trajectory, we generate trajectories over one month. Let $\mathcal{D}_\mathcal{H}$ and $\mathcal{P}_\mathcal{H}$ be the sets of all unique donors and patients, respectively, who were

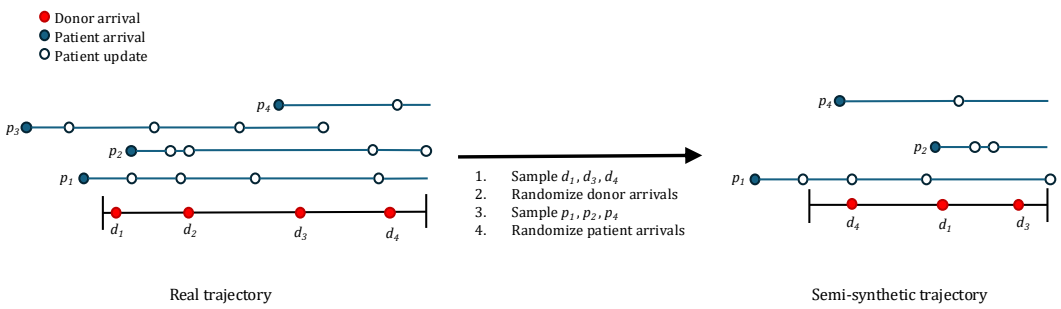


Figure A2: Generation of a semi-synthetic trajectory from a single real trajectory.

present during $\mathcal{H}$. Let $\mathcal{P}_0 \subset \mathcal{P}_\mathcal{H}$ be the subset of patients already on the waitlist at the start of $\mathcal{H}$ (*i.e.*, at $t = 0$). We compute $D_{\min}$ and $D_{\max}$ as the minimum and maximum number of donors observed in any historical window of length $|\mathcal{H}'|$ over $\mathcal{H}$. Similarly, let $P_{\min}$ and $P_{\max}$ be the lower and upper bounds on the number of unique patients on the waitlist. For each synthetic trajectory, we sample the total volume of donors N_D and patients N_P uniformly from these bounds:

$$N_D \sim U(D_{min}, D_{max}), \quad N_P \sim U(P_{min}, P_{max}).$$

We sample patients and donors uniformly with replacement from the historical population. We treat the real registry as a representative sample of the underlying clinical distribution, allowing us to generate an arbitrary number of independent training scenarios.

It is also important to distinguish between the initial waitlist and new arrivals. For new arrivals, we assign a random arrival time $t \sim U(1, |\mathcal{H}'|)$. For patients on the waitlist, their trajectories are altered by a small offset k up to $t = 0$. This preserves the realistic structure of the waitlist while slightly altering the starting state across trajectories. In all cases, a patient's internal progression (*e.g.*, time from arrival to death) is preserved relative to their new arrival time. We include a brief analysis in Section G.1 to prove that the semi-synthetic trajectory generation preserves the key statistical properties of the underlying population.

Figure A2 illustrates this process. Three patients and three donors are sampled from the real trajectory, and their introduction times are altered. However, the true patient progression is maintained. These new trajectories are semi-synthetic since the clinical features and progressions are real, but the relative timing and concurrent availability of patient-donor pairs are altered. The temporal shifts force learners to learn the underlying matching logic rather than memorizing historical timestamps. We show in Section K.2 that training with semi-synthetic data significantly improves the learned policies.

G.1 Properties of semi-synthetic trajectories

We provide a brief analysis to demonstrate that the semi-synthetic trajectory generation preserves the key statistical properties of the underlying population.

Proposition 1 (Volume consistency) *Let D' and P' be the random variables for the number of donors and patients in a semi-synthetic trajectory. The expected volumes are the midpoints of the observed historical range:*

$$\mathbb{E}[D'] = \frac{D_{\min} + D_{\max}}{2}, \quad \mathbb{E}[P'] = \frac{P_{\min} + P_{\max}}{2}. \tag{6}$$

The sampling explores the full range of observed donors and patients, ensuring the semi-synthetic trajectories capture both high-congestion and low-resource scenarios.

Proposition 2 (Feature invariance) *Let ϕ'_p be the feature vector of a randomly sampled patient in the synthetic trajectory (e.g., blood type, urgency, location). Let $\bar{\phi}_\mathcal{H}$ be the mean feature vector of the historical population $\mathcal{P}_\mathcal{H}$. Because patients are sampled uniformly at random, the expected feature vector satisfies:*

$$\mathbb{E}[\phi'_p] = \bar{\phi}_\mathcal{H}. \tag{7}$$

The semi-synthetic generation process randomizes the temporal aspect of patient arrivals, but preserves the representative distribution of patient features, including blood types and medical urgency.

Proposition 3 (Waitlist structure preservation) *Let P_0' be the random variable for the size of the initial waitlist at the start of $\mathcal{H}'$. The expected value of P_0' is:*

$$\mathbb{E}[P_0'] = \mathbb{E}[P'] \cdot \left(\frac{|\mathcal{P}_0|}{|\mathcal{P}_{\mathcal{H}}|} \right). \tag{8}$$

The ratio of pre-existing patients to new arrivals remains consistent, maintaining a realistic waitlist composition.

H Baseline allocation policies for comparison

We evaluate the potential-based policies against three allocation algorithms: the myopic utility maximizer, the current rule-based *status quo* heart allocation policy in the US, and the proposed *composite allocation score (CAS)*, the foundation of the continuous distribution system. We detail these algorithms below.

H.1 Myopic algorithm

The myopic algorithm serves as a baseline for greedy matching, maximizing the immediate life years gained without regard for future allocations. Formally, when a donor $d^{(t)}$ arrives, the algorithm selects:

$$p_{selected} = \underset{p^{(t)} \in \mathcal{P}_{d,t}}{\arg\max}\ \mathcal{U}\left(d^{(t)}, p^{(t)}\right).$$

All policies—except the *status quo* policy described next—will discard an organ if the utility of matching to $p_{selected}$ is less than or equal to zero (indicating a non-beneficial transplant) or the set of compatible patients is empty. The myopic baseline explicitly optimizes the primary objective via greedy matching. However, as illustrated in Figure A1, it is prone to suboptimal decision making in dynamic environments.

H.2 *Status quo* in the United States

The current allocation policy for adult heart transplants in the US is a hierarchical, rule-based system (Kilic et al., 2021). Patients are placed into six mutually exclusive tiers based on medical urgency status, blood type match, and geographic proximity to the donor. The medical status is a measure of the severity of the patient's condition. It is determined by factors such as sensitization levels (CPRA) and mechanical circulatory support status (*e.g.*, LVAD/ECMO). Patients within the same tier are sorted based on time on the waitlist. Priority is given to those who were listed earlier. Allocation follows a rigid lexicographic ordering. The *status quo* policy is agnostic to value. It does not directly optimize post-transplant survival or population outcomes. It is primarily designed to reduce waitlist mortality among the sickest patients, often at the expense of overall system efficiency.

The full description of the tiers is provided in Table A2. It comprises 68 priority tiers, where 1 represents the highest priority. Each tier is defined by a combination of i) medical status ranging from status 1 (highest urgency) to status 6; ii) blood type compatibility (where *secondary* blood compatibility applies only between type O donors and type A or type AB patients); and geographic proximity between donor and recipient location.

H.3 Continuous distribution and the composite allocation score (CAS)

The continuous distribution framework, currently under review by the OPTN, proposes assigning a CAS to each candidate. CAS ranks patients using a linear combination of attributes:

$$p_{selected} = \underset{p^{(t)} \in \mathcal{P}_{d,t}}{\arg\max}\ \theta^{\top} \phi_{CAS}\left(p^{(t)}\right).$$

Table A2: *Status quo* priority tiers.

Tier	Status	Blood match	Distance (nm)	Tier	Status	Blood match	Distance (nm)
1	1	Primary	≤ 500	35	2	Primary	≤ 2500
2	1	Secondary	≤ 500	36	2	Secondary	≤ 2500
3	2	Primary	≤ 500	37	3	Primary	≤ 2500
4	2	Secondary	≤ 500	38	3	Secondary	≤ 2500
5	3	Primary	≤ 250	39	4	Primary	≤ 1000
6	3	Secondary	≤ 250	40	4	Secondary	≤ 1000
7	1	Primary	≤ 1000	41	5	Primary	≤ 1000
8	1	Secondary	≤ 1000	42	5	Secondary	≤ 1000
9	2	Primary	≤ 1000	43	6	Primary	≤ 1000
10	2	Secondary	≤ 1000	44	6	Secondary	≤ 1000
11	4	Primary	≤ 250	45	1	Primary	Any
12	4	Secondary	≤ 250	46	1	Secondary	Any
13	3	Primary	≤ 500	47	2	Primary	Any
14	3	Secondary	≤ 500	48	2	Secondary	Any
15	5	Primary	≤ 250	49	3	Primary	Any
16	5	Secondary	≤ 250	50	3	Secondary	Any
17	3	Primary	≤ 1000	51	4	Primary	≤ 1500
18	3	Secondary	≤ 1000	52	4	Secondary	≤ 1500
19	6	Primary	≤ 250	53	5	Primary	≤ 1500
20	6	Secondary	≤ 250	54	5	Secondary	≤ 1500
21	1	Primary	≤ 1500	55	6	Primary	≤ 1500
22	1	Secondary	≤ 1500	56	6	Secondary	≤ 1500
23	2	Primary	≤ 1500	57	4	Primary	≤ 2500
24	2	Secondary	≤ 1500	58	4	Secondary	≤ 2500
25	3	Primary	≤ 1500	59	5	Primary	≤ 2500
26	3	Secondary	≤ 1500	60	5	Secondary	≤ 2500
27	4	Primary	≤ 500	61	6	Primary	≤ 2500
28	4	Secondary	≤ 500	62	6	Secondary	≤ 2500
29	5	Primary	≤ 500	63	4	Primary	Any
30	5	Secondary	≤ 500	64	4	Secondary	Any
31	6	Primary	≤ 500	65	5	Primary	Any
32	6	Secondary	≤ 500	66	5	Secondary	Any
33	1	Primary	≤ 2500	67	6	Primary	Any
34	1	Secondary	≤ 2500	68	6	Secondary	Any

The feature vector ϕ_{CAS} includes characteristics such as medical urgency, sensitization (CPRA), waiting time, and blood type compatibility (Papalexopoulos et al., 2024).

Following public comment in 2024, the OPTN published preliminary patient features for CAS (OPTN, 2024). However, as of early 2026, no weights have been proposed for the heart transplant CAS. To provide the strongest possible comparison, we optimize the weights for the CAS feature set using the potential learning framework from Section 4. We treat CAS as a restricted policy class: a linear ranking function that does not explicitly observe the predicted life years gained. In Section K.1 we show that learning weights for CAS via imitation learning substantially outperforms black-box optimization.

There are two main key distinctions between CAS and our potential-based policies:

1. CAS is a weighted linear sum of patient attribute values, whereas our method explicitly includes the immediate utility in decision making.

2. CAS values are patient-centric (except for the geographic distance from the donor to the patient) and static relative to other patients. Our potential-based policies can include state features, allowing the policy to adapt to the composition of the waitlist.

I Simulation Environment

All experiments are conducted on an M4 Pro processor with 24GB unified memory.

We rely on Cox proportional hazards models (Cox, 1972) to estimate the life years gained. Let $Y^T\left(d^{(t)}, p^{(t)}\right)$ denote the expected post-transplant survival (in years) for patient p receiving organ d at time t. Let $Y^W\left(p^{(t)}\right)$ denote the expected survival of patient p at time t if they do not receive a transplant and remain on the waitlist. The utility of a match is defined as the difference $Y^T\left(d^{(t)}, p^{(t)}\right) - Y^W\left(p^{(t)}\right)$.

The predictors are trained on the UNOS patient registry containing transplant outcomes dating back to 1987. This dataset contains over $60,055$ examples of transplant outcomes and $120,0282$ examples of waitlist survival times. For graft survival prediction, we utilize 120 covariates of the patient and donor, and for waitlist survival prediction, we utilize 30 covariates of the patient. While more complex models have been developed for survival prediction (Lee et al., 2018), their concordance is only marginally better than Cox regression. The simulator relies on these counterfactual predictors to compute patient outcomes. In fact, the simulator is agnostic to the survival model, allowing for modular upgrades. The simulator enforces compatibility constraints: a match is only feasible if the donor and patient have compatible blood types and are located within 1,000 nautical miles of each other. To ensure a uniform comparison, the distance constraint is applied to all policies, including the *status quo*.

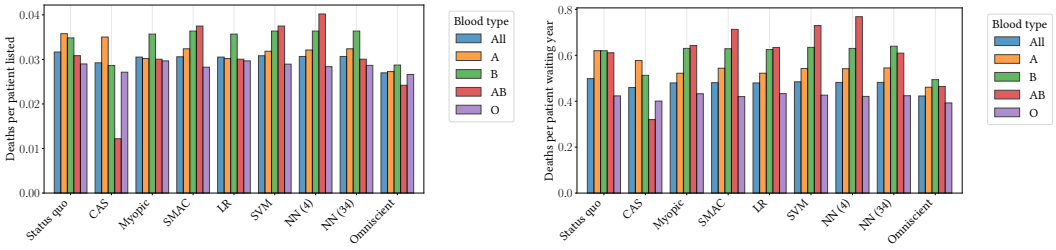
We employ a chronological split into a training horizon and a test horizon. We use the trajectory of events from the first quarter of 2019—January 1 to March 31—as the seed for training. As described in Section G, we augment this period by generating semi-synthetic trajectories for model training. We evaluate the policies on the unseen, real trajectory of the remaining nine months (April 1 to December 31). We report metrics strictly on this period without data augmentation, ensuring that our results reflect performance on real, historical dynamics. For the omniscient algorithm, we use the Gurobi Optimizer version 12.0.3 (Gurobi Optimization, LLC, 2026).

J Fairness and alternate optimization objectives

In this section, we first analyze the fairness of the policies using the PLYG objective for the optimization. We then show that our imitation learning method can directly optimize a fairness-aware objective, highlighting the generalizability of our approach.

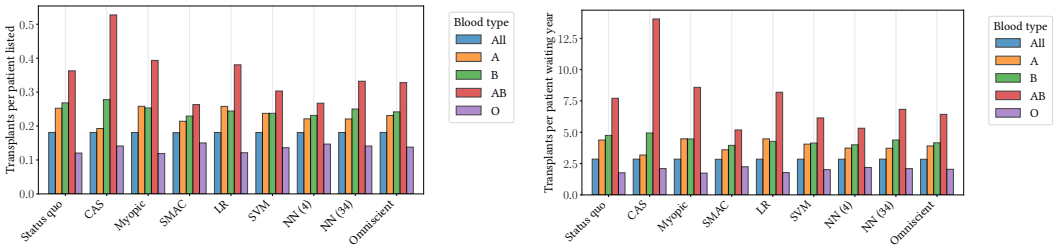
J.1 Fairness results from the main experiments

We report important secondary objectives from the experiments, specifically mortality rates and transplant rates across blood types, in Figures A3 and A4. For both metrics, we report rates normalized by the number of patients listed and the cumulative time spent waiting.



(a) Mortality rate per patient. (b) Mortality rate per wait year.

Figure A3: Average waitlist mortality rates normalized by number of patients in each group (a) and cumulative years on the waiting list per group (b).



(a) Transplant rate per patient. (b) Transplant rate per wait year.

Figure A4: Average transplant rates normalized by number of patients in each group (a) and cumulative years on the waiting list per group (b).

The policies compare similarly across the aggregate patient population. Interestingly, the omniscient algorithm yields the lowest mortality rate, indicating that maximizing PLYG correlates with reducing waitlist mortality. CAS successfully lowers mortality rates compared to the *status quo*, a primary goal of the continuous distribution system.

However, when disaggregated by blood type, significant disparities appear. Across all policies, patients with blood type AB experience the highest transplant rates and the highest number of transplants per waiting year. The advantage is most pronounced under CAS, in which AB transplant rates spike and mortality rates drop disproportionately. Blood type O patients consistently face the lowest transplant rates and the highest mortality risks. The disparity is partly structural. Type O patients constitute the largest group on the waitlist but can only receive type O organs, creating more competition for compatible donors.

The analysis of deaths per patient waiting year highlights the trade-offs in allocation. While CAS minimizes mortality for blood type AB, it does so at the cost of higher death rates for types A and B. Potential-based policies appear to smooth the extreme variances seen in CAS, resulting in a more equitable distribution of transplant rates among blood types A, B, and AB. The potential-based policies achieve higher utility while simultaneously sacrificing less fairness. Yet even potential-based policies suffer from systemic disparities in access across different blood types.

J.2 Fairness-aware optimization

Using our methodology, we can directly learn potentials that optimize a fairness-aware objective. We construct a hybrid utility function that is a weighted combination of the life years gained and the transplant rate for biologically disadvantaged patients. A biologically disadvantaged patient is

a patient with blood type O or cPRA above 0.5. We normalize each objective to the range $[0, 1]$ using upper and lower bounds from historical data and create an objective function by summing the two objectives with weight 0.5 on each. We use the same simulation setup in which the potential weights are learned using data from January-March 2019 and evaluated on April-December 2019 and the same architectures for the potential functions. We report the results in Table A3. Our neural network potentials policy again captures 96.5% of the omniscient utility in the fairness-aware setting, successfully balancing the two objectives and Pareto dominating the *status quo* and myopic policies. Our SVM potentials policy also Pareto dominates the two baseline policies, showing that our imitation learning approach can generalize to other utility functions beyond the PLYG.

	Baselines		Imitation learning		Upper bound
	Status quo	Myopic	SVM	NN (34)	Omniscient
Mean LYG per donor	4.53	10.21	**10.23**	10.21	10.80
Biologically disadvantaged transplant rate	58.5%	56.2%	59.0%	**60.4%**	60.5%
Normalized utility	0.307	0.455	0.479	**0.490**	0.508

Table A3: Comparison of fairness-aware optimization on April-December 2019.

K Ablation studies

We compare the potential-based policies across two axes: the complexity of the potential function (linear vs. neural network and dimensionality of input) and the optimization method (black-box vs. imitation learning).

We evaluate linear models with varying feature granularity, including a 4-dimensional feature vector encoding patient blood type and a 13-dimensional variation encoding blood type and location. These models are trained using three distinct optimizers: i) SMAC, ii) support vector machines (SVM), and iii) logistic regression (LR). For SMAC, we allocate a budget of 100 iterations. We observe that SMAC typically converges after approximately 50 iterations, suggesting it found a local maximum. Executing SMAC for this many iterations takes orders of magnitude more time compared to the imitation learning, which runs in minutes.

We evaluate two neural network architectures: a small network using only patient blood type as input (4 features), and a larger network using a 34-dimensional state vector. The neural networks are trained exclusively via the imitation learning framework (using listwise and pairwise regression), as SMAC is ill-suited for high-dimensional parameter spaces. The hyperparameters of the neural networks are detailed in Table A1.

K.1 Optimizing composite allocation score

We explore using both imitation learning and black-box optimization to learn the weights of CAS. We summarize the results in Table A4. As before, we learn the weights using the semi-synthetic trajectories from January-March 2019 and evaluate on April-December 2019. Using the imitation learning framework with both SVM and LR significantly outperforms SMAC. These results highlight the scalability limitations of black-box optimizers in high-dimensional spaces. We also see that the imitation learning framework is effective at learning general policies beyond potential-based ones.

	Black-box	Imitation learning	
	SMAC	SVM	LR
Mean PLYG	1688.22	2322.30	**2353.80**

Table A4: Comparison of mean PLYG per month (utility) between CAS with weights learned via black-box optimization and imitation learning methods.

K.2 Learning with semi-synthetic data

We evaluate the impact of data augmentation by comparing policies learned on the single real trajectory (January–March 2019) versus those trained on semi-synthetic variations. Table A5 reports the average difference in monthly utility between the augmented and non-augmented versions. A positive difference indicates that the semi-synthetic training yielded a higher utility. Across almost all policies, training on semi-synthetic data provides a substantial improvement. This benefit is most pronounced for CAS and the 34-feature neural networks, confirming that data augmentation effectively mitigates overfitting as the dimensionality and expressiveness of the policy increase.

Policy	Input dimension	Optimization method	Synthetic vs. real
CAS	14	SMAC	+135.58
	14	SVM	-30.99
	14	LR	+4.91
Linear potential	4	SMAC	+4.14
	4	SVM	+4.94
	4	LR	+4.23
Neural network potential	4	Pairwise loss	+3.01
	4	Listwise loss	+3.08
	34	Pairwise loss	+41.30
	34	Listwise loss	+9.91

Table A5: Difference in mean PLYG (utility) between algorithms trained on real versus semi-synthetic data. A positive difference indicates the semi-synthetic version performs better.

K.3 Distance in linear potential functions

Table A6 compares linear potential-based policies using the full 13-dimensional feature set (including location) versus a restricted 4-dimensional set (blood type only). Increasing model complexity does not universally improve performance; for SMAC and SVM, the additional dimensions degrade performance. SMAC significantly struggles as the search space expands. LR is the only method to successfully leverage the expanded feature set. These results suggest that patient location may act as a noisy signal in linear models, complicating optimization without providing a strong predictive signal for allocation efficiency.

	Black-box	Imitation learning	
	SMAC	SVM	LR
Difference in PLYG	-11.93	-6.82	+3.96

Table A6: Difference in mean PLYG per month between linear policies with 13 dimensions (location & blood type) versus 4 dimensions (blood type only). A positive difference favors the 13-dimensional model.

K.4 Neural network loss functions

Table A7 compares the performance of neural network policies trained with pairwise (Equation (4)) versus listwise (Equation (5)) regression losses. The 34-feature models generally outperform the 4-feature configurations. We hypothesize that the pairwise loss benefits the low-dimensional model by providing a denser signal through $O(N^2)$ comparisons, stabilizing learning when feature information is scarce. In contrast, the high-dimensional model has sufficient capacity to exploit the listwise loss, which directly optimizes the global ranking distribution.

K.5 Temporal generalization

We demonstrate that the potential-based policies can generalize to different temporal distributions either directly or through retraining on newer data. We evaluate the SVM and neural network models

	4 features		34 features	
Loss function	Pairwise	Listwise	Pairwise	Listwise
Mean PLYG	3298.17	3291.86	3299.97	3302.12

Table A7: Comparison of mean PLYG per month (utility) between potential-based policies using neural networks trained with weights learned via pairwise vs listwise loss.

trained in 2019 on data from April-December of 2022. We also compare against the same models but retrained on data from January-March of 2022. We show the results in Table A8. The linear potentials learned in 2019 produce strong results in 2022, demonstrating the robustness of these potentials. The neural network potentials, which may overfit more to the distribution it was trained on, does not perform as well in 2022, recording mean PLYG below the myopic policy. However, the neural network policy trained in 2022 again achieves the highest mean PLYG. This result suggests a very natural procedure: as new data becomes available (*e.g., each year*), retrain the potentials to reflect the current population.

			SVM		NN (34)		Upper bound
	Status quo	Myopic	2019	2022	2019	2022	Omniscient
Mean PLYG	1368.06	2824.11	2836.20	2834.55	2751.80	**2837.44**	2995.36

Table A8: Mean PLYG of policies on real trajectories from April to December 2022. We report variants of the SVM and NN(34) potential-based policies trained in 2019 and 2022.

L Omitted tables and figures

	Status quo	CAS	Myopic	SMAC	LR	SVM	NN (4)	NN (34)	Omniscient
Status quo	–	✓	✓	✓	✓	✓	✓	✓	✓
CAS		–	✓	✓	✓	✓	✓	✓	✓
Myopic			–	⋆	⋆	✓	✓	✓	✓
SMAC				–			✓	✓	✓
LR					–	⋆	⋆	✓	✓
SVM						–		⋆	✓
NN (4)							–		✓
NN (34)								–	✓

Table A9: Statistical significance of Wilcoxon paired signed-rank test for Table 2. Significance is based on monthly outcomes. A ✓ denotes statistical significance ($p < 0.05$) and ⋆ denotes a weaker significance ($0.05 < p < 0.10$).