Pipeline Interventions

Eshwar Ram Arunachaleswaran[*] Sampath Kannan[†] Aaron Roth[‡] Juba Ziani[§]

September 1, 2020

Abstract

We introduce the *pipeline intervention* problem, defined by a layered directed acyclic graph and a set of stochastic matrices governing transitions between successive layers. The graph is a stylized model for how people from different populations are presented opportunities, eventually leading to some reward. In our model, individuals are born into an initial position (i.e. some node in the first layer of the graph) according to a fixed probability distribution, and then stochastically progress through the graph according to the transition matrices, until they reach a node in the final layer of the graph; each node in the final layer has a *reward* associated with it. The pipeline intervention problem asks how to best make costly changes to the transition matrices governing people's stochastic transitions through the graph, subject to a budget constraint. We consider two objectives: social welfare maximization, and a fairness-motivated maximin objective that seeks to maximize the value to the population (starting node) with the *least* expected value. We consider two variants of the maximin objective that turn out to be distinct, depending on whether we demand a deterministic solution or allow randomization. For each objective, we give an efficient approximation algorithm (an additive FPTAS) for constant width networks. We also tightly characterize the "price of fairness" in our setting: the ratio between the highest achievable social welfare and the social welfare consistent with a maximin optimal solution. Finally we show that for polynomial width networks, even approximating the maximin objective to any constant factor is NP hard, even for networks with constant depth. This shows that the restriction on the width in our positive results is essential.

1 Introduction

Inequality can be difficult to correct by the time it manifests itself in consequential domains. For example, faculty in computer science departments are disproportionately male (Way et al. [2016]), and although the reasons for this are varied and complex, it seems difficult to correct *only* by intervening in the process of faculty hiring (although the solution likely involves some intervention at this stage). The problem is that interventions at the final stage of a long pipeline may not be enough (or the best way) to address iniquities that compound starting from earlier stages in the pipeline such as graduate school, college, high school, enrichment programs, all the way back to birth circumstances. Because each stage of, for example, employment pipelines feeds into the next, interventions that are isolated to any one stage can have difficulty controlling effects on final

[*] University of Pennsylvania. Email: eshwar@seas.upenn.edu.
[†] University of Pennsylvania. Email: kannan@cis.upenn.edu.
[‡] University of Pennsylvania. Email: aaroth@cis.upenn.edu.
[§] University of Pennsylvania. Email: jziani@seas.upenn.edu.

outcomes — and although in practice it is difficult to fully understand such a system, we would ideally like to design proposed interventions at a system-wide level, rather than myopically.

Thus motivated, we study an optimization problem within a stylized (and highly simplified) model of such a pipeline. Our model is a layered directed acyclic graph. The vertices in the first layer represent a coarse partitioning of possible birth circumstances into a small number of types — each vertex representing one of these types. There is a probability vector over these vertices and individuals are "born" into some vertex with these probabilities. The graph represents a Markov process that determines how individuals progress through the pipeline. From every vertex there is a stochastic transition matrix specifying the probability that an individual will progress to each vertex in the next layer of the pipeline. We might imagine, for example, that the proportion of children that enroll in each of several elementary schools (the second layer of such a pipeline) varies according to the neighborhood that they are raised in (the first layer). The proportion of children that then go on to enroll in each of several high schools may then vary according to the elementary school they attend, and so on. Finally, vertices at the last layer of the pipeline are associated with payoffs. One may then calculate the expected payoff of an individual as a function of their initial position. These payoffs may vary widely depending on this position.

We are concerned with the problem of how best to invest limited resources so as to *modify* the transition matrices governing different layers of this pipeline to achieve some goal. In the main body of the paper, we focus on a stylized model where the costs of modifying transition matrices are linear, for simplicity of exposition; we extend our results to more complex and realistic cost functions in the Appendix. We consider two goals: the first is simply maximizing social welfare — the expected payoff for an individual chosen according to the given probability vector for the first layer. Although this is a natural objective, it can easily lead to solutions that are "unfair" in the sense that they will prioritize investments that lead to improvements for majority populations over minority populations, simply because majority populations, by their sheer numbers, contribute more to social welfare. The second goal we study is therefore to maximize the *minimum* expected payoff of individuals, where the minimum is taken over all of the initial positions, i.e., layer 1 vertices. This "maximin" objective is a standard fairness-motivated objective in allocation problems [see, e.g., Barman and Krishnamurthy, 2017, Procaccia and Wang, 2014, Budish, 2011]. In fact, we study two different variants of this objective, that can be distinguished by the timing with which one wants to evaluate fairness. The *ex-ante* maximin objective asks for a *distribution* over budget-feasible modifications of the transition matrices, that maximize the minimum expected payoff over all initial positions. The *ex-post* maximin objective asks for a single (i.e. deterministic) budget-feasible modification to the transition matrices. Because the problem we study is non-convex, these two goals are distinct — which is preferred depends on *when* one wants to evaluate the fairness of a solution: before or after the randomization.

1.1 Overview of Our Results

Briefly, our main contributions are the following:

1. We define and formalize the *pipeline intervention problem* with the social welfare, ex-ante maximin, and ex-post maximin objectives. We also prove a separation between the ex-post and ex-ante maximin solutions.

2. We give an additive fully polynomial-time approximation scheme (FPTAS) for both the social welfare and ex-post maximin objectives for networks of constant width (but arbitrarily long

depth).

3. We give an efficient reduction from the ex-ante maximin objective problem to the ex-post maximin objective problem via equilibrium computation in two-player zero-sum games. Combined with our results from 2, this yields an additive FPTAS for the ex-ante maximin objective problem for constant width networks as well.

4. We define and prove tight bounds on the "price of fairness", which compares the optimal social welfare that can be achieved with a given budget to the social welfare of ex-post maximin optimal solutions.

5. Finally, we show that the pipeline intervention problem is NP hard even to approximate in the general case when the width w is not bounded — and hence that our efficient approximation algorithms cannot be extended to the general case (or even the case of constant depth, polynomial width networks).

1.2 Related Work

There is an enormous literature in "algorithmic fairness" that has emerged over the last several years, that we cannot exhaustively summarize here — but see Chouldechova and Roth [2018] for a recent survey. Most of this literature is focused on the myopic effects of a single intervention, but what is more conceptually related to our paper is work focusing on the longer-term effects of algorithmic interventions.

Dwork and Ilvento [2018] and Bower et al. [2017] study the effects of imposing fairness constraints on machine learning algorithms that might be composed together in various ways to reach an eventual outcome. They show that generally fairness constraints imposed on constituent algorithms in a pipeline or other composition do not guarantee that the same fairness constraints will hold on the entire mechanism as a whole. (They also study conditions under which fairness guarantees *are* well behaved under composition). Two recent papers (Liu et al. [2019a], Mouzannar et al. [2019]) study parametric models by which classification interventions in an earlier stage can have effects on the data distribution at later stages, and show that for many commonly studied fairness constraints, their effects can either be positive or negative in the long term, depending on the functional form of the relationship between classification decisions and changes in the agent type distribution.

There is also a substantial body of work studying game theoretic models for how interventions affect "fairness" goals. This work dates back to Coate and Loury [1993], Foster and Vohra [1992] in the economics literature, who propose game theoretic models to rationalize how unequal outcomes might emerge despite two populations being symmetrically situated. More recently, in the computer science literature, several papers consider more complicated models that are similar in spirit to Coate and Loury [1993], Foster and Vohra [1992]. Hu and Chen [2018] propose a two-stage model of a labor market with a "temporary" (i.e. internship) and "permanent" stage, and study the equilibrium effects of imposing a fairness constraint on the temporary stage. Liu et al. [2019b] consider a model of the labor market with higher dimensional signals, and study equilibrium effects of subsidy interventions which can lessen the cost of exerting effort. Kannan et al. [2019] study the effects of admissions policies on a two-stage model of education and employment, in which a downstream employer makes rational decisions. Jung et al. [2020] study a model of criminal justice in which crime rates are responsive to the classifiers used to determine criminal guilt, and study which fairness constraints are consistent with the goal of minimizing crime.

2 Model

The *pipeline intervention* problem is defined by a layered directed acyclic graph $G = (V, E)$, where V is the set of vertices (or nodes), and E is the set of edges. The vertices are partitioned into k *layers* $L_1, L_2, \cdots L_k$, each consisting of w vertices. We say that w is the *width* of the graph. For every $t \in [k-1]$, there is a directed edge from every $u \in L_t$ to every $v \in L_{t+1}$; the graph contains no other edge. In turn, every path from layer L_1 to layer L_k must go through exactly one vertex in each layer $L_2, \ldots, L_{k-1}$ in this order. Intuitively, such a layered graph represents a pipeline, in which individuals start at initial positions in layer 1, and transition through the graph to final positions in layer k, stochastically according to transition matrices which we define next. This layered model can be used to abstractly represent real-life pipelines; such a pipeline, that has received attention in previous work (e.g. Kannan et al. [2019]), is the education and job market one. Nodes in the initial layer represent a coarse partitioning of the population based on family income levels and educational background. The second layer could represent pre-K experience. For example, one could have 3 nodes in the second layer representing no pre-K, Headstart, and private pre-K. See for example Barnum [2019] for a general discussion of as well as pointers to recent studies on the efficacy of Headstart programs. At the next level or two, nodes can represent different qualities of K-12 schools, based on a coarse partitioning of their performance under one of several widely-available metrics, such as the ones provided by U.S. News [2020], Niche [2020].

The layer after that could be a coarse partitioning where nodes represent, for example, no college, technical or vocational school, and 2 and 4-year colleges coarsely grouped together based on perceived quality according to one of several college rankings. A subsequent layer could encode the details of a student's performance in college, such as their major and GPA, again under a coarse bucketing. The last layer, with numerical rewards could represent different types of employment with rewards determined by starting salaries and prospects for advancement.

In a more accurate model, we might perhaps condition the probability of transition from node u in layer i to node v in layer $i+1$ on the entire path taken by an individual leading up to node u. However, for mathematical tractability, we make the simplifying assumption that the process is Markovian, and this transition probability from u is independent of prior history.

Let $\mathcal{M}$ be the set of left stochastic matrices in $\mathbb{R}^{w \times w}$: i.e., $M \in \mathcal{M}$ if and only if for all $j \in [w]$, $\sum_{i \in [w]} M(i,j) = 1$, and for all $(i,j) \in [w]^2$, $M(i,j) \geq 0$. Let $\mathcal{D} \triangleq \{x \in [0,1]^w : \sum_{k=1}^{w} x(k) = 1\}$ be the set of probability distributions over $[w]$. An instance of the pipeline intervention problem is defined by three elements:

1. A set of initial transition matrices $M_t^0 \in \mathcal{M}$ between layers L_t and L_{t+1}, for all $t \in [k-1]$, such that for all $u \in L_t$, $v \in L_{t+1}$, $M_t^0(v,u)$ denotes the probability of transitioning from node u to node v. Note that we will multiply any input distribution to the right of any transition matrix we use in the paper.

2. An input distribution D_1 over the vertices in layer 1, where $D_1(u)$ denotes the fraction of the population that starts at u in L_1 as their initial position. Without loss of generality we assume $D_1(u) > 0$ for all initial positions $u \in L_1$.

3. Finally, a reward $R(v) \geq 0$ corresponding to each vertex $v \in L_k$ in the final layer. We let $R = (v)_{v \in L_k}^{\top}$ denote the vector of all rewards on layer k. We assume without loss of generality that the rewards on any two vertices in the final layer are distinct: for all

$v, v' \in L_k$, $R(v) \neq R(v')$. We can also assume without loss of generality (up to renaming) that $R(1) > \ldots > R(w)$.

In our model, each vertex u in the *starting layer* L_1 represents the initial position of some population; abusing notation, we refer to this population also as u. An individual in population u transitions to a node in layer L_2, then a node in layer L_3, up until they reach a node v in *destination layer* L_k, and obtains a reward of $R(v)$, with probability given by the transition matrices M_1^0 to M_{k-1}^0. The expected reward of an individual from population u is therefore given by $R^\top M_{k-1} \cdot \ldots \cdot M_1 e_u$, where e_u represents the w-dimensional standard basis vector corresponding to index u. The aim of the pipeline intervention problem is to *modify* the transition matrices between pairs of adjacent layers so as to improve these expected rewards in some way (we study several objectives) given a finite resource constraint.

We will take the point of view of a centralized designer, who can invest money into modifying the transition matrices between layers. We assume some edges can be modified, while some edges cannot; the edges that can be modified are called *malleable*, and the edges that cannot be modified are called *non-malleable*. We denote the set of malleable edges between layers t and $t+1$ by E_{mal}^t and the set of non-malleable edges by $\overline{E_{mal}^t}$ its complement. Further, we assume that modifying these transitions matrices comes at a cost, and that on a given layer t, the cost of transforming M_t^0 to some alternative $M_t \in \mathcal{M}$ is given by:

$$c(M_t, M_t^0) \triangleq \sum_{(i,j) \in [w]^2} \left| M_t(i,j) - M_t^0(i,j) \right|.$$

Remark 1. *A critique of such cost functions is that they may not be rich enough to model the cost of improving transitions and opportunities between different stages of, say, the education pipeline.*

To address this, we note that while we focus on these simple cost functions in the main body of the paper for simplicity of exposition, our algorithmic results (of Sections 4, 5 and 6) extend to more general and possibly more realistic cost functions — so long as they are convex and increase at least linearly as the distance $\sum_{(i,j) \in [w]^2} \left| M_t(i,j) - M_t^0(i,j) \right|$ *between modified transition matrix* M_t *and initial transition matrix* M_t^0 *increases. We discuss this extension in more detail in Appendix B.*

This extension allows us to model more realistic situations such as those where the cost functions are not linear, but also those where different edges have different costs — as motivated by the fact that real-life interventions often become more expensive the later they happen.

The designer has a total budget of B, and can select target transition matrices $(M_1, \ldots, M_{k-1})$ so long as the cost of modifying the initial transition matrices to his targets does not exceed his budget, and only malleable edges have been modified. That is, he must select target transition matrices subject to the constraint:

$$\sum_{t=1}^{k-1} c(M_t, M_t^0) \leq B.$$

We let

$$\begin{aligned}
&\mathcal{F}\left(B, M_1^0, \ldots, M_{k-1}^0\right) \\
&= \left\{ (M_1, \ldots, M_{k-1}) : \ \sum_{t=1}^{k-1} c\left(M_t, M_t^0\right) \leq B, \ M_t \in \mathcal{M} \ \forall t, \ M_t(i,j) = M_t^0(i,j) \ \forall (i,j) \in \overline{E_t^{mal}} \right\}
\end{aligned}$$

be the set of feasible sets of transition matrices, given initial matrices $M_1^0, \dots, M_{k-1}^0$ and budget B. We will consider several objectives that we may wish to optimize. The first is simply to maximize the overall social welfare (i.e. the expected reward of an individual chosen according to D_1), which is given by

$$W(M_1, \dots, M_{k-1}) \triangleq R^\top M_{k-1} \dots M_1 D_1.$$

The second objective aims to compute a "fair" outcome in the sense that it evaluates a solution according to the expected payoff of the worst-off members of society (here interpreted as individuals starting at the pessimal initial position), rather than according to the average. This is the classic maximin objective. It turns out that there are two distinct variants of this problem, depending on whether one wishes to allow randomized solutions (i.e. distributions over matrices) or not. We will elaborate on this distinction in the next section, but in the deterministic variant we wish to optimize

$$\min_{j \in [w]} R^\top M_{k-1} \dots M_1 e_j,$$

where $e_j \in \mathbb{R}^w$ is the unit vector with $e_j(j) = 1$, and $e_j(i) = 0$ for all $i \neq j$.

Remark 2. *We have assumed that each layer has* exactly *w vertices. In fact, all of our results generalize to the case in which each layer has $\leq w$ vertices.*

2.1 Optimization Problems of Interest

In this paper, we will provide algorithms to solve the following three optimization problems. We note at the outset that these optimization problems are non-convex, due to the fact that our objective values are not convex for $k \geq 2$. Hence we should not expect efficient algorithms in the fully general setting; we will give efficient algorithms for networks of constant width w (i.e. algorithms whose running time is polynomial in the depth of the network k), and show that outside of this class, the problem is NP hard even to approximate.

Social welfare maximization The first optimization problem we aim to solve is that of maximizing the social welfare of our network, under our budget constraint:

$$\begin{aligned} OPT_{SW} = \max_{M_1, \dots, M_{k-1}} \quad & R^\top M_{k-1} \dots M_1 D_1^0 \\ \text{s.t.} \quad & (M_1, \dots, M_{k-1}) \in \mathcal{F}\left(B, M_1^0, \dots, M_{k-1}^0\right) \end{aligned} \tag{1}$$

Ex-post maximin problem The second optimization problem aims to maximize the minimum expected reward that a population can obtain, where the minimum is taken over all initial positions:

$$\begin{aligned} OPT_{MM} = \max_{M_1, \dots, M_{k-1}} \quad & \min_{j \in [w]} R^\top M_{k-1} \dots M_1 e_j \\ \text{s.t.} \quad & (M_1, \dots, M_{k-1}) \in \mathcal{F}\left(B, M_1^0, \dots, M_{k-1}^0\right) \end{aligned} \tag{2}$$

Ex-ante maximin problem The third optimization problem has the same objective as Program 2, but allows randomization over sets of transition matrices that satisfy the budget constraint. Note that the budget constraint must be satisfied *ex-post*, for *any realization* of the set of transition matrices. To define this optimization problem, we let $\Delta\mathcal{F}\left(B, M_1^0, \ldots, M_{k-1}^0\right)$ the set of probability distributions with support $\mathcal{F}\left(B, M_1^0, \ldots, M_{k-1}^0\right)$. The optimization program is given by:

$$\begin{aligned} OPT_{RMM} = \max_{\Delta M} \quad & \min_{j\in[w]} R^\top \mathbb{E}_{M\sim\Delta M}\left[M_{k-1}\ldots M_1\right] e_j \\ \text{s.t.} \quad & \Delta M \in \Delta\mathcal{F}\left(B, M_1^0, \ldots, M_{k-1}^0\right), \end{aligned} \tag{3}$$

where the expectation is taken over the randomness of distribution ΔM. Note that where Program 3 can be viewed as optimizing an *ex-ante* notion of fairness, in which we are evaluated on the minimum expected value of individuals starting at any initial position, *before the coins of ΔM are flipped.* In contrast, Program 2 evaluates the minimum expected value of individuals starting at any initial position for an already established set of transition matrices.

Remark 3. *Programs (1), (2) and (3) all have solutions, and as such the use of maxima instead of suprema is well defined. To see this, first note that the feasible sets are non-empty since $\left(M_1^0, \ldots, M_{k-1}^0\right) \in \mathcal{F}\left(B, M_1^0, \ldots, M_{k-1}^0\right)$ for all $B \geq 0$. For Program (1), the existence of a maximum is an immediate consequence of the fact that the objective function is continuous in $(M_1, \ldots, M_{k-1})$ and $\mathcal{F}$ and $\mathcal{M}$ are compact sets. For Program (2), note that no solution can have $R^\top M_{k-1}\ldots M_1 e_j \geq \|R\|_\infty$ for any j, as $M_{k-1}\ldots M_1 e_j$ is a probability distribution. Hence, we can rewrite the program as*

$$\begin{aligned} \max_{v, M_1, \ldots, M_{k-1}} \quad & v \\ s.t. \quad & 0 \leq v \leq \|R\|_\infty, \\ & R^\top M_{k-1}\ldots M_1 e_j \geq v \;\forall j \in [w], \\ & (M_1, \ldots, M_{k-1}) \in \mathcal{F}\left(B, M_1^0, \ldots, M_{k-1}^0\right). \end{aligned}$$

This is an optimization problem with a continuous objective function over a compact set, so it admits a solution. A similar argument follows for Program (3).

3 Algorithmic Preliminaries

Our paper uses a dynamic programming approach for solving programs (1) and (2). (Our solution to program (3) is a game-theoretic reduction to our solution to program (2)). Our algorithms will search over possible input distributions in $\mathcal{D}$ starting from layer L_t for all $t \in \{2, \ldots, k-2\}$, and over possible ways of splitting the total budget B and allocating budget B_t to the transition from layer L_t to layer L_{t+1}, for all $t \in [k-1]$. To do so, we will need to discretize both the budget space $[0, B]$ and the probability space $\mathcal{D}$.

Cost of Discretizing the Budget To discretize the budget space, we define $\mathcal{B}(\varepsilon) = \{k\varepsilon,\ \forall k \in \mathcal{N}\}$ to be the set of numbers on the real line that are multiples of ε. We consider the following discretized version of Programs 1 and 2 (We do not need to explicitly consider Program (3), since our solution for this one will be a reduction to our solution to Program (2)):

$$
\begin{aligned}
OPT^{\varepsilon}_{SW} = \max_{M_1,\ldots,M_{k-1}} \quad & R^\top M_{k-1}\ldots M_1 D_1 \\
\text{s.t.} \quad & c(M_t, M_t^0) \le B_t \ \forall t \in [k-1] \\
& B_t \in \mathcal{B}(\varepsilon) \ \forall t \in [k-1], \ \sum_{t=1}^{k-1} B_t \le B \\
& M_t(i,j) = M_t^0(i,j) \ \forall (i,j) \in \overline{E_t^{mal}}, \ M_t \in \mathcal{M} \ \forall t
\end{aligned} \tag{4}
$$

and

$$
\begin{aligned}
OPT^{\varepsilon}_{MM} = \max_{M_1,\ldots,M_{k-1}} \quad & \min_{j\in[w]} R^\top M_{k-1}\ldots M_1 e_j \\
\text{s.t.} \quad & c(M_t, M_t^0) \le B_t \ \forall t \in [k-1] \\
& B_t \in \mathcal{B}(\varepsilon) \ \forall t \in [k-1], \ \sum_{t=1}^{k-1} B_t \le B \\
& M_t(i,j) = M_t^0(i,j) \ \forall (i,j) \in \overline{E_t^{mal}}, \ M_t \in \mathcal{M} \ \forall t.
\end{aligned} \tag{5}
$$

We show that this discretization does not affect the optimal value of our problems by much:

Claim 1. *There exists a feasible solution $\left(M_1^\varepsilon, \ldots, M_{k-1}^\varepsilon\right)$ to Program (4) (resp. Program (5)) with objective value at least $OPT_{SW} - (k-1)\varepsilon \left\|R\right\|_\infty$ (resp. $OPT_{MM} - (k-1)\varepsilon \left\|R\right\|_\infty$).*

We provide a brief proof sketch below, and defer the full proof to Appendix A.1.

Proof Sketch. We prove this result by constructing transition matrices M_t^ε that use roughly ε budget less than M_t^*. We show that we can do so so as to only lose welfare of the order of ϵ in each of the $k-1$ layer transitions we consider, and that this loss composes additively. □

Definition 1. *Let $K \subseteq \mathbb{R}^w$. We call a subset S of K an ε-net for K with respect to the ℓ_1-norm if and only if for every $D \in K$, there exists $D' \in S$ such that*

$$\|D - D'\|_1 \le \varepsilon.$$

Claim 2 (ε-nets in ℓ_1-distance for $\mathcal{D}$)**.** *Take $\varepsilon > 0$. There exists an ε-net $\mathcal{D}(\varepsilon)$ of $\mathcal{D}$ with respect to the ℓ_1-norm that has size $\left(\frac{1}{\varepsilon}\right)^w$.*

This is a standard proof, included in Appendix A.2 for completeness.

4 Social Welfare Maximization

We want to solve the following optimization problem:

$$
\begin{aligned}
\max_{M_1,\ldots,M_{k-1}} \quad & R^\top M_{k-1}\ldots M_1 D_1 \\
\text{s.t.} \quad & \sum_{t=1}^{k-1} c\left(M_t, M_t^0\right) \le B, \\
& M_t \in \mathcal{M} \ \forall t \in [k-1],
\end{aligned} \tag{6}
$$

4.1 A Dynamic Programming Algorithm for Social Welfare Maximization

In this section, we describe a dynamic programming algorithm for approximately solving the problem above on long skinny networks. The algorithm will run in polynomial time when the width w of the network is small; its running time is polynomial in the depth k of the network, but exponential in the width w. The formal description is given in Algorithm 1. Our algorithm works backwards, starting from the final transition matrix from layer L_{k-1} to L_k. It builds up the solutions to sub-problems parameterized by three parameters — a layer t, a starting distribution over the vertices in layer t, and a budget $B_{\geq t}$ that can be used at layers $\geq t$. For each sub-problem, it computes an approximately welfare-optimal solution. Once all of these sub-problems have been solved, the optimal solution to the original problem can be read off from the "sub-problem" in which $t = 1$, the starting distribution is the distribution on initial positions, and $B_{\geq 1} = B$. Here is the informal description of the algorithm:

1. For t going backwards from $k-1$ to 1, the algorithm does the following exploration over budget splits and probability distributions $D_t, D_{t+1} \in \mathcal{D}(\varepsilon)$ (an ε-net for the w-dimensional simplex in ℓ_1 norm) on L_t:

   (a) The algorithm explores all discretized splits of a budget $B_{\geq t}$ to be used for layers t to $k-1$ into a budget B_t to expend on layer t and a budget $B_{\geq t+1}$ to expend on the remaining layers $t+1$ to $k-1$, as well as all choices of target output probability distribution $D_{t+1} \in \mathcal{D}(\varepsilon)$ on layer L_{t+1} and the starting probability distribution $D_t \in \mathcal{D}(\varepsilon)$. Informally, we can think of these "target" and "initial" probability distributions as guesses for what the distribution on vertices in layer $t+1$ and layer t look like in the optimal solution. Recall that for each D_{t+1} and $B_{\geq t+1}$, our algorithm has already computed a near-optimal solution for a smaller sub-problem, which we will utilize in the next step.

   (b) The algorithm then finds a transition matrix from L_t to L_k that maximizes welfare when the starting distribution on layer t is D_t and the remaining transition matrices are fixed as in the solution to the corresponding sub-problem. Although the overall welfare-maximization problem is non-convex, this sub-problem can be solved as a linear program (Program 7) because all transition matrices except for one have been fixed as the solution to our sub-problem.

   (c) Finally, the algorithm picks and stores the recovered transition matrices from layer L_t to L_k that yield the highest reward, among all the transition matrices recovered from step 1b.

We remark that while (for notational simplicity) our algorithm is written as if all layers have size exactly w, it can easily be extended to the case in which all layers have size *at most* w.

We briefly note why Program (7) is a linear program. The objective is linear because only the matrix M_t represents variables. Thus we simply need to verify that the constraint on the cost is linear.

Definition 2. *We say that a transition matrix $M_t \in \mathcal{M}$ is feasible with respect to a budget split $B_{\geq t}, B_{\geq t+1}$ if and only if*

$$c\left(M_t, M_t^0\right) \leq B_{\geq t+1} - B_{\geq t}.$$

and $M_t(i,j) = M_t^0(i,j)$ for every non-malleable edge (i,j).

ALGORITHM 1: Dynamic Program for (Approximate) Social Welfare Maximization.

```
Input: Input distribution $D_1$, reward vector $R$, initial transition matrices $M_1^0, \dots, M_{k-1}^0$, budget $B$,
       discretization parameter $\varepsilon$.
Output: Transition $M(B^\varepsilon, D_1^0)$ from $L_1$ to $L_k$.
Initialization: Let $B_{\geq k} = 0$, $M(B_{\geq k}, D_k) = I$, $B^\varepsilon = \max\{x \in \mathcal{B}(\varepsilon) : \ x \leq B\}$.
for layer $t = k-1, \dots, 1$ do
    for all distributions $D_t \in \mathcal{D}(\varepsilon)$ if $t \neq 1$ ($D_t = D_1^0$ if $t = 1$) and budgets $B_{\geq t} \in \mathcal{B}(\varepsilon)$ with $B_{\geq t} \leq B$ do
        for all distributions $D_{t+1} \in \mathcal{D}(\varepsilon)$ and budgets $B_{\geq t+1} \leq B_{\geq t}$ such that $B_{\geq t+1} \in \mathcal{B}(\varepsilon)$ do
            Solve linear program

            $$\begin{aligned} M_t(B_{\geq t}, B_{\geq t+1}, D_t, D_{t+1}) = \arg\max_{M_t} \quad & R^\top M(B_{\geq t+1}, D_{t+1}) M_t D_t \\ \text{s.t.} \quad & c\left(M_t, M_t^0\right) \leq B_{\geq t} - B_{\geq t+1}, \\ & M_t(i,j) = M_t^0(i,j) \ \forall (i,j) \in \overline{E_t^{mal}} \\ & M_t \in \mathcal{M} \end{aligned} \qquad (7)$$

        end
        Pick $B_{\geq t+1}, D_{t+1}$ leading to the highest objective value in Program 7, and set
          $M(B_{\geq t}, D_t) = M(B_{\geq t+1}, D_{t+1}) M_t(B_{\geq t}, B_{\geq t+1}, D_t, D_{t+1})$.
    end
end
Return $M(B^\varepsilon, D_1)$.
```

Note that saying that M_t feasible with respect to $B_{\geq t}, B_{\geq t+1}$ is equivalent to saying that M_t is a feasible solution to Program (7) with parameters $B_{\geq t}, B_{\geq t+1}, D_t, D_{t+1}$ for any $D_t, D_{t+1} \in \mathcal{D}(\varepsilon)$. The constraint $c\left(M_t, M_t^0\right) \leq B_{\geq t+1} - B_{\geq t}$ can be equivalently replaced by $2w^2 + 1$ linear constraints. To do so, we introduce w^2 variables - $a_1, a_2, \cdots a_{w^2}$. The constraint can then be rewritten in the form $\sum_{i=1}^{w^2} |f_i| \leq B_{\geq t+1} - B_{\geq t}$, where each f_i is a linear combination of the variables. We can thus express the budget constraint of Program 7 by the following set of linear constraints:

1. $f_i \leq a_i \quad \forall i \in [w^2]$
2. $-f_i \leq a_i \quad \forall i \in [w^2]$
3. $\sum_{i=1}^{w^2} a_i \leq B_{\geq t+1} - B_{\geq t}$.

Thus, Program 7 can be written as a linear program with the number of constraints and variables being polynomial in w.

4.2 Running Time and Social Welfare Guarantees

We provide the running time and social welfare guarantees of Algorithm 1 below.

Theorem 1. *Algorithm 1 instantiated with discretization parameter ε yields a solution achieving social welfare at least $OPT - 3(k-1)\varepsilon \|R\|_\infty$, and has running time $O\left(k \frac{B}{\varepsilon} \left(\frac{1}{\varepsilon}\right)^{w^2} f(w)\right)$, where $f(w)$ is any upper-bound on the running time for solving linear Program 7, which is always polynomial in w.*

This immediately yields the following corollary:

Corollary 1. *Algorithm 1 with discretization parameter $\varepsilon' = \frac{\varepsilon}{3(k-1)}$ yields social welfare at least $OPT - \varepsilon\|R\|_\infty$, and has running time $O\left(k^2 \frac{B}{\varepsilon}\left(\frac{k}{\varepsilon}\right)^{w^2} f(w)\right)$, where $f(w)$ is any upper-bound on the running time for solving linear Program 7, which is always polynomial in w.*

We observe that this running time is polynomial in k (the depth of the network) and $1/\varepsilon$ (the inverse additive error tolerance), but exponential in w (the width of the network). Hence our algorithm runs in polynomial time for the class of constant width networks.

Remark 4. *We note that our additive near-optimality guarantee can be translated into a multiplicative guarantee. In the case where* all edges are malleable*, this follows from noting that given budget B, $OPT \geq \frac{B}{2w}\|R\|_\infty$: this can be reached by investing the totality of the budget into transitioning every node in the second-to-last layer to the highest reward node in the last layer, with probability $\frac{B}{2w}$ for each such node. Taking $\varepsilon = \delta \cdot \frac{B}{6(k-1)w}$ for some constant $\delta < 1$ gives a multiplicative approximation to the optimal social welfare with approximation factor $1 - \delta$.*

For the case in which non-malleable edges are allowed, a lower bound on OPT is given by $OPT \geq W_0$. Taking $\varepsilon = \delta \cdot \frac{W_0}{3(k-1)\|R\|_\infty}$ yields a multiplicative $1 - \delta$ approximation still.

Proof of Theorem 1 The proof of Theorem 1 relies on the following lemma, and its corollary:

Lemma 1. *Let $M \in \mathbb{R}^{w \times w}$ be a left stochastic matrix, and let $D, D' \in \mathcal{D}$ be probability distributions.*

$$\|MD - MD'\|_1 \leq \|D - D'\|_1.$$

Proof. Note that

$$\begin{aligned}
\|M(D - D')\|_1 = \sum_{i=1}^{w} \left|\left(M(D - D')\right)(i)\right| &= \sum_{i=1}^{w} \left|\sum_{j=1}^{w} M(i,j)(D(j) - D'(j))\right| \\
&\leq \sum_{i=1}^{w} \sum_{j=1}^{w} \left|M(i,j)(D(j) - D'(j))\right| \\
&= \sum_{j=1}^{w} \left|D(j) - D'(j)\right| \sum_{i=1}^{w} |M(i,j)| \\
&= \sum_{j=1}^{w} \left|D(j) - D'(j)\right| \\
&= \|D - D'\|_1,
\end{aligned}$$

where the inequality follows from the triangle inequality, and the second-to-last equality from the fact that

$$\sum_{i=1}^{w} |M(i,j)| = \sum_{i=1}^{w} M(i,j) = 1 \quad \forall j \in [w]$$

as M is a left stochastic matrix. □

Corollary 2. *Let $R \in \mathbb{R}^w$ be a real vector and $D, D' \in \mathcal{D}$ be probability distributions such that $\|D - D'\|_1 \leq \varepsilon$, and $M \in \mathbb{R}^{w \times w}$ a left stochastic matrix. Then*

$$R^\top MD \geq R^\top MD' - \|R\|_\infty \cdot \varepsilon.$$

Proof of Corollary 2. $\|R^\top M(D' - D)\|_1 \leq \|R\|_\infty \|M(D' - D)\|_1 \leq \|R\|_\infty \|D' - D\|_1 \leq \|R\|_\infty \cdot \varepsilon$, where the first step follows from Holder's inequality. □

We are now ready to prove Theorem 1:

Proof of Theorem 1. Let us denote by $B_1^\varepsilon, \ldots, B_{k-1}^\varepsilon$ a split of the budget for the discretized problem with $B_{\geq t}^\varepsilon = B_t^\varepsilon + \ldots + B_{k-1}^\varepsilon$. Let $M_1^\varepsilon, \ldots, M_{k-1}^\varepsilon$ a set of transition matrices achieving welfare $R^\top M_{k-1}^\varepsilon \ldots M_1^\varepsilon D_1^0 \geq OPT^\varepsilon \triangleq OPT - (k-1)\varepsilon \left\|R\right\|_\infty$ that is feasible with respect to budget split $B_1^\varepsilon, \ldots, B_{k-1}^\varepsilon$. Note that such a budget split and matrices exist by Claim 1. Let D_t^ε the probability distribution on layer t defined by these transition matrices, i.e.

$$D_t^\varepsilon = M_{t-1}^\varepsilon \ldots M_1^\varepsilon D_1^0.$$

To prove the result, we will show by induction that for all $B_{\geq t} \geq B_{\geq t}^\varepsilon$, and for $D_t \in \mathcal{D}(\varepsilon)$ such that $\|D_t - D_t^\varepsilon\|_1 \leq \varepsilon$,

$$R^\top M(B_{\geq t}, D_t) D_t \geq OPT^\varepsilon - 2(k-t)\varepsilon \|R\|_\infty.$$

This will directly imply that as B^ε is one of the possible values of $B_{\geq 1}$,

$$R^\top M(B^\varepsilon, D_1) D_1 \geq OPT^\varepsilon - 2(k-1)\varepsilon \left\|R\right\|_\infty.$$

Combined with Claim 1 that states $OPT_\varepsilon \geq OPT - (k-1)\varepsilon \left\|R\right\|_\infty$, we will obtain the result.

Let us now provide our inductive proof. First, consider the transition from layer L_{k-1} to layer L_k. Note that

$$OPT^\varepsilon \leq R^\top M_{k-1}^\varepsilon \ldots M_1^\varepsilon D_1^0 = R^\top M_{k-1}^\varepsilon D_{k-1}^\varepsilon.$$

Let $D_{k-1} \in \mathcal{D}(\varepsilon)$ be such that $\|D_{k-1} - D_{k-1}^\varepsilon\| \leq \varepsilon$. Note then that by Corollary 2,

$$R^\top M_{k-1}^\varepsilon D_{k-1} \geq R^\top M_{k-1}^\varepsilon D_{k-1}^\varepsilon - \varepsilon \|R\|_\infty.$$

Further, M_{k-1}^ε is feasible for Program (7) with respect to $B_{\geq k-1}, B_{\geq k} = 0$, given $B_{\geq k-1} \geq B_{\geq k-1}^\varepsilon$. As such, for $B_{\geq k-1} \geq B_{\geq k-1}^\varepsilon$, we have that

$$R^\top M(B_{\geq k-1}, D_{k-1}) D_{k-1} \geq R^\top M_{k-1}^\varepsilon D_{k-1},$$

and in turn

$$R^\top M(B_{\geq k-1}, D_{k-1}) D_{k-1} \geq OPT^\varepsilon - \varepsilon \|R\|_\infty.$$

Now, suppose the induction hypothesis holds at layer $t+1$. I.e., for all $B_{\geq t+1} \geq B_{\geq t+1}^\varepsilon$, for $D_{t+1} \in \mathcal{D}(\varepsilon)$ such that $\|D_{t+1} - D_{t+1}^\varepsilon\|_1 \leq \varepsilon$,

$$R^\top M(B_{\geq t+1}, D_{t+1}) D_{t+1} \geq OPT^\varepsilon - 2(k-t-1)\varepsilon \|R\|_\infty.$$

For any $B_{\geq t} \geq B_{\geq t}^\varepsilon$, note that one can set $B_{\geq t+1} = B_{\geq t+1}^\varepsilon$ and $B_t \geq B_t^\varepsilon$; hence, M_t^ε is feasible for Program (7) with respect to $B_t \geq B_t^\varepsilon, B_{\geq t+1}^\varepsilon$. Since $\|D_t - D_t^\varepsilon\|_1 \leq \varepsilon$ and $\|D_{t+1} - M_t^\varepsilon D_t^\varepsilon\|_1 \leq \varepsilon$, we have that by Corollary 2,

$$R^\top M(B_{\geq t+1}^\varepsilon, D_{t+1}) M_t^\varepsilon D_t \geq R^\top M(B_{\geq t+1}^\varepsilon, D_{t+1}) M_t^\varepsilon D_t^\varepsilon - \varepsilon \|R\|_\infty \geq R^\top M(B_{\geq t+1}^\varepsilon, D_{t+1}) D_{t+1} - 2\varepsilon \|R\|_\infty.$$

Using the induction hypothesis, we obtain that

$$R^\top M(B^\varepsilon_{\geq t+1}, D_{t+1}) M^\varepsilon_t D_t \geq OPT^\varepsilon - 2(k-t)\varepsilon \|R\|_\infty.$$

In particular,

$$R^\top M(B_{\geq t}, D_t) D_t \geq OPT^\varepsilon - 2(k-t)\varepsilon \|R\|_\infty,$$

which concludes the proof of the social welfare guarantee. For the running time, we note that at each time step t, we solve one instance of Program 7 for each of the (at most) $\frac{B}{\varepsilon}$ possible budget splits of $B_{\geq t}$ and for each of the $\left(\frac{1}{\varepsilon}\right)^w$ (by Claim 2) probability distributions in $\mathcal{D}(\varepsilon)$ in layer L_t and layer L_{t+1}; i.e., for each t, the algorithm solves $O\left(\frac{B}{\varepsilon}\left(\frac{1}{\varepsilon}\right)^{w^2}\right)$ optimization programs. Then, the algorithm finds the solution of all of these programs with the best objective value, which can be done in time linear in the number of such solutions, i.e. $O\left(\frac{B}{\varepsilon}\left(\frac{1}{\varepsilon}\right)^{w^2}\right)$. This is repeated for $k-1$ values of t. □

5 (Ex-post) Maximin Value Maximization

Although social welfare maximization is a natural objective, it is well-known that it can be "unfair" in the sense that it explicitly prioritizes the welfare of larger populations (here represented as initial positions that have larger probability mass) over smaller populations. We can alternately evaluate a solution according to the welfare of the *least-well-off* population (here represented by the initial position with the smallest expected value) and ask to optimize *that* objective. We show how to optimize this objective in this section, when one demands a deterministic solution.

5.1 A Dynamic Programming Algorithm for Computing an Ex-post Maximin Allocation

Algorithm and proof: In this subsection, we adapt the dynamic programming approach in Section 4.1 to give an approximation algorithm for the problem of maximizing the minimum expected reward over all initial positions. Recall that $\mathcal{D}$, the probability simplex, denotes the set of all possible probability distributions on a layer. Intuitively, our algorithm for maximizing social welfare kept track of a single probability distribution in each subproblem: the overall probability of arriving at each vertex in the layer *over both the randomness of an individual's initial position, and the randomness of the transition matrix*. In order to optimize the *minimum* expected value over all initial positions, we will need to keep track of more state. At every layer L_t, we will keep track of the probability of reaching each vertex in that layer *from each initial position in the starting layer*. So, we will now keep track of collections of w probability distributions in $\mathcal{D}^w$, one for each starting position. We call the elements of $\mathcal{D}^w$ *population-wise distributions*.

We introduce a discretization $\mathcal{A}(\varepsilon)$ of $\mathcal{D}^w$, as follows: $\mathcal{A}(\varepsilon) \triangleq (\mathcal{D}(\varepsilon))^w$, where $\mathcal{D}(\varepsilon)$ denotes a ε-net of $\mathcal{D}$ (of size $\left(\frac{1}{\varepsilon}\right)^w$). Given a population-wise probability distribution $A_t \in \mathcal{A}(\varepsilon)$ at layer t, we write A^j_t for the probability distribution corresponding to population j. The algorithm works as follows, just as before, running backwards from the final layer to the first layer:

1. For t going backwards from $k-1$ to 1, the algorithm does the following, for every population-wise distribution $A_t \in \mathcal{A}(\varepsilon)$ on L_t:

(a) The algorithm explores all splits of the budget $B_{\geq t}$ for layers t to k into a budget B_t for the transition from L_t to L_{t+1} and a budget $B_{\geq t+1}$ for L_{t+1} to L_k, as well as all choices of output population-wise probability distributions $A_{t+1} \in \mathcal{A}(\varepsilon)$ on layer L_{t+1}.

(b) The algorithm then finds a near-optimal transition matrix from L_t to L_k for every budget decomposition, by using the previously computed near-optimal solution for layers L_{t+1} to L_k, and solving Program 8. The program maximizes the *minimum reward obtained from any initial position*, assuming the population-wise distribution of individuals at layer L_t is given by $\mathcal{A}_t$.

(c) Finally, the algorithm picks and stores the best recovered transition matrices from layer L_t to L_k that yield the highest reward, among all the transition matrices recovered from step 1b.

The input population-wise probability distribution $A_1 \in \mathcal{D}^w$ on the first layer is defined in the following manner, $A_1^j := e_j$ (the j-th basis vector in the usual orthonormal basis of $\mathbb{R}^w$) for all $j \in [w]$.

ALGORITHM 2: Dynamic Program for (Approximate) Maximin Value.

```
Input: Reward vector $R$ on layer $L_k$, initial transition matrices $M_1^0, \dots, M_{k-1}^0$, budget $B$, discretization
       parameter $\varepsilon$.
Output: Transition $M(B^\varepsilon, A_1)$ from $L_1$ to $L_k$.
Initialization: Let $B_{\geq k} = 0$, $M(B_{\geq k}, A_k) = I_{w \times w}$, $B^\varepsilon = \max\{x \in \mathcal{B}(\varepsilon) : \ x \leq B\}$.
for layer $t = k-1, \dots, 1$ do
    for all distributions $A_t \in \mathcal{A}(\varepsilon)$ if $t \neq 1$ (resp. for $A_t = A_1$ if $t = 1$), budget $B_{\geq t} \in \mathcal{B}(\varepsilon)$ with $B_{\geq t} \leq B$ do
        for all distributions $A_{t+1} \in \mathcal{A}(\varepsilon)$, budget $B_{\geq t+1} \leq B_{\geq t}$ such that $B_{\geq t+1} \in \mathcal{B}(\varepsilon)$ do
            Solve
            $$\begin{aligned} M_t(B_{\geq t}, B_{\geq t+1}, A_t, A_{t+1}) = \arg\max_{M_t} \quad & \min_{j \in [w]} R^\top M(B_{\geq t+1}, A_{t+1}) M_t A_t^j \\ \text{s.t.} \quad & c\left(M_t, M_t^0\right) \leq B_{\geq t} - B_{\geq t+1} \ \forall t \in [k-1], \\ & M_t(i,j) = M_t^0(i,j) \ \forall (i,j) \in \overline{E_t^{mal}} \\ & M_t \in \mathcal{M}. \end{aligned} \qquad (8)$$
        end
        Pick $B_{\geq t+1}, A_{t+1}$ with the best objective value in Program 8, and let
            $$M(B_{\geq t}, A_t) \triangleq M(B_{\geq t+1}, A_{t+1}) M_t(B_{\geq t}, B_{\geq t+1}, A_t, A_{t+1}).$$
    end
end
return $M(B^\varepsilon, A_1)$
```

Note that Program (8) can be written as a linear program of size polynomial in w, using the same method that was employed to write Program (7) as a linear program.

5.2 Running Time and Ex-Post Maximin Value Guarantees

Remember that we let OPT_{MM} denote the maximin value value of the given network. The running time and accuracy guarantees of Algorithm 2 are provided below:

Theorem 2. *Algorithm 2 with discretization parameter ε yields maximin value at least $OPT_{MM} - 3(k-1)\varepsilon\|R\|_\infty$, and has running time $O\left(k\frac{B}{\varepsilon}\left(\frac{1}{\varepsilon}\right)^{w^4} g(w)\right)$, where $g(w)$ is any upper-bound on the running time for solving linear Program 8, which is always polynomial in w.*

This immediately induces the following corollary:

Corollary 3. *Algorithm 2 with discretization parameter $\varepsilon' = \frac{\varepsilon}{3(k-1)}$ yields maximin value at least $OPT_{MM} - \varepsilon\|R\|_\infty$, and has running time $O\left(k^2\frac{B}{\varepsilon}\left(\frac{k}{\varepsilon}\right)^{w^4} g(w)\right)$, where $g(w)$ is a polynomial upper-bound on the running time of linear Program 8.*

The proof of Theorem 2 is almost identical to that of Theorem 1. We provide the full proof in Appendix D.1.

6 (Ex-ante) Maximin Value Maximization

In this section, we consider the problem of optimizing the *ex-ante* minimum expected value over all initial positions: in other words, we allow ourselves to find a *distribution* over solutions, and take expectations over the randomness of this distribution, solving:

$$\begin{aligned} \max_{\Delta M} \quad & \min_{j\in[w]} R^\top \mathbb{E}_{M\sim\Delta M}\left[M_{k-1}\dots M_1\right] e_j \\ \text{s.t.} \quad & \Delta M \in \Delta\mathcal{F}\left(B, M_1^0, \dots, M_{k-1}^0\right) \end{aligned} \tag{9}$$

We show in Appendix C that this can yield strictly higher utility than optimizing the ex-post minimum value. We then give an algorithm for solving the ex-ante problem by exhibiting a game theoretic reduction to the ex-post problem.

6.1 Solving the Ex-ante Maximization Problem Using Algorithm 1

Because Program 3 is a max min problem over a polytope, we can view it as a zero-sum game, and the solution that we want corresponds to a maxmin equilibrium strategy of this game. As first shown by Freund and Schapire [1996], it is possible to compute an approximate equilibrium of a zero-sum game if we can implement a *no-regret* learning algorithm for one of the players, and an approximate best-response algorithm for the other player: if we simply simulate repeated play of the game between a no-regret player and a best-response player, then the empirical average of player actions in this simulation converges to the Nash equilibrium of the game.

This forms the basis of our algorithm. One player plays the "multiplicative weights" algorithm over the initial positions in layer 1 of the graph. This induces at every round a distribution over initial positions. The best response problem, which must be solved by the other player, corresponds to solving a welfare-maximization problem given the distribution over initial positions represented by the multiplicative weights distribution. Fortunately, this is exactly the problem that we have already given a dynamic programming solution for. The solution in the end corresponds to the uniform distribution over the solutions computed by the best-response player over the course of the dynamics. The algorithm is formally described below:

ALGORITHM 3: 2-Player Dynamics for the Ex-Ante Maximin Problem

```
Input: Time horizon $T$, reward vector $R$ on layer $L_k$, initial transition matrices $M_1^0, \dots, M_{k-1}^0$, budget $B$,
       discretization parameter $\varepsilon$.
Output: $M^1, \dots, M^T \in \mathcal{F}\left(B, M_1^0, \dots, M_{k-1}^0\right)$.
Initialization: The no-regret player picks $D^1 = \left(\frac{1}{w}, \dots, \frac{1}{w}\right) \in \mathcal{D}$, the uniform distribution over $[w]$.
for $t = 1, \dots, T$ do
    The no-regret player plays distribution $D^t \in \mathcal{D}$.
    The best-response player chooses $M^t \in \mathcal{F}\left(B, M_1^0, \dots, M_{k-1}^0\right)$ such that
        $$R^\top M_{k-1}^t \dots M_1^t D^t \geq \max_{M \in \mathcal{F}} R^\top M_{k-1} \dots M_1 D^t - \varepsilon \left\|R\right\|_\infty,$$
      using Algorithm 1.
    The no-regret player observes $u_i^t = \frac{R^\top M_{k-1}^t \dots M_1^t e_i}{\|R\|_\infty}$ for all $i \in [w]$, and picks $D^{t+1}$ via multiplicative weight
      update, as follows:
        $$D^{t+1}(i) = \frac{D^t(i) \beta^{u_i^t}}{\sum_{j=1}^w D^t(j) \beta^{u_j^t}} \quad \forall i \in [w],$$
      with $\beta = \frac{1}{1+\sqrt{2 \frac{\ln w}{T}}} \in [0,1)$.
end
```

Lemma 2. *Let $T > 0$, $\overline{\Delta M}$ be the probability distribution that picks $(M_1, \dots, M_{k-1}) \in \mathcal{F}(B, M_1^0, \dots, M_k^0)$ with probability*

$$\frac{1}{T} \sum_{t=1}^{T} \mathbb{1}\left\{(M_1, \dots, M_{k-1}) = \left(M_1^t, \dots, M_{k-1}^t\right)\right\},$$

where $M^1, \dots, M^T$ are the outputs of Algorithm 3. Then $\overline{\Delta M}$ $\left(\varepsilon + \sqrt{2 \frac{\ln w}{T}} + \frac{\ln w}{T}\right) \|R\|_\infty$-approximately optimizes Program 3.

The proof of Lemma 2 follows from interpreting Program 3 as zero-sum game, noting that the best response problem for the maximization player corresponds to the welfare-maximization problem for which we have an efficient algorithm, and then applying the no-regret dynamic analysis from Freund and Schapire [1996]. The details are provided in Appendix D.2.

7 Price of Fairness

In this section, we compute lower bounds on a notion of "price of fairness", and we show these lower bounds are tight when restricting attention to pipelines whose edges are *all* malleable. Specifically, we compare the optimal welfare achievable with the welfare that is achievable if we instead use our budget to maximize the *minimum* value over initial positions — i.e. if we solve the maximin problem. We focus on the ex-post maximin problem — i.e. we prove our bounds with respect to *deterministic* solutions. We note that there may be many different maximin optimal solutions that differ in their overall welfare, and so we consider two variants of the price of fairness in our setting — comparing with both the *maximum* welfare consistent with a maximin optimal solution, and the *minimum* welfare consistent with a maximin optimal solution.

Let OPT_{SW} the optimal value of Program (1) (the optimal social welfare). Let S^f be the set of solutions to Program (2) (the deterministic maximin problem). Further, define

$$W(M_1, \dots, M_{k-1}) \triangleq R^\top M_{k-1} \dots M_1 D_1^0$$

to be the social welfare achieved by transition matrices $M_1, \dots, M_{k-1}$, and

$$W_{fair}^+ \triangleq \max_{(M_1,\dots,M_{k-1}) \in S^f} W(M_1, \dots, M_{k-1}),$$

$$W_{fair}^- \triangleq \min_{(M_1,\dots,M_{k-1}) \in S^f} W(M_1, \dots, M_{k-1})$$

to be the maximum and minimum social welfare respectively that are consistent with maximin optimal solutions. We define two variants of "the price of fairness" in our setting as:

$$P_f^+ \triangleq \frac{OPT_{SW}}{W_{fair}^+} \geq 1,$$

and

$$P_f^- \triangleq \frac{OPT_{SW}}{W_{fair}^-} \geq 1.$$

Note that $P_f^+ \leq P_f^-$ always, as P_f^+ compares the optimal social welfare with the solution of Program 2 with highest social welfare , while P_f^- considers the solution that has the lowest social welfare. We provide matching lower bounds on P_f^+ and upper bounds on P_f^-. This, in turn, provides tight bounds on the price of fairness with respect to *any* choice of maximin solution.

7.1 Lower Bounds on P_f^+

Our lower bounds are based on the following construction:

Example 1. *Consider a network with only two layers, L_1 and L_2, such that L_1 has w nodes and L_2 has 2 nodes. Suppose the starting distribution is given by $D_1^0 = (1-(w-1)\varepsilon, \varepsilon, \dots, \varepsilon)^\top$ for $\varepsilon > 0$ small enough, the reward vector is given by $R = (1, 0)^\top$, and the initial transition matrix M_1^0 is given by*

$$M_1^0 = \begin{pmatrix} 0 & \dots & 0 \\ 1 & \dots & 1 \end{pmatrix}.$$

I.e., in the initial transition matrix, every starting node transitions to the destination node that has reward 0, and the welfare of the initial network is 0. We assume all edges are malleable.

Theorem 3. *For all $w \in \mathbb{N}$, for any $\delta > 0$, there exists a network with $k = 2$ with price of fairness*

$$P_f \geq \begin{cases} w - \delta & \text{if } 0 < B \leq 2 \\ \frac{2w}{B} - \delta & \text{if } 2 < B \leq 2w \\ 1 & \text{if } B \geq 2w \end{cases}.$$

The proof follows from solving the social welfare maximization problem and the maximin value problem on Example 1. The full proof is provided in Appendix D.3.1.

7.2 Upper Bounds on P_f^-

Importantly, in this section, we restrict ourselves to pipelines such that *all* edges are malleable. In this case, we show upper bounds that tightly match the lower bounds of Section 7.1.

Our upper bounds will make use of the following claim, which bounds the maximum social welfare that can be achieved under budget B.

Lemma 3.

$$OPT_{SW} \leq \|R\|_\infty$$

and

$$OPT_{SW} \leq W^0 + \frac{B}{2} \|R\|_\infty ,$$

where $W^0 = R^\top M_{k-1}^0 \dots M_1^0 D_1^0$ *is the initial welfare.*

The proof of this lemma is straightforward and is deferred to Appendix D.3.2.

We will also need lower bounds on the social welfare achieved by any optimal solution to the maximin program. The first lower bound is a function of B and w, but is independent of W^0.

Lemma 4. *When all edges are malleable, for any* $\left(M_1^f, \dots, M_{k-1}^f\right) \in S^f$*,*

$$W\left(M_1^f, \dots, M_{k-1}^f\right) \geq \min\left(1, \frac{B}{2w}\right) \|R\|_\infty .$$

The proof of Lemma 4 is deferred to Appendix D.3.3. We provide a brief proof sketch below:

Proof sketch. The budget B can be fully invested in improving edges from the second-to-last layer L_{k-1} to the last layer L_k. The idea is to increase the transition from any node $u \in L_{k-1}$ to the best node $v \in L_k$ with reward $\|R\|_\infty$, by an amount of $B/2w$ each. Doing so guarantees the result, noting that every starting node in the first layer transitions to a node in L_{k-1} with probability 1, then to reward $\|R\|_\infty$ on the last layer L_k with probability at least $B/2w$. Importantly, note that this proof relies on the fact that the edges from the second-to-last to the last layer are malleable. □

The second lower bound we need shows that the social welfare achieved by a solution to Program (2) is lower-bounded by the initial social welfare $W^0 = R^\top M_{k-1}^0 \dots M_1^0 D_1^0$.

Lemma 5. *When all edges are malleable, for any* $\left(M_1^f, \dots, M_{k-1}^f\right) \in S^f$*,*

$$W\left(M_1^f, \dots, M_{k-1}^f\right) \geq W^0.$$

We defer the full proof of Lemma 5 to Appendix D.3.4 and provide a proof sketch below:

Proof sketch. The proof follows from the fact that increasing the expected reward of any given node u in any layer L_t in the network can be done by taking some of the transition probability from u to a low-reward node and re-allocating it to the transition between u and a higher reward node. Doing so does not decrease the expected reward of any other vertex in the network. In turn, it is always sub-optimal to invest budget into decreasing the expected reward of any node in the network. □

We can now use Lemmas 3, 4 and 5 to derive nearly tight upper bounds on the price of fairness with respect to the worst maximin solution:

Theorem 4. *For every instance of the problem in which edges are malleable, we have that*

$$P_f^- \leq \begin{cases} w+1 & \text{if } 0 < B \leq 2 \\ \frac{2w}{B} & \text{if } 2 < B \leq 2w \\ 1 & \text{if } B \geq 2w \end{cases} .$$

Proof. We divide the proof in three cases:

1. $B \geq 2w$. By Lemma 4, it must be the case that any optimal solution to Program (2) has welfare at least $\min\left(1, \frac{B}{2w}\right) \|R\|_\infty = \|R\|_\infty$. It is then immediately the case that $OPT_{SW} = \|R\|_\infty$ by Lemma 3 and $P_f = 1$.

2. $2 < B \leq 2w$. By Lemma 3, we have $OPT_{SW} \leq \|R\|_\infty$. Further, by Lemma 4, we have that any solution to Program (2) has welfare at least $\frac{B}{2w} \|R\|_\infty$. This immediately yields the result.

3. $0 < B \leq 2$. By Lemma 3, we have $OPT_{SW} \leq W^0 + \frac{B}{2} \|R\|_\infty$. By Lemmas 4 and 5, we have that the social welfare of any maximin solution is at least W^0 and at least $\frac{B}{2w} \|R\|_\infty$. Therefore, the price of fairness is upper-bounded on the one hand by

$$P_f^- \leq \frac{W^0 + \frac{B}{2} \|R\|_\infty}{W^0} = 1 + \frac{\frac{B}{2} \|R\|_\infty}{W^0}.$$

and on the other hand by

$$P_f^- \leq \frac{W^0 + \frac{B}{2} \|R\|_\infty}{\frac{B}{2w} \|R\|_\infty} = w + \frac{W^0}{\frac{B}{2w} \|R\|_\infty}.$$

When $W^0 \geq \frac{B}{2w} \|R\|_\infty$, the first bound gives

$$P_f^- \leq 1 + \frac{\frac{B}{2} \|R\|_\infty}{\frac{B}{2w} \|R\|_\infty} = w + 1,$$

and when $W^0 \leq \frac{B}{2w} \|R\|_\infty$, the second bound yields

$$P_f^- \leq w + \frac{\frac{B}{2w} \|R\|_\infty}{\frac{B}{2w} \|R\|_\infty} = w + 1,$$

which concludes the proof.

□

8 Hardness of Approximation

In this section, we show that the problem of finding the ex-post maximin value of a pipeline intervention problem instance within an approximation factor of 2 is NP-hard in the general case, where the width w of the network is not bounded. More specifically, we show that no algorithm that has a time bound polynomial in w, k and B can give a 2-approximation to the maximin value unless $P = NP$. This hardness result holds for k as small as 17. We remark that our result and proof can be immediately extended to show hardness of C-approximation, for any constant C, for an appropriate choice of constant depth k.

We show this hardness result via a reduction from a gap version of the vertex cover problem. The result of Dinur and Safra [2005] shows that it is NP-hard to approximate the minimum vertex cover by a factor smaller than 1.306. In particular, their result shows that the following gap version of vertex cover is NP-hard: given $(\mathcal{G}, \kappa)$, we wish to either know if the graph $\mathcal{G}$ has a vertex cover of size κ, or has no vertex cover smaller than size 1.306κ.

The Reduction Our reduction works as follows: we construct a pipeline intervention instance of constant width (17 layers) from the given graph. The first layer has a node corresponding to each edge (u, v) of the original graph, and is connected by edges to nodes corresponding to vertices u and v on the second layer. We set up the instance so that positive probability mass is only ever added to a set of edge disjoint paths, where each path corresponds to a vertex in the original graph. These paths are shown by the dark, solid lines in Figure 1. The main idea behind the reduction is the following - by observing how allocations finding the maximin value split the budget over these edge disjoint paths, we can find out which vertices would form a small vertex cover of the original graph.

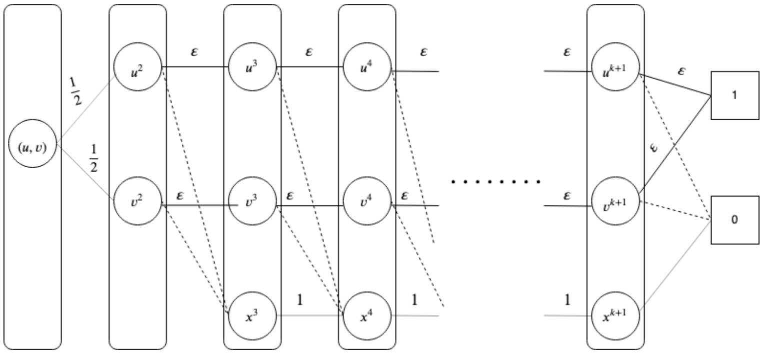

Figure 1: Constructed Instance of the Pipeline Intervention problem

Formally, let the given graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ we reduce from have n vertices ($|\mathcal{V}| = n$) and m edges ($|\mathcal{E}| = m$). We construct of a pipeline intervention problem instance $\mathcal{I}'$ with $k + 2$ layers and width w, where $k = 15$ and w is polynomial in n. The instance $\mathcal{I}'$ has an associated budget $B(\kappa, \varepsilon) = 2k\kappa\varepsilon$ where $\varepsilon < \frac{1}{2}$. For the sake of clarity, we refer to the set of vertices $\mathcal{V}$ in the vertex cover instance as "vertices" and the vertices in the instance $\mathcal{I}'$ as "nodes". A complete description of instance $\mathcal{I}'$ is as follows:

1. The first layer, L_1, has exactly m nodes, with each edge (u, v) in graph $\mathcal{G}$ having a unique corresponding node of the same label in layer L_1.

2. The second layer has exactly n nodes, with each vertex v in graph $\mathcal{G}$, having a unique corresponding node in layer L_2 with label v^2.

3. The next $k - 1$ layers are of the following form - layer L_i, for $i = 3$ to $k + 1$, has $n + 1$ nodes. The first n nodes have labels from the set $\{v^i\}_{v \in V}$, i..e, each vertex v in the original graph $\mathcal{G}$ has a corresponding node v^i in layer L_i. The last node is indexed by x^i and exists to capture the "leftover" outward probability from the nodes $\{v^{i-1}\}_{v \in V}$ in layer L_{i-1}.

4. The final layer L_{k+2} has two reward nodes - y, of reward 1 and z, of reward 0.

We now describe the initial transition matrices.

1. From layer L_1 to layer L_2: for every node (u, v) in layer L_1, the outgoing probability is equally split between edges to nodes u^1 and v^1 in layer L_2, i.e., edges $((u, v), u^1)$ and $((u, v), v^1)$ each have probability $\frac{1}{2}$.

2. From layer L_i to layer L_{i+1} for $i = 2$ to k: For all vertices $v \in \mathcal{V}$ (i.e., the original graph), the corresponding edge (v^i, v^{i+1}) (in our construction) has probability ε. The remaining outgoing probability out of node v^i goes to the leakage node x^{i+1}. We call edges of the form (v^i, x^{i+1}) "leakage" edges. For $i \geq 3$, the edge (x^i, x^{i+1}) has all the outward probability, i.e., 1, from node x^i.

3. From layer L_{k+1} to layer L_{k+2}: each node in layer L_{k+1} is connected to z, the zero reward node, with probability 1.

We let P_v be the path going through nodes $v^2, v^3 \cdots v^{k+1}, y$ in our construction. We will refer to $\{P_v\}_{v \in \mathcal{V}}$ as vertex paths. Let E' be the set of edges found on paths $\{P_v\}_{v \in V}$. Let E'' contain of all the "leakage" edges in the instance $\mathcal{I}'$,i.e., edges of the form (v^i, x^{i+1}) as well as all edges of the form (v^{k+1}, z). We stipulate, as part of the description of the instance, that $E' \cup E''$ is the set of malleable edges in $\mathcal{I}'$ and that the probability mass on any other edge cannot be changed. This completes the description of instance $\mathcal{I}'$.

Formal Hardness Result In this setup, we argue that any algorithm computing a maximin value of our instance will only use budget to improve the probability mass on the vertex paths. To begin, we observe that the only nodes with more than one outgoing malleable edge are the nodes on the vertex paths, exempting the reward node y. Hence, these are the only nodes whose associated outgoing transition matrices can be modified. Furthermore, these nodes have exactly one edge from E', the edges on the vertex paths, and one leakage edge, implying that probability mass can only be transferred between these two edges. Note that the reward associated with starting at any leakage node is always 0, since all outgoing edges from leakage nodes lead to the zero sink with probability 1, and are not malleable. In contrast, nodes on the vertex paths start with expected positive welfare initially. Therefore, it would be a (strictly) sub-optimal strategy to route any more probability mass towards a leakage node , i.e., increase the probability mass on any edge in E'' leading to reward 0, since it would come at the cost of decreasing probability mass on an edge towards a node on a vertex path with strictly positive reward. Consequently, any algorithm that optimizes the maximin value would only increase the probability on the other malleable edges, i.e., edges in E', by removing probability mass from edges E''. To give a more local picture, budget is spent on the following operation - increasing the probability mass on an edge of the form (v^i, v^{i+1}) and balancing the outgoing probability mass from vertex v^i by correspondingly decreasing the probability mass on edge (v^i, x^{i+1}). Note that there are no malleable edges in the transition between layer L_1 and layer L_2, and hence the corresponding transition matrix remains unchanged.

We fix a threshold $T \triangleq \frac{(2\varepsilon)^k}{4}$. The following lemma shows the desired reduction:

Lemma 6. *If graph $\mathcal{G}$ has a vertex cover of size κ, then the maximin value of the constructed instance is at least $2T$. Complementarily, if graph $\mathcal{G}$ has no vertex cover smaller than 1.3605κ, then the maximin value of the constructed instance is less than T.*

Observe that the above lemma shows achieving a 2-approximation to the maximin value is NP-hard, since such an algorithm would be able to solve the gap version of vertex cover in polynomial time. Note that this also rules out an additive PTAS for the pipeline interventions problem, even for networks of constant depth , since our hard instance $\mathcal{I}'$ has constant depth.

Proof. The forward direction of the proof is straightforward. Given that the original graph has a vertex cover $\mathcal{V}^* \subseteq \mathcal{V}$ of size κ, we show a candidate solution that guarantees a minimum reward of $2T$. Consider the subset $E^* \subseteq E$ which consists only of edges on the vertex paths indexed by vertices in the vertex cover $\mathcal{V}^*$, i.e., $\{P_v\}_{v\in\mathcal{V}^*}$. Since each such path has k edges and there are κ number of paths, $|E^*| = k\kappa$. We spend a budget 2ε to increase the probability mass one each edge of E^* by ε and decrease the probability mass of the corresponding leakage edge by ε. Note that this is always possible, since $\varepsilon + \varepsilon \leq 1$ (i.e., we never try to increase the probability mass on an edge beyond 1). This exactly utilizes our budget $B(\kappa, \varepsilon) = 2k\kappa\varepsilon$. The new probability on every edge of E^* is now 2ε. Consider any population starting on a node (u, v) of the first layer. Since $\mathcal{V}^*$ is a vertex cover, at least one of u or v is present in the set $\mathcal{V}^*$. Without loss of generality, let us assume $v \in \mathcal{V}^*$. The reward at node v^2 is now exactly $(2\varepsilon)^k$. Since the transition matrices for layer L_1 to L_2 are unchanged, a population starting at node (u, v) arrives at node v^2 with probability $\frac{1}{2}$; thus, an agent starting at (u, v) has expected reward at least $\frac{(2\varepsilon)^k}{2} = 2T$.

Now, consider the case where the graph has no vertex cover of size 1.305κ. To prove a contradiction, let us assume that the maximin value is greater than or equal to T. We will show that we can recover a vertex cover of size less than size 1.305κ from the maximin value solution, thus proving a contradiction. Let $W(l)$ represent the reward associated with starting at node l. We know that all the budget is spent on the paths $\{P_v\}_{v\in V}$. Let use write $2k\varepsilon a_v$ be the budget spent on improving path P_v; note that the net increase in probability mass across all edges of P_v is $k\varepsilon a_v$. By the AM-GM inequality, we know that $W(v^2) \leq (\varepsilon + a_v\varepsilon)^k$, with equality when $ka_v\varepsilon$ is split equally across edges on path P_v. A population starting at node (u, v) on layer L_1 reaches node u_2 and node v^2 each with probability $\frac{1}{2}$ each. Thus, we have $W((u, v)) \leq \frac{(\varepsilon+a_u\varepsilon)^k+(\varepsilon+a_v\varepsilon)^k}{2}$. Since this solution guarantees a reward of at least T for every vertex within a total budget $B(\kappa, \varepsilon) = 2k\kappa\varepsilon$, we have the following inequalities:

$$\frac{(\varepsilon + a_u\varepsilon)^k + (\varepsilon + a_v\varepsilon)^k}{2} \geq \frac{(2\varepsilon)^k}{4} \quad \forall (u, v) \in E$$

$$\sum_{v\in V} a_v \leq \kappa\varepsilon$$

Dividing the first inequality by ε^k, multiplying by 2, and substituting $k = 15$, we get:

$$(1 + a_u)^{15} + (1 + a_v)^{15} \geq 2^{14} \quad \forall (u, v) \in E$$

$$\sum_{v\in V} a_v \leq \kappa$$

We now generate a vertex set $\mathcal{V}^*$ of original graph $\mathcal{G}$ as follows - include vertex v in $\mathcal{V}^*$ if $a_v \geq 0.823$. We complete the proof by showing that $\mathcal{V}^*$ is a vertex cover of G and $|\mathcal{V}^*| < 1.3605\kappa$. The fact that $|\mathcal{V}^*| < 1.3605\kappa$ follows directly from the rounding scheme that is employed. Observe that each a_v is scaled upward by a factor of at most $\frac{1}{0.823} < 1.22$. Thus, $|\mathcal{V}^*| < 1.22\kappa$. Now, assume that $\mathcal{V}^*$ is not a vertex cover. Then, there exists an edge $(u, v) \in \mathcal{E}$ in the original graph such that $a_u, a_v < 0.823$. Thus, we get:

$$(1 + a_u)^{15} + (1 + a_v)^{15} < 2(1.823)^{15} < 16325 < 16384 = 2^{14}$$

This violates the reward guarantee for the population starting at node (u, v), thus resulting in a contradiction. □

References

Siddharth Barman and Sanath Kumar Krishnamurthy. Approximation algorithms for maximin fair division. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, pages 647–664, 2017.

Matt Barnum. *A new study questions whether Head Start still produces long-run gains seen in past research*, 2019. URL `https://www.chalkbeat.org/2019/8/8/21108602/a-new-study-questions-whether-head-start-still-produces-long-run-gains-seen-in-past-researc`

Amanda Bower, Sarah N Kitchen, Laura Niss, Martin J Strauss, Alexander Vargas, and Suresh Venkatasubramanian. Fair pipelines. *arXiv preprint arXiv:1707.00391*, 2017.

Eric Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6):1061–1103, 2011.

Alexandra Chouldechova and Aaron Roth. The frontiers of fairness in machine learning. *arXiv preprint arXiv:1810.08810*, 2018.

Stephen Coate and Glenn C Loury. Will affirmative-action policies eliminate negative stereotypes? *The American Economic Review*, pages 1220–1240, 1993.

Irit Dinur and Samuel Safra. On the hardness of approximating minimum vertex cover. *Annals of mathematics*, pages 439–485, 2005.

Cynthia Dwork and Christina Ilvento. Fairness under composition. In *10th Innovations in Theoretical Computer Science Conference (ITCS 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

Dean P Foster and Rakesh V Vohra. An economic argument for affirmative action. *Rationality and Society*, 4(2):176–188, 1992.

Yoav Freund and Robert E Schapire. Game theory, on-line prediction and boosting. In *Proceedings of the ninth annual conference on Computational learning theory*, pages 325–332, 1996.

Lily Hu and Yiling Chen. A short-term intervention for long-term fairness in the labor market. In *Proceedings of the 2018 World Wide Web Conference*, pages 1389–1398, 2018.

Christopher Jung, Sampath Kannan, Changwa Lee, Mallesh M. Pai, Aaron Roth, and Rakesh Vohra. Fair prediction with endogenous behavior. *Manuscript*, 2020.

Sampath Kannan, Aaron Roth, and Juba Ziani. Downstream effects of affirmative action. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 240–248, 2019.

Lydia T Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. Delayed impact of fair machine learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 6196–6200. AAAI Press, 2019a.

Lydia T Liu, Ashia Wilson, Nika Haghtalab, Adam Tauman Kalai, Christian Borgs, and Jennifer Chayes. The disparate equilibria of algorithmic decision making when individuals invest rationally. *arXiv preprint arXiv:1910.04123*, 2019b.

Hussein Mouzannar, Mesrob I Ohannessian, and Nathan Srebro. From fair decision making to social equality. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 359–368, 2019.

Niche. *Best Public Elementary Schools in America*, 2020. URL https://www.niche.com/k12/search/best-public-elementary-schools/.

Ariel D Procaccia and Junxing Wang. Fair enough: Guaranteeing approximate maximin shares. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pages 675–692, 2014.

U.S. News. *Best High Schools Rankings*, 2020. URL https://www.usnews.com/education/best-high-schools.

Samuel F Way, Daniel B Larremore, and Aaron Clauset. Gender, productivity, and prestige in computer science faculty hiring networks. In *Proceedings of the 25th International Conference on World Wide Web*, pages 1169–1179, 2016.

A Preliminaries, continued

A.1 Proof of Claim 1

We will prove both results at once, noting that they both directly follow from showing that there exists matrices $M_1^\varepsilon, \ldots, M_k^\varepsilon$ that are feasible for the discretized problems, such that for all $j \in [w]$,

$$R^\top M_{k-1}^\varepsilon \ldots M_1^\varepsilon e_j \geq R^\top M_{k-1}^* \ldots M_1^* e_j - (k-1)\varepsilon,$$

where $M_1^*, \ldots, M_{k-1}^*$ is an optimal solution to Program (1), respectively (2). To do so, we first note that it is feasible for Program (4) to pick a split of the budget $B_1^\varepsilon, \ldots, B_{k-1}^\varepsilon$ such that for all t, $B_t^\varepsilon \geq \max(c(M_t^*, M_t^0) - \varepsilon, 0)$, by construction of $\mathcal{B}(\varepsilon)$. We are going to construct a matrix M_t^ε that is close to M_t^* and requires budget at most B_t^ε.

When $M_t^* = M_t^0$, one can just let $M_t^\varepsilon = M_t^0$. Now, suppose $c(M_t^*, M_t^0) > 0$. For every pair of nodes $u \in L_t$, we let S_u^+ the set of vertices $v \in L_{t+1}$ such that $M_t^*(v,u) > M_t^0(v,u)$ (i.e. the transition from u to v has higher probability in M_t^* than in M_0), and S_u^- the set of vertices $v \in L_{t+1}$ such that $M_t^*(v,u) < M_t^0(v,u)$. We note immediately that

$$c(M_t^*, M_t^0) = \sum_{u \in L_t} \left(\sum_{v \in S_u^+} \left(M_t^*(v,u) - M_t^0(v,u)\right) + \sum_{v \in S_u^-} \left(M_t^0(v,u) - M_t^*(v,u)\right), \right)$$

Now, let us construct $M_t^\varepsilon \in \mathcal{M}$ such that for every u,

$$M_t^\varepsilon(v,u) = M_t^*(v,u) - \alpha(v,u) \;\forall v \in S_u^+$$

and

$$M_t^\varepsilon(v,u) = M_t^*(v,u) + \alpha(v,u) \;\forall v \in S_u^-,$$

where $\alpha(v,u) \geq 0$ for all u, v, $\sum_{u,v} \alpha(v,u) = \min(\varepsilon, c(M_t^*, M_t^0))$, $\sum_{v \in S_u^+} \alpha(v,u) = \sum_{v \in S_u^-} \alpha(v,u)$, and $M_t^\varepsilon(u,v) \geq M_t^0(u,v)$ for $v \in S_u^+$ and $M_t^\varepsilon(u,v) \leq M_t^0(u,v)$ for $v \in S_u^-$. Note that such α's exist by virtue of $\min(\varepsilon, c(M_t^*, M_t^0)) \leq c(M_t^*, M_t^0)$, which is the absolute value amount by which M_t^* differs from M_t^0 coordinate-by-coordinate. Second, note that only malleable edges (u,v) have $M_t^\varepsilon(v,u) \neq M_t^0(v,u)$, since we only modify malleable edges where $M_t^*(v,u) \neq M_t^0(v,u)$. Further, $M_t \in \mathcal{M}$ since all the coefficients of M_t remain between 0 and 1, and for all u,

$$\sum_v M_t(v,u) = \sum_v M_t^*(v,u) + \sum_{v \in S_u^-} \alpha(v,u) - \sum_{v \in S_u^+} \alpha(v,u) = \sum_v M_t^*(v,u) = 1.$$

Further, the cost of moving from M_t^0 to M_t^ε is given by

$$\begin{aligned}
c(M_t^\varepsilon, M_t^0) &= \sum_{u \in L_t} \left(\sum_{v \in S_u^+} \left(M_t^\varepsilon(v,u) - M_t^0(v,u)\right) + \sum_{v \in S_u^-} \left(M_t^0(v,u) - M_t^\varepsilon(v,u)\right) \right) \\
&= \sum_{u \in L_t} \left(\sum_{v \in S_u^+} \left(M_t^*(v,u) - M_t^0(v,u) - \alpha(v,u)\right) + \sum_{v \in S_u^-} \left(M_t^0(v,u) - M_t^*(v,u) - \alpha(v,u)\right) \right) \\
&= c(M_t^*, M_t^0) - \sum_{u,v} \alpha(v,u) \\
&= \max\left(0, c(M_t^*, M_t^0) - \varepsilon\right),
\end{aligned}$$

noting that if v was in S_u^+ (resp S_u^-), it is still the case that $M_t^\varepsilon(v,u) \geq M_t^0(v,u)$ (resp. $M_t^\varepsilon(v,u) \leq M_t^0(v,u)$). In turn, M_t^ε requires at most budget B_t^ε, and $M_1^\varepsilon, \ldots, M_{k-1}^\varepsilon$ is a feasible solution for the discretized programs. Finally, for any transition matrices $M_1, \ldots, M_{k-1}$, letting $R_{t+1}^\top = R^\top M_{k-1} \ldots, M_{t+1}$ (trivially, $0 \leq R_{t+1} \leq \|R\|_\infty$) and $D_{t,j} = M_{t-1} \ldots M_1 e_j$ (trivially, $D_{t,j} \in \mathcal{D}$), we note that (letting $\alpha(v,u) = 0$ where not defined)

$$\begin{aligned}
R_{t+1}^\top M_t D_{t,j} &= \sum_{u \in L_t,\; v \in L_{t+1}} M_t^\varepsilon(v,u) R_{t+1}(v) D_{t,j}(u) \\
&\geq \sum_{u,v} M_t^*(v,u) R_{t+1}(v) D_{t,j}(u) - \sum_{u,v} \alpha(v,u) R_{t+1}(v) D_{t,j}(u) \\
&\geq \sum_{u,v} M_t^*(v,u) R_{t+1}(v) D_{t,j}(u) - \|R\|_\infty \sum_{u,v} \alpha(v,u) \\
&\geq \sum_{u,v} M_t^*(v,u) R_{t+1}(v) D_{t,j}(u) - \|R\|_\infty \varepsilon,
\end{aligned}$$

where the first inequality uses that for all u, v, $M_t^\varepsilon(v,u) \geq M_t^*(u,v) - \alpha(v,u)$ by construction, the second inequality that $0 \leq \alpha(v,u)$, $0 \leq D_{t,j}(u) \leq 1$ and $0 \leq R_{t+1}(v) \leq \|R\|_\infty$, and the last inequality from the fact that $\sum_{u,v} \alpha(v,u) = \min(\varepsilon, c(M_t^*, M_t^0)) \leq \varepsilon$. The proof can be concluded noting that for all t, the above inequality implies

$$R^\top M_{k-1}^* \dots M_{t+1}^* M_t^\varepsilon \dots M_1^\varepsilon e_j \geq R^\top M_{k-1}^* \dots M_{t+2}^* M_{t+1}^\varepsilon M_t^\varepsilon \dots M_1^\varepsilon e_j - \|R\|_\infty \varepsilon$$

and applying this new inequality recursively.

A.2 Proof of Claim 2

This is a well-known result; we provide a proof for completeness. The first observation is that such a net can be constructed recursively, as follows. Start with an empty set S. Initialize by picking any point D in $\mathcal{D}$, and let $S = \{D\}$. Then, recursively keep finding points $D' \in \mathcal{D}$ such that for all $D \in S$, $\|D - D'\|_1 > \varepsilon$, and augment $S := S \cup D'$. Finally, stop the algorithm when no such point $D' \in \mathcal{D}$ exists. By construction, it must be that when the algorithm terminates, for all $D' \in \mathcal{D}$, there exists $D \in S$ with $\|D - D'\|_1 \leq \varepsilon$. As such, S constitutes an ε-net in ℓ_1 distance for $\mathcal{D}$.

Second, we bound the number of steps needed. To do so, we remark that by construction, for all $D_1, D_2 \in S$, it must be the case that $\|D_1 - D_2\|_1 > \varepsilon$; in turn, the ℓ_1-balls of radius $\varepsilon/2$ around each of the elements of S must be disjoint, and the sum of their volumes is less than the volume of $\mathcal{D}$. Since the volume of the probability simplex is given by $\frac{1}{w!}$, and the volume of an ℓ_1-ball of radius r is given by $\frac{1}{w!}(2r)^w$, this yields that $|S| \times \frac{\varepsilon^w}{w!} \leq \frac{1}{w!}$, or equivalently $|S| \leq \left(\frac{1}{\varepsilon}\right)^w$.

B A More General Cost Model

In this section, we show how to extend our algorithmic results to convex costs whose variations are lower-bounded. More specifically, we make the following assumptions on $c(M_t, M_t^0)$, the cost of transforming the initial transition matrix M_t^0 into an alternative $M_t \in \mathcal{M}$.

Assumption 1 (Initial Condition). $c(M_t^0, M_t^0) = 0$.

This encodes the natural assumption that not intervening on the transition matrix incurs no cost.

Assumption 2 (Convexity). *The function $M_t \to c(M_t, M_t^0)$ is convex for all $M_t^0 \in \mathcal{M}$.*

This is a standard assumption, that ensures the optimization problem solved for a single layer is convex and efficiently solvable.

Assumption 3 (Linearly increasing costs). *There exists a constant $L > 0$ such that*

$$\left|c(M_t^2, M_t^0) - c(M_t^1, M_t^0)\right| \geq L \sum_{(i,j) \in [w]^2} \left|M_t^2(i,j) - M_t^1(i,j)\right|$$

for any M_t^0, M_t^1, $M_t^2 \in \mathcal{M}$ such that for all $(i,j) \in [w^2]$, either $M_t^2(i,j) \leq M_t^1(i,j) \leq M_t^0(i,j)$ or $M_t^2(i,j) \geq M_t^1(i,j) \geq M_t^0(i,j)$.

This assumption formalizes the natural property that modifying M_t^0 does not come for free: the further away M_t is from M_t^0, the higher the cost of the transformation from M_t^0 to M_t. The assumption requires that an increase of the distance between $M_t(i,j)$ and $M_t^0(i,j)$ by one translates into at least a L increase in the incurred cost. The condition on M_t^1 and M_t^2 implies that M_t^2 is further away from M_t^0 than M_t^1 by an amount given exactly by $\sum_{(i,j)\in[w]^2} \left|M_t^2(i,j) - M_t^1(i,j)\right|$.

Note that this condition on M_t^1 and M_t^2 fixes the direction — defined by the set of edges that are increased and the set of edges that are decreased — in which we move from M_t^0 to M_t, and prevents comparisons between matrices that have been obtained by changing M_t^0 in *different* directions. Such modifications in different directions can be incomparable in practice, which is why we make no assumption on how they compare in terms of cost. Note that the cost function $c(M_t, M_t^0)$ used in the main body of the paper immediately satisfies Assumptions 1, 2 (as a sum of convex functions), and 3 (with equality for $L = 1$).

To argue that our results carry through to cost functions that satisfy Assumptions 1, 2, 3, we first note that by the same proof as that of Theorem (1), Algorithms (1) and (2) find solutions with social welfare at least $OPT^\varepsilon - 2(k-t)\varepsilon\|R\|_\infty$ and maximin value at least $OPT_{MM}^\varepsilon - 2(k-t)\varepsilon\|R\|_\infty$. Further, since the costs are convex, Programs (7) and (8) are convex optimization programs that can be solved in polynomial running times $f(w)$, $g(w)$. The total running times of Algorithms (1) and (2) are otherwise unchanged.

To show that Algorithms (1) and (2) efficiently find near-optimal solutions to Programs (1) and (2), it is therefore enough to bound the difference between OPT^ε (resp. OPT_{MM}^ε) and OPT (resp. OPT_{MM}). We do so in Claim 3 below:

Claim 3. *There exists a feasible solution $\left(M_1^\varepsilon, \dots, M_{k-1}^\varepsilon\right)$ to Program (4) (resp. Program (5)) with objective value at least $OPT_{SW} - \frac{(k-1)\varepsilon}{L}\|R\|_\infty$ (resp. $OPT_{MM} - \frac{(k-1)\varepsilon}{L}\|R\|_\infty$).*

Proof. We show that there exists matrices $M_1^\varepsilon, \dots, M_k^\varepsilon$ that are feasible for the discretized problem of Program (4), such that for all $j \in [w]$, $R^\top M_{k-1}^\varepsilon \dots M_1^\varepsilon e_j \geq R^\top M_{k-1}^* \dots M_1^* e_j - (k-1)\varepsilon$, where $M_1^*, \dots, M_{k-1}^*$ is an optimal solution to Program (1), respectively (2). We let $B_t = c(M_t^*, M_t^0)$ for simplicity of notations. We also write $\varepsilon' = \frac{\varepsilon}{L}$, and consider a budget split for Program (4) such that for all t, $B_t^\varepsilon \geq \max(B_t - \varepsilon, 0)$. Such a budget split is feasible by construction of $\mathcal{B}(\varepsilon)$.

We now construct M_t^ε as follows: we let $M_t^\varepsilon \triangleq \lambda M_t^0 + (1-\lambda)M_t^*$ where $\lambda = \frac{\varepsilon'}{\sum_{i,j}\left|M_t^*(i,j) - M_t^0(i,j)\right|}$ if $\varepsilon' \leq \sum_{i,j}\left|M_t^*(i,j) - M_t^0(i,j)\right|$, and $\lambda = 1$ otherwise. In the first case, note that since $\lambda \in [0,1]$, it must be that for all (i,j), either $M_t^*(i,j) \geq \lambda M_t^0(i,j) + (1-\lambda)M_t^*(i,j) \geq M_t^0(i,j)$ or $M_t^*(i,j) \leq \lambda M_t^0(i,j) + (1-\lambda)M_t^*(i,j) \leq M_t^0(i,j)$. We can therefore apply Assumption 3 to show that

$$\begin{aligned}
c(M_t^\varepsilon, M_t^0) &\leq c(M_t^*, M_t^0) - L\sum_{i,j}|M_t^*(i,j) - M_t^\varepsilon(i,j)| \\
&= B_t - L\sum_{i,j}\left|M_t^*(i,j) - \lambda M_t^0(i,j) - (1-\lambda)M_t^*(i,j)\right| \\
&= B_t - L\lambda\sum_{i,j}\left|M_t^*(i,j) - M_t^0(i,j)\right| \\
&= B_t - L\varepsilon' \\
&= B_t - \varepsilon.
\end{aligned}$$

Therefore, M_t^ε is feasible for budget B_t^ε. In the second case, $\lambda = 1$, hence $M_t^\varepsilon = M_t^0$ and is feasible for budget 0 hence B_t^ε.

It remains to show that M_t^ε yields a good approximation to M_t. To see this, for any transition matrices $M_1, \dots, M_{k-1}$, let $R_{t+1}^\top = R^\top M_{k-1} \dots, M_{t+1}$ (trivially, $0 \leq R_{t+1} \leq \|R\|_\infty$) and $D_{t,j} =$

$M_{t-1}\dots M_1 e_j$ (trivially, $D_{t,j}\in\mathcal{D}$). We note that

$$\begin{aligned}R_{t+1}^\top(M_t^*-M_t^\varepsilon)D_{t,j} &= \lambda R_{t+1}^\top(M_t^*-M_t^0)D_{t,j}\\ &= \lambda\sum_{u,v}R_{t+1}(v)(M_t^*(v,u)-M_t^0(v,u))D_{t,j}(u)\\ &\le \lambda\|R\|_\infty\sum_{u,v}\left|M_t^*(v,u)-M_t^0(v,u)\right|.\end{aligned}$$

There are now two cases. Either i) $M_t^*=M_t^0$, in which case $M_t^*=M_t^\varepsilon$ hence $R_{t+1}^\top(M_t^*-M_t^\varepsilon)D_{t,j}=0$, or ii) $\lambda=\frac{\varepsilon'}{\sum_{u,v}\left|M_t^*(v,u)-M_t^0(v,u)\right|}$, and we obtain $R_{t+1}^\top(M_t^*-M_t^\varepsilon)D_{t,j}\le\|R\|_\infty\varepsilon'=\|R\|_\infty\frac{\varepsilon}{L}$ by the above equation. In turn, for all t, R_{t+1}, and $D_{t,j}$, we have that $R_{t+1}^\top M_t^* D_{t,j}-\|R\|_\infty\frac{\varepsilon}{L}\le R_{t+1}^\top M_t^\varepsilon D_{t,j}$. Therefore, we have that

$$R^\top M_{k-1}^*\dots M_{t+1}^* M_t^\varepsilon\dots M_1^\varepsilon e_j\ge R^\top M_{k-1}^*\dots M_{t+2}^* M_{t+1}^\varepsilon M_t^\varepsilon\dots M_1^\varepsilon e_j-\|R\|_\infty\frac{\varepsilon}{L},$$

and the result can be obtained via a straightforward induction on t. □

C A Separation between Ex-ante and Ex-post Maximin Values

In this section, we show there is a separation between ex-ante and ex-post maximin welfare. We do so by constructing a specific instance $\mathcal{I}$ for which the ex-ante maximin value is strictly larger than the ex-post maximin value:

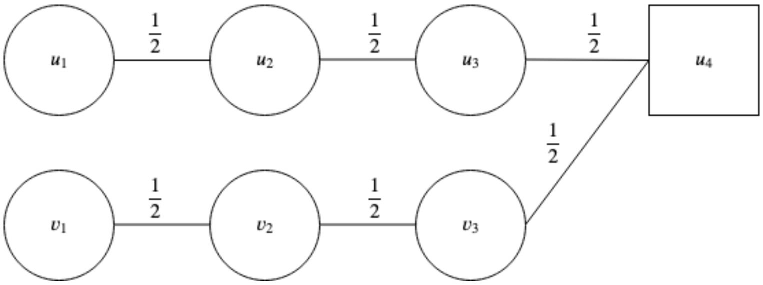

Figure 2: Part $\mathcal{I}_1$ of Instance $\mathcal{I}$. Edges not explicitly drawn have transition probability 0.

We begin by giving a complete description of our instance $\mathcal{I}$. The instance consists of two parts, $\mathcal{I}_1$ (depicted in Figure 2) and $\mathcal{I}_2$. Note that the transition probabilities from any given node in $\mathcal{I}_1$, shown in Figure 2, do not sum to 1. Part $\mathcal{I}_2$ of the instance, described below, serves to complete these transitions and ensure that the total outgoing probability of any node is 1. We assume every edge starting from a node in $\mathcal{I}_1$ is malleable (this includes edges pointing to $\mathcal{I}_2$), and every edge starting from a node in $\mathcal{I}_2$ is non-malleable (this includes edges pointing to $\mathcal{I}_1$).

Our proof primarily focuses on part $\mathcal{I}_1$, as we argue that a centralized designer should only ever invest his budget into increasing the weight on edges with both ends in $\mathcal{I}_1$ (as long as the budget is not too big).

Formally, the instance consists of four layers (L_1 to L_4) and a small enough total budget B. The vertices u_1 to u_4 and v_1 to v_3, combined with all edges between them can be thought of as the part $\mathcal{I}_1$ of instance $\mathcal{I}$. All the remaining vertices and edges are considered to be part of $\mathcal{I}_2$.

1. The first layer consists only of vertices u_1 and v_1.
2. The second layer consists of 3 vertices in total. There are two special vertices - u_2 and v_2, seen in Figure 2. The remaining vertex is called x and is part of $\mathcal{I}_2$.
3. The third layer consists of 3 vertices in total. There are two special vertices - u_3 and v_3, seen in Figure 2. The remaining vertex is labeled y and is part of $\mathcal{I}_2$.
4. The fourth layer, which is the reward layer, consists of two vertices, vertex u_4 of reward 1 and vertex z, of reward 0.

The initial transition matrices are given as follows:

1. From layer L_1 to layer L_2 - the edge (u_1, u_2) has probability $\frac{1}{2}$, and the remaining output probability goes to vertex x, i.e. (u_1, x) has probability $\frac{1}{2}$. Similarly, the edge (v_1, v_2) has probability $\frac{1}{2}$, and the remaining probability is such that (v_1, x) has probability 1/2.
2. From layer L_2 to layer L_3 - the edge (u_2, u_3) has probability $\frac{1}{2}$, and (u_2, y) has the remaining probability $\frac{1}{2}$. Similarly, edge (v_2, v_3) has probability $\frac{1}{2}$, and the remaining outgoing probability is such that (v_2, y) has probability $\frac{1}{2}$. Edge (x, y) has probability 1.
3. From layer L_3 to layer L_4 - the edge (u_3, u_4) has probability $\frac{1}{2}$ and the edge (u_3, z) has the remaining outgoing probability $\frac{1}{2}$ from vertex u_3. Similarly, the edge (v_3, u_4) has probability $\frac{1}{2}$ and the edge (v_3, z) has the remaining outgoing probability $\frac{1}{2}$ from vertex v_3. Edge (y, z) has probability 1.

We refer to the path (u_1, u_2, u_3, u_4) as path P_1 or the upper path and path (v_1, v_2, v_3, u_4) as path P_2 or the lower path. Our proof relies on the following two lemmas:

Lemma 7. *The ex-ante maximin value that can be guaranteed for instance $\mathcal{I}$ is at least*

$$U \triangleq \frac{1}{2}\left(\frac{1}{8} + \left(\frac{1}{2} + \frac{B}{6}\right)^3\right).$$

Proof. We show a candidate solution that gives expected reward U to both agents. We assign a budget of $B/3$ each for the transition between L_1 and L_2, the transition between L_2 and L_3, and the transition between L_3 and L_4. We construct a feasible transition matrix M_1 by decreasing the budget invested between any two layers in $\mathcal{I}_2$ by $B/6$ (which can be done for B small enough), and increasing the probabilities of edges (u_1, u_2), (u_2, u_3), and (u_3, u_4) by $B/6$. We leave other edges in $\mathcal{I}_1$ untouched. M_1 yields reward $\left(\frac{1}{2} + \frac{B}{6}\right)^3$ for u_1 and 1/8 for u_2. Symmetrically, we construct a solution M_2 with expected reward 1/8 for u_1 and $\left(\frac{1}{2} + \frac{B}{6}\right)^3$ for u_2. The solution that picks M_1 with probability 1/2 and M_2 with probability 1/2 guarantees maximin value of U. □

We now show that the ex-post maximin value is strictly smaller than U. We will reason exclusively on $\mathcal{I}_1$, noting that when the budget B is small and w is large enough, investing any money on edges not contained within $\mathcal{I}_1$ is sub-optimal. This can be seen immediately: since one cannot invest in outgoing edges from nodes x and y in $\mathcal{I}_2$, these nodes point to z with probability 1 (as (x, y) and (y, z) are not malleable) and have reward 0; on the other hand, the input probability and the reward of any vertex in $\mathcal{I}_1$ is strictly positive for B small enough. In turn, for B small enough, nodes in $\mathcal{I}_1$ always have strictly higher rewards and input probabilities than any node in $\mathcal{I}_2$ for any feasible allocation, and it is optimal to invest in improving only the transition probabilities from nodes in $\mathcal{I}_1$ to nodes in $\mathcal{I}_1$.

To reason about how to optimally use the budget to improve edges in $\mathcal{I}_1$, we introduce new notations. We let $W(v)$ denote the expected reward obtained by an agent starting at any vertex v in the instance. We define $w_u^i \triangleq W(u_i)$ and $w_v^i \triangleq W(v_i)$ for $i \in \{1, 2, 3, 4\}$. We let B_1, B_2, B_3 denote the budget split across the three transitions, and let $B_{\geq i}$ represent the budget spent to the right of layer L_i. Note that, in our instance, $B_{\geq 3} = B_3$. We now state the following the key lemma and show how it implies the desired separation.

Lemma 8.

$$w_u^1 + w_v^1 \leq 2U.$$

Further, $w_u^1 + w_v^1 = 2U$ only holds when the budget is split equally across the transition between layers, i.e., $B_1 = B_2 = B_3 = \frac{B}{3}$; in this case,

$$w_u^2 + w_v^2 \leq \frac{1}{4} + \left(\frac{1}{2} + \frac{B}{6}\right)^2,$$

$$w_u^2, w_v^2 \leq \left(\frac{1}{2} + \frac{B}{6}\right)^2.$$

We defer the proof of Lemma 8 to Appendix C.1, and conclude the proof of separation by showing that Lemma 8 implies the following corollary:

Corollary 4. *The ex-post maximin value is strictly less than U.*

Proof. Observe that the minimum reward over all agents, $\min\{w_u^1, w_v^1\}$, is upper bounded by $\frac{w_u^1 + w_v^1}{2}$. Thus, by Lemma 8, we have that $\min\{w_u^1, w_v^1\} \leq \frac{w_u^1 + w_v^1}{2} \leq U$. If $w_u^1 + w_v^1 < 2U$, the result holds. Hence, we only need to consider the case when $w_u^1 + w_v^1 = 2U$. In this case, the budget split is given by $B_1 = B_2 = B_3 = B/3$ by Lemma 8. We assume with loss of generality that $w_u^2 \geq w_v^2$ (otherwise invert the roles of u_2 and v_2); note that when the budget is small enough, it must be the case that w_u^2, w_v^2 are necessarily bigger than the rewards of node x in layer 2 in $\mathcal{I}_2$ — as B tends to 0, the rewards on nodes in the second layer in $\mathcal{I}_1$ tend to 1/4, while the reward of node x remains 0 always by non-malleability of (x, y) and (y, z)). This directly implies that there exists an optimal maximin solution in which all of the budget $B/6$ allocated to improving edges (remembering that we need to spend half the budget, i.e. $B/6$, decreasing edges in $\mathcal{I}_2$ for our matrix to remain stochastic) is allocated to (u_1, u_2) and (v_1, u_2). Let $\delta \in [0, 1]$ be such that the probability transition of edge (u_1, u_2) is set to $\frac{1}{2} + \delta \frac{B}{6}$, and the probability transition for (v_1, u_2) is set to $(1 - \delta)\frac{B}{6}$. We have that there exists δ such that

$$\begin{aligned} w_u^1 + w_v^1 &\leq \left(\frac{1}{2} + \delta\frac{B}{6}\right) w_u^2 + (1-\delta)\frac{B}{6} w_u^2 + \frac{1}{2} w_v^2 \\ &= \left(\frac{1}{2} + \frac{B}{6}\right) w_u^2 + \frac{1}{2} w_v^2, \end{aligned}$$

noting that all other transitions must go to x and yield reward 0, as with probability 1, x goes to y and y goes to z with reward 0 by non-malleability. Since $w_u^2 \leq \left(\frac{1}{2} + \frac{B}{6}\right)^2$, it must be that $w_u^2 = \left(\frac{1}{2} + \frac{B}{6}\right)^2$ and $w_v^2 = 1/4$, otherwise we would have $w_u^1 + w_v^1 < 2U$ which is a contradiction. This implies in particular that

$$\begin{aligned} w_v^1 &\leq \frac{B}{6} w_u^2 + \frac{1}{2} w_v^2 \\ &= \frac{B}{6}\left(\frac{1}{2} + \frac{B}{6}\right)^2 + \frac{1}{8}. \end{aligned}$$

A simple calculation shows that for B small enough,

$$\frac{B}{6}\left(\frac{1}{2} + \frac{B}{6}\right)^2 + \frac{1}{8} < \frac{1}{2}\left(\frac{1}{2} + \frac{B}{6}\right)^3 + \frac{1}{16} = U,$$

which concludes the proof. □

C.1 Proof of Lemma 8

We prove the lemma by proving a similar result at each layer by induction, starting backward from the penultimate layer.

Lemma 9. *For any layer L_i with $i \in \{1, 2, 3\}$, we have*

$$w_u^i + w_v^i \leq \left(\frac{1}{2^{4-i}} + \left(\frac{1}{2} + \frac{B_{\geq i}}{2(4-i)}\right)^{4-i}\right)$$
$$\max\{w_u^i, w_v^i\} \leq \left(\frac{1}{2} + \frac{B_{\geq i}}{2(4-i)}\right)^{4-i}$$

The first inequality is tight at layer L_i only when $B_{\geq i}$ is split equally across the transition between layers to the right of L_i.

We note that the above lemma applied at layer 1 directly gives that $w_u^1 + w_v^1 \geq 2U$ only when $B_1 = B_2 = B_3 = \frac{B}{3}$ when equality holds, and all vertices in $\mathcal{I}_2$ have 0 reward. The second part of Lemma 8 holds from applying Lemma 9 at layer 2 with $B_2 = B_3 = \frac{B}{3}$ or equivalently, $B_{\geq 2} = \frac{2B}{3}$.

Proof. Note that this lemma is trivially true for any layer L_i when $B_{\geq i} = 0$. Henceforth, we only look at layer L_i when $B_{\geq i} > 0$.

We begin by proving the lemma statement for Layer L_3 and work backwards towards layer L_1. Note that $w_u^3 + w_v^3$ is maximized only by spending $B_{\geq 3} = B_3$ on edges from either u_3 or v_3 to the reward node u_4 with reward 1. This gives us $w_u^3 + w_v^3 \leq \frac{1}{2} + \frac{1}{2} + \frac{B_{\geq 3}}{2}$. Without loss of generality (due to the symmetry in the instance), let $w_u^3 \geq w_v^3$. Then w_u^3 is maximized by spending all the budget $B_{\geq 3}$ on edge (u_3, u_4), i.e., path P_1 (in the other case, all the budget is spent on path P_2). Thus, $\max\{w_u^3, w_v^3\} \leq \frac{1}{2} + \frac{B_{\geq 3}}{2}$. Thus, both parts of the lemma are true for Layer L_3.

Now, let us assume our induction hypothesis holds at layer L_{i+1}. Consider layer L_i. Let us assume w.l.o.g that $w_u^{i+1} \geq w_v^{i+1}$. To maximize $w_u^i + w_v^i$, it is easy to see that all budget must be spent on edges from u_i or v_i to u_{i+1} — since x, y, and z always have reward 0. Thus, we get:

$$\begin{aligned} w_u^i + w_v^i &\le \left(\frac{1}{2} + \frac{B_i}{2}\right) w_u^{i+1} + \frac{1}{2} w_v^{i+1} \\ &\le \left(\frac{1}{2} + \frac{B_i}{2}\right) w_u^{i+1} + \frac{1}{2}\left(\frac{1}{2^{4-i-1}} + \left(\frac{1}{2} + \frac{B_{\ge i+1}}{2(4-i-1)}\right)^{4-i-1} - w_u^{i+1}\right) \\ &= \frac{B_i}{2} w_u^{i+1} + \frac{1}{2^{4-i}} + \frac{1}{2}\left(\frac{1}{2} + \frac{B_{\ge i+1}}{2(4-i-1)}\right)^{4-i-1} \\ &\le \frac{1}{2^{4-i}} + \frac{B_i}{2}\left(\frac{1}{2} + \frac{B_{\ge i+1}}{2(4-i-1)}\right)^{4-i-1} + \frac{1}{2}\left(\frac{1}{2} + \frac{B_{\ge i+1}}{2(4-i-1)}\right)^{4-i-1} \\ &= \frac{1}{2^{4-i}} + \left(\frac{1}{2} + \frac{B_i}{2}\right)\left(\frac{1}{2} + \frac{B_{\ge i+1}}{2(4-i-1)}\right)^{4-i-1}, \end{aligned} \tag{10}$$

where the second and second-to-last inequalities follow from our induction hypothesis. When $i = 2$, the above bound becomes

$$\frac{1}{4} + \left(\frac{1}{2} + \frac{B_2}{2}\right)\left(\frac{1}{2} + \frac{B_3}{2}\right),$$

which is uniquely maximized (given total budget $B_{\ge 2}$ for layers more than 2) if and only if $B_2 = B_3 = B_{\ge 3}$. When $i = 1$, this bound becomes

$$\frac{1}{8} + \left(\frac{1}{2} + \frac{B_1}{2}\right)\left(\frac{1}{2} + \frac{B_2 + B_3}{4}\right)^2,$$

which is similarly uniquely maximized (when $B = B_1 + B_2 + B_3$) if and only if $B_1 = \frac{B_2+B_3}{2}$, i.e. only if $B_1 = B/3$, $B_2 + B_3 = 2B/3$. In both cases, the unique maximizer satisfies $B_i = \frac{B_{\ge i}}{4-i}$ and $\frac{B_{\ge i+1}}{4-i-1} = B_i$. Therefore,

$$\begin{aligned} w_u^i + w_v^i &\le \frac{1}{2^{4-i}} + \left(\frac{1}{2} + \frac{B_{\ge i}}{2(4-i)}\right)\left(\frac{1}{2} + \frac{B_{\ge i}}{2(4-i)}\right)^{4-i-1} \\ &= \frac{1}{2^{4-i}} + \left(\frac{1}{2} + \frac{B_{\ge i}}{2(4-i)}\right)^{4-i}, \end{aligned}$$

and this equality can only be tight when i) $B_i = \frac{B_{\ge i}}{4-i}$ (by the unique maximizer argument above) and ii) the second inequality in Equation (10) is tight, which means

$$B_{\ge i+1} = B_{\ge i} - B_i = \frac{4-i-1}{4-i} B_{\ge i}$$

is split equally across the $4-i-1$ transitions between layers to the right of L_{i+1}, by induction hypothesis. This immediately implies that $B_i = \ldots = B_3 = \frac{B_{\ge i}}{4-i}$ when the inequality is tight.

We conclude our proof by showing an upper bound on w_u^i, w_v^i. By induction hypothesis,

$$w_u^{i+1}, w_v^{i+1} \le \left(\frac{1}{2} + \frac{B_{\ge i+1}}{2(4-i-1)}\right)^{4-i-1}.$$

It immediately implies that

$$w_u^{i+}, w_v^i \leq \left(\frac{1}{2}+\frac{B_i}{2}\right)\left(\frac{1}{2}+\frac{B_{\geq i+1}}{2(4-i-1)}\right)^{4-i-1},$$

noting that the maximum transition probability from either w_u^i or w_v^i to the best of w_u^{i+1}, w_v^{i+1} is at most $\frac{1}{2}+\frac{B_i}{2}$ (half the budget must be spent decreasing other edges, and half the budget B_i is spent increasing transitions to the best node in $\mathcal{I}_1$ on layer $i+1$). By the exact same argument as for the first part of the lemma, this is upper-bounded by $\left(\frac{1}{2}+\frac{B_{\geq i}}{2(4-i)}\right)^{4-i}$. Hence the induction hypothesis holds at layer i. □

D Omitted Proofs

D.1 Proof of Theorem 2: Algorithmic Guarantees for Ex-Post-Fairness

Recall that OPT^{ε}_{MM} is the optimum maximin value under discretized splits of the budget. Let $M_1^{\varepsilon}, \ldots, M_{k-1}^{\varepsilon}$ be a set of transition matrices with expected reward for each starting position i lower-bounded by by $R^\top M_{k-1}^{\varepsilon} \ldots M_1^{\varepsilon} e_i \geq OPT^{\varepsilon}_{MM} \triangleq OPT_{MM} - (k-1)\varepsilon \|R\|_\infty$ that is feasible with respect to budget split $B_1^{\varepsilon}, \ldots, B_{k-1}^{\varepsilon}$. Note that such matrices exist by Claim 1. Let $E_t \in \mathcal{D}^w$ denote the population-wise probability distribution that is induced by these transition matrices on layer t, i.e.

$$E_t^j = M_{t-1}^{\varepsilon} \ldots M_1^{\varepsilon} e_j \;\forall j \in [w].$$

To prove the result, we will show by induction that for all $B_{\geq t} \geq B^{\varepsilon}_{\geq t}$, for $A_t \in \mathcal{A}(\varepsilon)$ such that $\|A_t^j - E_t^j\|_1 \leq \varepsilon \quad \forall j \in [w]$, we have

$$R^\top M(B_{\geq t}, A_t) A_t^j \geq OPT^{\varepsilon}_{MM} - 2(k-t)\varepsilon \|R\|_\infty, \quad \forall j \in [w],$$

i.e., a population-wise welfare approximation guarantee starting at any layer t. Since we can take $B_{\geq 1} = B^{\varepsilon}$, this directly implies

$$R^\top M(B^{\varepsilon}, A_1) e_j \geq OPT^{\varepsilon}_{MM} - 2(k-1)\varepsilon \|R\|_\infty \;\forall j \in [w].$$

Combined with Claim 1, which states that $OPT^{\varepsilon}_{MM} \geq OPT_{MM} - (k-1)\varepsilon \|R\|_\infty$, we obtain the result.

Let us now provide our inductive proof. First, consider the transition from layer L_{k-1} to layer L_k. Note that

$$OPT^{\varepsilon}_{MM} \leq R^\top M_{k-1}^{\varepsilon} \ldots M_1^{\varepsilon} e_j = R^\top M_{k-1}^{\varepsilon} E_{k-1}^j \quad \forall j \in [w]$$

using the fact that $E_{t+1}^j = M_t^{\varepsilon} E_t^j$. Let $A_{k-1} \in \mathcal{A}(\varepsilon)$ be such that $\|A_{k-1}^j - E_{k-1}^j\| \leq \varepsilon \quad \forall j \in [w]$. Note that such a A_{k-1} always exists (by definition of $\mathcal{A}(\varepsilon)$), and is considered by Algorithm 2. By Corollary 2,

$$R^\top M_{k-1}^{\varepsilon} A_{k-1}^j \geq R^\top M_{k-1}^{\varepsilon} E_{k-1}^j - \varepsilon \|R\|_\infty \geq OPT^{\varepsilon}_{MM} - \varepsilon \|R\|_\infty \;\; \forall j \in [w].$$

Further, M_{k-1}^{ε} is feasible for Program (8) with respect to $B_{\geq k-1}, B_{\geq k} = 0$, given $B_{\geq k-1} \geq B^{\varepsilon}_{\geq k-1}$. As such, for $B_{\geq k-1} \geq B^{\varepsilon}_{\geq k-1}$, by optimality of $M(B_{\geq k-1}, A_{k-1})$, we have that

$$\min_{j \in [w]} R^\top M(B_{\geq k-1}, A_{k-1}) A_{k-1}^j \geq \min_{j \in [w]} R^\top M_{k-1}^{\varepsilon} A_{k-1}^j,$$

and in turn

$$R^\top M(B_{\geq k-1}, A_{k-1}) A_{k-1}^j \geq OPT_{MM}^\varepsilon - \varepsilon \|R\|_\infty \;\; \forall j \in [w].$$

Now, suppose the induction hypothesis holds at layer $t+1$. I.e., for all $B_{\geq t+1} \geq B_{\geq t+1}^\varepsilon$, for $A_{t+1} \in \mathcal{A}(\varepsilon)$ such that $\|A_{t+1}^j - E_{t+1}^j\|_1 \leq \varepsilon \;\; \forall j \in [w]$,

$$R^\top M(B_{\geq t+1}, A_{t+1}) A_{t+1}^j \geq OPT_{MM}^\varepsilon - 2(k-t-1)\varepsilon \|R\|_\infty \;\; \forall j \in [w].$$

For any given $B_{\geq t} \geq B_{\geq t}^\varepsilon$, note that one can set $B_{\geq t+1} = B_{\geq t+1}^\varepsilon$ and have $B_t \geq B_t^\varepsilon$; hence, M_t^ε is feasible for Program (8) with respect to $B_t \geq B_t^\varepsilon, B_{\geq t+1}^\varepsilon$. Consider $A_t \in \mathcal{A}(\varepsilon)$ such that $\|A_t^j - E_t^j\|_1 \leq \varepsilon \;\; \forall j \in [w]$. Note that such a A_t always exists (by definition of $\mathcal{A}(\varepsilon)$, and is considered by Algorithm 2. Since we have $\|A_t^j - E_t^J\|_1 \leq \varepsilon$ and $\|A_{t+1}^j - M_t^\varepsilon E_t^j\|_1 \leq \varepsilon \; \forall j \in [w]$, applying Corollary 2 yields that $\forall j \in [w]$,

$$\begin{aligned} R^\top M(B_{\geq t+1}^\varepsilon, A_{t+1}) M_t^\varepsilon A_t^j &\geq R^\top M(B_{\geq t+1}^\varepsilon, A_{t+1}) M_t^\varepsilon E_t^j - \varepsilon \|R\|_\infty \\ &\geq R^\top M(B_{\geq t+1}^\varepsilon, A_{t+1}) A_{t+1}^j - 2\varepsilon \|R\|_\infty. \end{aligned}$$

Using the induction hypothesis, we obtain that

$$R^\top M(B_{\geq t+1}^\varepsilon, D_{t+1}) M_t^\varepsilon A_t^j \geq OPT_{MM}^\varepsilon - 2(k-t)\varepsilon \|R\|_\infty \;\; \forall j \in [w],$$

which can be rewritten as

$$\min_{j \in [w]} R^\top M(B_{\geq t+1}^\varepsilon, D_{t+1}) M_t^\varepsilon A_t^j \geq OPT_{MM}^\varepsilon - 2(k-t)\varepsilon \|R\|_\infty.$$

In particular, by optimality of $M(B_{\geq t}, A_t)$, it must be the case that

$$\min_{j \in [w]} R^\top M(B_{\geq t}, A_t) A_t^j \geq OPT_{MM}^\varepsilon - 2(k-t)\varepsilon \|R\|_\infty,$$

which concludes the proof of the accuracy guarantee. The running time is obtained noting that at each time step t, we solve one Program 8 for each of the (at most) $\frac{B}{\varepsilon}$ possible budget splits of $B_{\geq t}$ and for each of the $\left(\left(\frac{1}{\varepsilon}\right)^w\right)^w = \left(\frac{1}{\varepsilon}\right)^{w^2}$ population-wise probability distributions in $\mathcal{A}(\varepsilon)$ on both layer L_t and layer L_{t+1}; i.e., in a given time step, the algorithm solves $O\left(\frac{B}{\varepsilon}\left(\frac{1}{\varepsilon}\right)^{w^4}\right)$ optimization programs. Then, the algorithm finds the solution of all of these programs with the best objective value, which can be done in time linear in the number of such solutions, i.e. $O\left(\frac{B}{\varepsilon}\left(\frac{k}{\varepsilon}\right)^{w^4}\right)$. The algorithm does so over $k-1$ time steps.

D.2 Proof of Lemma 2: Algorithmic Guarantees for Ex-Ante Fairness

The proof follows that of Theorem 1 of Freund and Schapire [1996]. Note that we can rewrite the objective in the normal form given in Corollary 2 of Freund and Schapire [1996], by letting the payoff matrix G be such that $G\left((M_1, \ldots, M_{k-1}), q\right) = R^\top M_{k-1} \ldots M_1 e_q$ when the designer plays $(M_1, \ldots, M_{k-1}) \in \mathcal{F}$ and the learner plays $q \in [w]$. Noting that G has entries bounded between 0 and $\|R\|_\infty$, we can apply Corollary 2 of Freund and Schapire [1996] to loss $R^\top M_{k-1}^t \ldots M_1^t D^t$ with an appropriate renormalization to show the following low-regret statement:

$$\frac{1}{T}\sum_{t=1}^T R^\top M_{k-1}^t \ldots M_1^t D^t \leq \min_{D \in \mathcal{D}} \frac{1}{T}\sum_{t=1}^T R^\top M_{k-1}^t \ldots M_1^t D + \left(\sqrt{2\frac{\ln w}{T}} + \frac{\ln w}{T}\right) \|R\|_\infty,$$

Since $R^\top M_{k-1}^t \dots M_1^t D$ is linear in D, we have

$$\min_{D\in\mathcal{D}} \frac{1}{T}\sum_{t=1}^{T} R^\top M_{k-1}^t \dots M_1^t D = \min_{q\in[w]} \frac{1}{T}\sum_{t=1}^{T} R^\top M_{k-1}^t \dots M_1^t e_q,$$

and the following low-regret statement

$$\frac{1}{T}\sum_{t=1}^{T} R^\top M_{k-1}^t \dots M_1^t D^t \leq \min_{q\in[w]} \frac{1}{T}\sum_{t=1}^{T} R^\top M_{k-1}^t \dots M_1^t e_q + \left(\sqrt{2\frac{\ln w}{T}} + \frac{\ln w}{T}\right)\|R\|_\infty. \tag{11}$$

We can now show the result, using a similar argument to that of Theorem 1 of Freund and Schapire [1996]. To do so, we let $\bar{D} \in \mathcal{D}$ be the probability distribution given by $\bar{D} \triangleq \frac{1}{T}\sum_{t=1}^{T} D^t$. We have that

$$\begin{aligned}
&\min_{q\in[w]} R^\top \mathbb{E}_{M\sim\overline{\Delta M}} \mathbb{E}\left[M_{k-1}\dots M_1\right] e_q \\
&= \min_{q\in[w]} \frac{1}{T}\sum_{t=1}^{T} R^\top M_{k-1}^t \dots M_1^t e_q && \text{(by definition of } \overline{\Delta M}\text{)} \\
&\geq \frac{1}{T}\sum_{t=1}^{T} R^\top M_{k-1}^t \dots M_1^t D^t - \left(\sqrt{2\frac{\ln w}{T}} + \frac{\ln w}{T}\right)\|R\|_\infty && \text{(by Equation (11))} \\
&\geq \frac{1}{T}\sum_{t=1}^{T} \max_{M\in\mathcal{F}} R^\top M_{k-1} \dots M_1 D^t - \varepsilon - \left(\sqrt{2\frac{\ln w}{T}} + \frac{\ln w}{T}\right)\|R\|_\infty \\
& && (M^t\ \varepsilon\text{-approx. best response to } D^t) \\
&\geq \max_{M\in\mathcal{F}} \frac{1}{T}\sum_{t=1}^{T} R^\top M_{k-1} \dots M_1 D^t - \varepsilon - \left(\sqrt{2\frac{\ln w}{T}} + \frac{\ln w}{T}\right)\|R\|_\infty \\
& && \text{(Max of sum less than sum of max)} \\
&\geq \max_{\Delta M\in\Delta\mathcal{F}} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}_{M\sim\Delta M}\left[R^\top M_{k-1} \dots M_1 D^t\right] - \varepsilon - \left(\sqrt{2\frac{\ln w}{T}} + \frac{\ln w}{T}\right)\|R\|_\infty \\
& && \text{(Expectation over } \Delta M \text{ less than best realization of } \Delta M\text{, and the realization is in } \mathcal{F}\text{)} \\
&= \max_{\Delta M\in\Delta\mathcal{F}} R^\top \mathbb{E}_{M\sim\Delta M}\left[M_{k-1}\dots M_1\right] \bar{D} - \varepsilon - \left(\sqrt{2\frac{\ln w}{T}} + \frac{\ln w}{T}\right)\|R\|_\infty \\
&\geq \min_{D\in\mathcal{D}} \max_{M\in\Delta\mathcal{F}} R^\top \mathbb{E}_{M\sim\Delta M}\left[M_{k-1}\dots M_1\right] D - \varepsilon - \left(\sqrt{2\frac{\ln w}{T}} + \frac{\ln w}{T}\right)\|R\|_\infty \\
&\geq \max_{\Delta M\in\Delta\mathcal{F}} \min_{D\in\mathcal{D}} R^\top \mathbb{E}_{M\sim\Delta M}\left[M_{k-1}\dots M_1\right] D - \varepsilon - \left(\sqrt{2\frac{\ln w}{T}} + \frac{\ln w}{T}\right)\|R\|_\infty \\
& && \text{(Max-min inequality)} \\
&= \max_{\Delta M\in\Delta\mathcal{F}} \min_{q\in[w]} R^\top \mathbb{E}_{M\sim\Delta M}\left[M_{k-1}\dots M_1\right] e_q - \varepsilon - \left(\sqrt{2\frac{\ln w}{T}} + \frac{\ln w}{T}\right)\|R\|_\infty.
\end{aligned}$$

This concludes the proof.

D.3 Omitted Proofs for Price of Fairness

D.3.1 Proof of the Lower Bound of Theorem 3

Note that $P_f \geq 1$ is always true, by definition. The proof of the other two cases when $B \leq 2w$ is based on Example 1. We divide the analysis of the construction into the following cases:

1. $B \leq 2$. It is easy to see that $OPT_{SW} = \frac{B}{2}(1-(w-1)\varepsilon)$, and is achieved by the following transition matrix:

$$M_1^* = \begin{pmatrix} B/2 & 0 & \dots & 0 \\ 1-B/2 & 1 & \dots & 1 \end{pmatrix}$$

Now note that the maximin solution is unique (and in particular, is the maximim solution with the highest social welfare), and this unique maximin solution splits the budget evenly among the starting nodes and yields social welfare $\frac{B}{2w}$, via transition matrix

$$M_1^f = \begin{pmatrix} \frac{B}{2w} & \dots & \frac{B}{2w} \\ 1-\frac{B}{2w} & \dots & 1-\frac{B}{2w} \end{pmatrix}$$

Therefore, we have that

$$P_f^+(\varepsilon) = \frac{B(1-w\varepsilon)/2}{\frac{B}{2w}} = w(1-w\varepsilon),$$

and

$$\lim_{\varepsilon \to 0} P_f(\varepsilon) = w.$$

2. Now, consider the case when $2 \leq B \leq 2w$. On the one hand, note that $OPT_{SW} \geq 1-(w-1)\varepsilon$, as setting

$$M_1^* = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 1 \end{pmatrix}$$

only requires a budget of 2 hence is feasible for Program (1). The unique maximin solution is still given by M_1^f and has welfare $\frac{B}{2w}$. As such, we have

$$P_f(\varepsilon) \geq \frac{1-w\varepsilon}{\frac{B}{2w}} = 2w\frac{1-(w-1)\varepsilon}{B}.$$

In particular, taking $\varepsilon \to 0$, we get that a lower bound on the price of fairness is given by $\frac{2w}{B}$.

The proof for $B \geq 2w$ is immediate, noting that

$$M_1^* = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 0 & 0 & \dots & 0 \end{pmatrix}$$

is feasible. As such $OPT_{SW} = 1$, and M_1^* is a maximin solution with welfare 1.

D.3.2 Proof of Lemma 3

Proof. The first part of the claim is immediate from noting that given an optimal solution $M_1^*, \dots, M_{k-1}^*$ to Program (1),

$$OPT_{SW} = R^\top M_{k-1}^* \dots M_1^* D_1^0 \leq \|R\|_\infty \left\| M_{k-1}^* \dots M_1^* D_1^0 \right\| = \|R\|_\infty ,$$

where the last equality follows from $M_{k-1}^* \dots M_1^* D_1^0$ being a probability distribution.

For the second part of the claim, consider any feasible solution $M_1, \dots, M_{k-1}$ with corresponding split $B_1, \dots, B_{k-1}$ of the budget. I.e., $B = \sum_{t=1}^{k-1} B_t$, and $\sum_i \sum_j \left| M_t(i,j) - M_t^0(i,j) \right| \leq B$ for all t. Note that at layer L_t, for any input distribution D_t, and vector R_{t+1} with non-negative coordinates at layer $t+1$, we have that

$$\begin{aligned} R_{t+1}^\top \left(M_t - M_t^0 \right) D_t &= \sum_{i=1}^{w} R_{t+1}(i) \sum_{j=1}^{w} \left(M_t(i,j) - M_t^0(i,j) \right) D_t(j) \\ &= \sum_{j=1}^{w} D_t(j) \sum_{i=1}^{w} R_{t+1}(i) \left(M_t(i,j) - M_t^0(i,j) \right) \\ &\leq \sum_{j=1}^{w} D_t(j) \sum_{i \in S_j} R_{t+1}(i) \left(M_t(i,j) - M_t^0(i,j) \right) \\ &\leq \|R_{t+1}\|_\infty \sum_{j=1}^{w} D_t(j) \sum_{i \in S_j} M_t(i,j) - M_t^0(i,j \end{aligned}$$

where $S_j = \{ i : \; M_t(i,j) - M_t^0(i,j) \geq 0 \}$ and where the second-to-last inequality follows from the fact that $R_{t+1}(i) \geq 0$. As $M_t, M_t^0 \in \mathcal{M}$, we have that

$$\sum_{i=1}^{w} M_t(i,j) - M_t^0(i,j) = 1 - 1 = 0,$$

which implies that

$$\sum_{i \in S_j} M_t(i,j) - M_t^0(i,j) = \sum_{i=1}^{w} M_t(i,j) - M_t^0(i,j) - \sum_{i \notin S_j} M_t(i,j) - M_t^0(i,j) = - \sum_{i \notin S_j} M_t(i,j) - M_t^0(i,j).$$

In turn, we have that

$$\begin{aligned} \frac{B_t}{2} \geq \frac{c(M_t, M_t^0)}{2} &= \frac{1}{2} \sum_j \sum_i \left| M_t(i,j) - M_t^0(i,j) \right| \\ &= \frac{1}{2} \sum_j \left(\sum_{i \in S_j} M_t(i,j) - M_t^0(i,j) - \sum_{i \notin S_j} M_t(i,j) - M_t^0(i,j) \right) \\ &= \sum_j \sum_{i \in S_j} M_t(i,j) - M_t^0(i,j). \end{aligned}$$

This can also be seen noting that increasing edges of M_t^0 by a total amount of δB_t requires decreasing other edges by a total amount of δB_t for M_t to be a stochastic matrix and so requires a total budget of $2\delta B_t$, which in turn implies that $\delta \leq \frac{1}{2}$ necessarily. This implies that

$$\begin{aligned} R_{t+1}^\top \left(M_t - M_t^0\right) D_t &\leq \|R_{t+1}\|_\infty \sum_{j=1}^{w} D_t(j) \sum_{i \in S_j} M_t(i,j) - M_t^0(i,j) \\ &\leq \|R_{t+1}\|_\infty \sum_{j=1}^{w} \sum_{i \in S_j} M_t(i,j) - M_t^0(i,j) \\ &\leq \frac{B_t}{2} \|R_{t+1}\|_\infty \end{aligned}$$

Applying the above inequality recursively, we have that

$$\begin{aligned} R^\top M_{k-1} \dots M_1 D_1^0 &\leq \frac{B_{k-1}}{2} \|R\|_\infty + R^\top M_{k-1}^0 M_{k-2} \dots M_1 D_1^0 \\ &\leq \frac{B_{k-1} + B_{k-2}}{2} \|R\|_\infty + R^\top M_{k-1}^0 M_{k-2}^0 M_{k-3} \dots M_1 D_1^0 \\ &\quad \vdots \\ &\leq \frac{\sum_t B_t}{2} \|R\|_\infty + R^\top M_{k-1}^0 \dots M_1^0 D_1^0 \\ &= \frac{B}{2} + W^0. \end{aligned}$$

□

D.3.3 Proof of Lemma 4

First, we consider the case when $\frac{B}{2w} \leq 1$. We focus on the transition from the second-to-last layer L_{k-1} to the last layer L_k. Remember that on layer L_k, $R(1) = \|R\|_\infty$. We re-number (w.l.o.g.) the nodes on layer $k-1$ so that $M_{k-1}^0(1,i) \geq \frac{B}{2w}$ for all nodes $i \in [l]$, and $M_{k-1}^0(1,i) < \frac{B}{2w}$ for all nodes $i \in \{l+1, \dots, k\}$, for some $l \in \{0, \dots, w\}$. Let us set

$$M_{k-1} = M_{k-1}^0 + \begin{pmatrix} 0 & 0 & \dots & 0 & \frac{B}{2w} - M_{k-1}^0(1, l+1) & \dots & \frac{B}{2w} - M_{k-1}^0(1, w) \\ 0 & 0 & \dots & 0 & -\alpha_{2,l+1} & \dots & -\alpha_{2,w} \\ \vdots & \vdots & \vdots & 0 & \vdots & \dots & \vdots \\ 0 & 0 & \dots & 0 & -\alpha_{w,l+1} & \dots & -\alpha_{w,w} \end{pmatrix},$$

and

$$M_t = M_t^0 \ \forall t < k-1,$$

where the $\alpha_{i,j}$'s are chosen to guarantee $M_{k-1}^0(i,j) \geq \alpha_{i,j} \geq 0$ for all $i,j \in [w]$ and $\sum_{i=2}^{w} \alpha_{i,j} = \frac{B}{2w} - M_{k-1}^0(1,j) > 0$ for all $j > l$. Such a choice of $\alpha_{i,j}$'s exists as

$$\sum_{j=2}^{w} M_{k-1}^0(i,j) = 1 - M_{k-1}^0(1,j) \geq /(2w) - M_{k-1}^0(1,j).$$

Now, note that $M_{k-1} \in \mathcal{M}$, as all coefficients are between 0 and 1 and the elements within the same column still sum to 1 by choice of $\alpha_{i,j}$'s. Finally,

$$c(M_{k-1}, M_{k-1}^0) = \sum_{j=l+1}^{w} 2\left(\frac{B}{2w} - M_{k-1}^0(1,j)\right) \leq \sum_{j=l+1}^{w} 2\frac{B}{2w} \leq B.$$

Therefore, $(M_{k-1}, \ldots, M_1)$ is a feasible solution for Programs 1 and 2 under budget B.

Now, note that by construction, $M_{k-1}(1,j) \geq \frac{B}{2w}$ for all $j \in [w]$. In turn, this implies that for any distribution D_{k-1},

$$\begin{aligned} R^\top M_{k-1} D_{k-1} = \sum_{i=1}^{w} R(i) \left(M_{k-1} D_{k-1}\right)(i) &= \sum_{i=1}^{w} R(i) \sum_{j=1}^{w} M_{k-1}(i,j) D_{k-1}(j) \\ &\geq R(1) \sum_{j=1}^{w} M_{k-1}(1,j) D_{k-1}(j) \\ &\geq R(1) \frac{B}{2w} \sum_{j=1}^{w} D_{k-1}(j) \\ &= \frac{B}{2w} \left\| R \right\|_\infty, \end{aligned}$$

since $\sum_{j=1}^{w} D_{k-1}(j) = 1$ by virtue of D_{k-1} being a probability distribution, and because $R(1) = \left\| R \right\|_\infty$ by choice of node indexing. In particular, for all j, $D_{k-1} = M_{k-2} \ldots M_1 e_j$ is a probability distribution, hence

$$R^\top M_{k-1} \ldots M_1 e_j \geq \frac{B}{2w} \left\| R \right\|_\infty.$$

Therefore, there exists a solution to Program (2) that has value $\frac{B}{2w} \left\| R \right\|_\infty$, implying any optimal solution to Program (2) has value at least $\frac{B}{2w} \left\| R \right\|_\infty$. In turn, such an optimal solution $M_1^f, \ldots, M_{k-1}^f \in S^f$ must have welfare

$$R^\top M_{k-1}^f \ldots M_1^f D_1^0 = \sum_{j=1}^{w} D_1^0(j) R^\top M_{k-1}^f \ldots M_1^f e_j \geq \sum_{j=1} D_1^0(j) \frac{B}{2w} \left\| R \right\|_\infty = \frac{B}{2w} \left\| R \right\|_\infty,$$

which concludes the proof when $B \leq 2w$.

When $B > 2w$, let $B' = 2w$. By the above, any optimal solution to maximin Program (2) with budget $B' = 2w$ has value at least $\frac{B'}{2w} \left\| R \right\|_\infty = \left\| R \right\|_\infty$. This immediately implies that any optimal solution to Program (2) with budget B also has value at least $\left\| R \right\|_\infty$, since any feasible solution for budget B' is feasible for budget B. In turn, any optimal solution to maximin Program (2) under budget $B > 2w$ must have welfare at least $\left\| R \right\|_\infty$.

D.3.4 Proof of Lemma 5

The proof of the lemma uses the following Claim 4, that shows that an optimal solution to Program (2) spends all the budget B:

Claim 4. *Let $(M_1, \ldots, M_{k-1})$ be a solution to maximin Program (2) with social welfare strictly less than $\left\| R \right\|_\infty$. If all edges are malleable, it must be the case that $\sum_{t=1}^{k-1} c(M_t, M_t^0) = B$.*

The proof idea is simple: if $\sum_{t=1}^{k-1} c(M_t, M_t^0) < B$, the leftover budget can be used to improve the scial welfare, unless this social welfare already is the maximum achievable value of $\|R\|_\infty$.

Proof. By contradiction, suppose that $B > \sum_{t=1}^{k-1} c(M_t, M_t^0)$. Remember the numbering of nodes on layer L_k is chosen such that $R(1) = \|R\|_\infty$. Let us pick all j such that $M_{k-1}(1,j) < 1$ (if such a j exists); note that since M_{k-1} is stochastic, there also exists $q \neq 1$ such that $M_{k-1}(q,j) > 0$. For ε arbitrarily small, there hence exists a matrix $M'_{k-1}(\varepsilon) \in \mathcal{M}$ such that for all such j, $M'_{k-1}(1,j) = M_{k-1}(1,j) + \varepsilon$ and $\sum_{q \neq 1} M'_{k-1}(q,j) = \sum_{q \neq 1} M_{k-1}(q,j) - \varepsilon$, and such that $M'_{k-1}(q,j) = M_{k-1}(q,j)$ for all $q \in [w]$ and all j with $M_{k-1}(1,j) = 1$.

Now, note that $c(M'_{k-1}, M^0_{k-1}) \leq c(M'_{k-1}, M_{k-1}) + c(M_{k-1}, M^0_{k-1})$ with $\lim_{\varepsilon \to 0} c(M'_{k-1}, M_{k-1}) = 0$. In turn, this implies that for ε small enough, $\sum_{t=1}^{k-2} c(M_t, M_t^0) + c(M'_t, M_t^0) \leq B$, hence $\left(M_1, \dots, M_{k-2}, M'_{k-2}\right)$ is feasible for Program (2). Further, by construction, for all j with $M_{k-1}(1,j) < 1$, we have

$$R^\top M'_{k-1} e_j - R^\top M_{k-1} e_j \geq \left(R(1) - \max_{q \neq 1} R(q) \right) \varepsilon > 0,$$

and for all j with $M_{k-1}(1,j) = 1$, we have $R^\top M'_{k-1} e_j = R^\top M_{k-1} e_j$. Since for all starting nodes $i \in L_1$ such that $R^\top M_{k-1} \dots M_1 e_i < \|R\|_\infty$, there must exist j such that $M_{k-1}(1,j) < 1$ and $\left(M_{k-2} \dots M_1\right)(j,i) > 0$ (otherwise $\left(M_{k-1} \dots M_1\right)(1,i) = 1$ and node i obtains reward $\|R\|_\infty$), it immediately follows that for all such i,

$$R^\top M'_{k-1} M_{k-2} \dots M_1 e_i > R^\top M_{k-1} M_{k-2} \dots M_1 e_i.$$

This contradicts $(M_1, \dots, M_{k-1})$ being an optimal solution for Program (2). □

We are now ready to prove Lemma 5. Let $M_1^f, \dots, M_{k-1}^f$ be an optimal solution for Program (2). Note that if the maximin value of this solution is $\|R\|_\infty$, then the solution necessarily has welfare $\|R\|_\infty \geq W^0$, which concludes the proof. So, without loss of generality, we can assume there exists at least one starting node q such that

$$R^\top M_{k-1}^f \dots M_1^f e_q < \|R\|_\infty .$$

Fix a layer t, and let $R_{out}^\top \triangleq R^\top M_k^f \dots M_{t+1}^f$ and $M_{in} = M_{t-1}^f \dots M_1^f$. Note that the utility obtained by the q-th node in the starting layer is immediately given by

$$R_i = \sum_{i=1}^{w} M_{in}(i,q) \sum_{j=1}^{w} M_t^f(j,i) R_{out}(j).$$

Indeed, starting from node q in the first layer, an individual transitions to node i on layer t with probability $M_{in}(i,q)$, then to node j with reward $R_{out}(j)$ on layer $t+1$ with probability $M_t(j,i)$.

Suppose by contradiction that for some i', $\sum_{j=1}^{w} M_t^f(j,i') R_{out}(j) < \sum_{j=1}^{w} M_t^0(j,i') R_{out}(j)$ (necessarily, $M_t^f(j,i') \neq M_t^0(j,i')$ for some j). We will construct a set of transition matrices that achieves the same maximin value, but requires budget strictly less than B. To do so, let M'_t be such that $M'_t(j,i') = M_t^0(j,i')$ for all j and $M'_t(j,i) = M_t^f(j,i)$ for all $i \neq i'$, for all j. First,

$c(M'_t, M^0_t) < c(M^f_t, M^0_t)$, as

$$\begin{aligned}\sum_{i,j} \left|M'_t(j,i) - M_0(j,i)\right| &= \sum_{i \neq i', j} \left|M^f_t(j,i) - M_0(j,i)\right| \\ &< \sum_{i \neq i', j} \left|M^f_t(j,i) - M_0(j,i)\right| + \sum_j \left|M^f_t(j,i') - M_0(j,i')\right| \\ &= c(M_t, M^0_t)\end{aligned}$$

where the strict inequality follows from the fact that $M^f_t(j,i') \neq M^0_t(j,i')$ for some j. Second, for all q, we immediately have that as the $M_{in}(i,q)$ are non-negative,

$$\sum_{i=1}^{w} M_{in}(i,q) \sum_{j=1}^{w} M'_t(j,i) R_{out}(j) \geq \sum_{i=1}^{w} M_{in}(i,q) \sum_{q=1}^{w} M^f_t(j,i) R_{out}(j),$$

since for all $i \neq i'$ we have $\sum_{j=1}^w M'_t(j,i)R_{out}(j) = \sum_{j=1}^w M^f_t(j,i)R_{out}(j)$, and by construction $\sum_{j=1}^w M'_t(j,i')R_{out}(j) = \sum_{j=1}^w M^0_t(j,i')R_{out}(j) > \sum_{j=1}^w M^f_t(j,i')R_{out}(j)$ for i'. In particular, this implies that $(M^f_1, \ldots, M^f_{t-1}, M'_t, M^f_{t+1}, \ldots, M^f_{k-1})$ is an optimal solution to Program (2) that uses budget strictly less than B. This contradicts Claim 4, that shows that the leftover budget can then be used to increase the optimal value of Program 2, implying that M^f cannot be an optimal solution. Therefore, it must be the case that for all $i \in [w]$, for all $t \in [k-1]$,

$$\sum_{j=1}^{w} M^f_t(j,i) R^\top M^f_{k-1} \ldots M^f_{t+1} e_j \geq \sum_{j=1}^{w} M^0_t(j,i) R^\top M^f_{k-1} \ldots M^f_{t+1} e_j,$$

or equivalently

$$R^\top M^f_{k-1} \ldots M^f_{t+1} M^f_t e_i \geq R^\top M^f_{k-1} \ldots M^f_{t+1} M^0_t e_i. \tag{12}$$

Applying this with $t = 1$, we have that for all starting q on layer L_1,

$$R^\top M^f_{k-1} \ldots M^f_1 e_q \geq R^\top M^f_{k-1} \ldots M^f_2 M^0_1 e_q.$$

Now, suppose by induction that for all $i \in [w]$,

$$R^\top M^f_{k-1} \ldots M^f_1 e_i \geq R^\top M^f_{k-1} \ldots M^f_t M^0_{t-1} \ldots M^0_1 e_i.$$

It follows that

$$\begin{aligned}R^\top M^f_{k-1} \ldots M^f_1 e_q &\geq R^\top M^f_{k-1} \ldots M^f_t M^0_{t-1} \ldots M^0_1 e_q \\ &= R^\top M^f_{k-1} \ldots M^f_t \sum_i \left(M^0_{t-1} \ldots M^0_1 e_q\right)(i) e_i \\ &= \sum_i \left(M^0_{t-1} \ldots M^0_1 e_q\right)(i) R^\top M^f_{k-1} \ldots M^f_t e_i \\ &\geq \sum_i \left(M^0_{t-1} \ldots M^0_1 e_q\right)(i) R^\top M^f_{k-1} \ldots M^0_t e_i \\ &= R^\top M^f_{k-1} \ldots M^f_{t+1} M^0_t \ldots M^0_1 e_q\end{aligned}$$

where the second-to-last equation follows from Equation (12). Therefore, by induction, we have that for all q,

$$R^\top M_{k-1}^f \dots M_1^f e_q \geq R^\top M_{k-1}^0 \dots M_1^0 e_q,$$

directly implying that

$$R^\top M_{k-1}^f \dots M_1^f D_1^0 \geq R^\top M_{k-1}^0 \dots M_1^0 e_i D_1^0 = W^0.$$