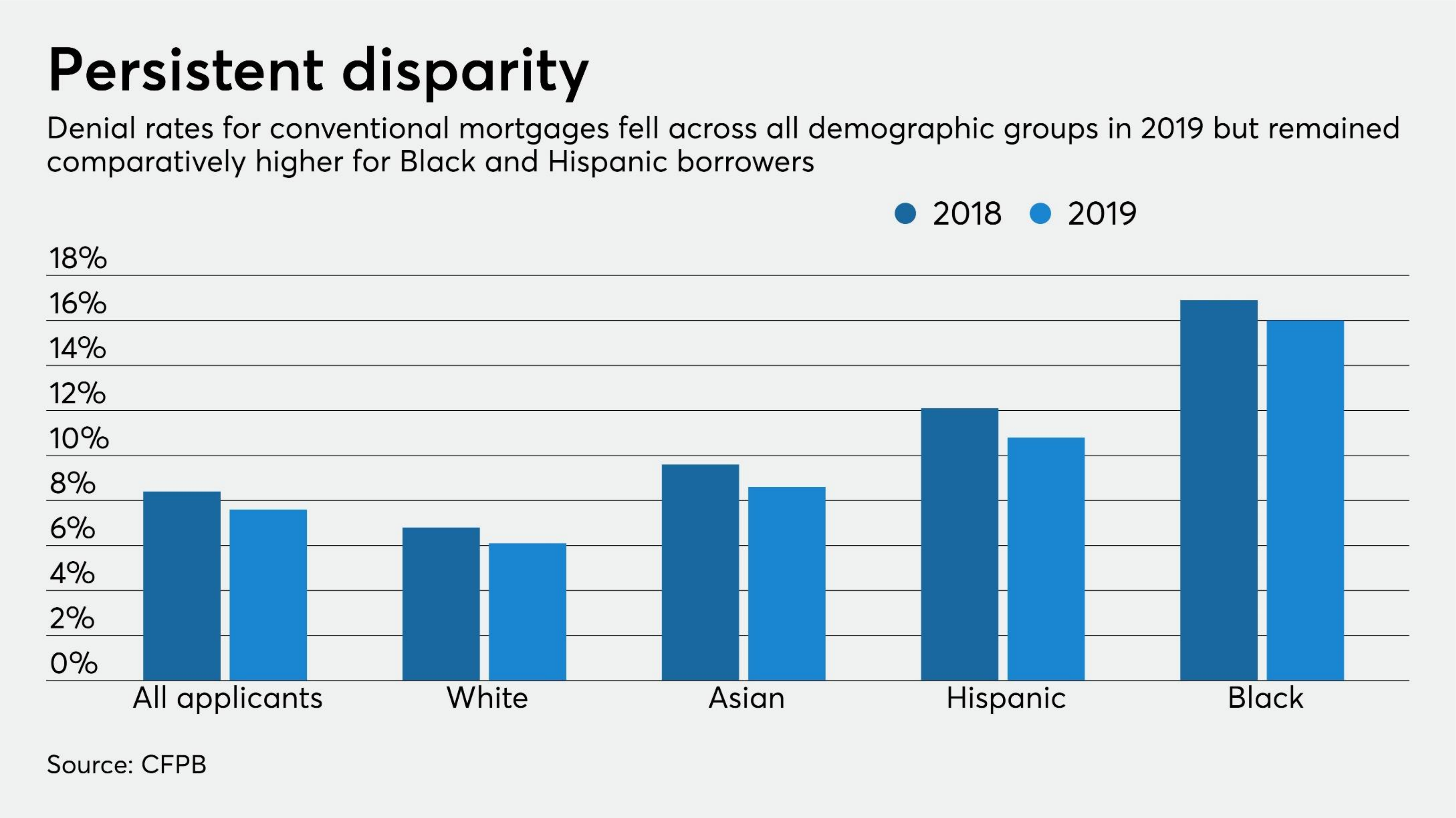
Pipeline Interventions

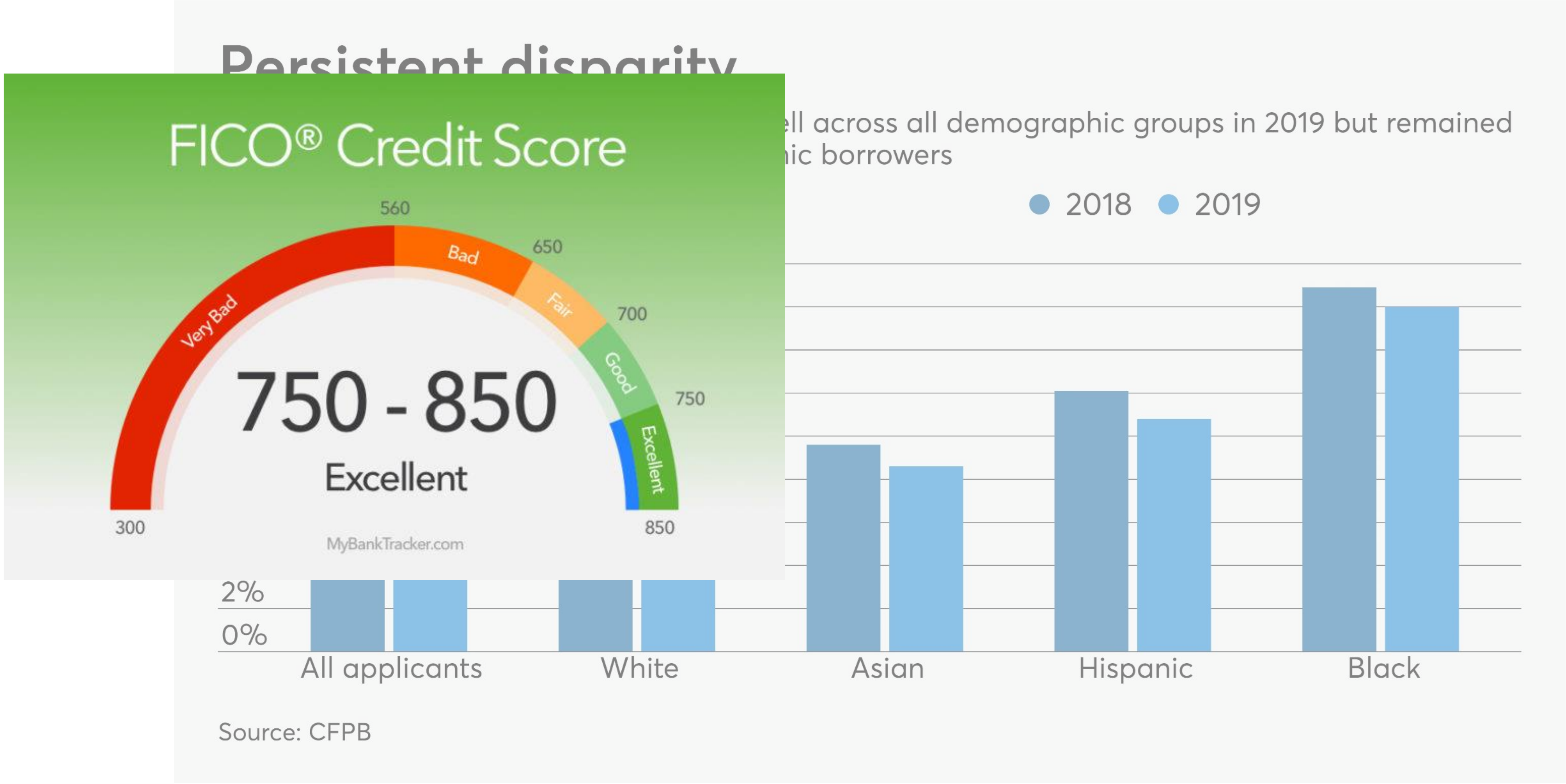
Eshwar Ram Arunachaleswaran*
Sampath Kannan*
Aaron Roth*
Juba Ziani#

**University of Pennsylvania*
#Georgia Tech

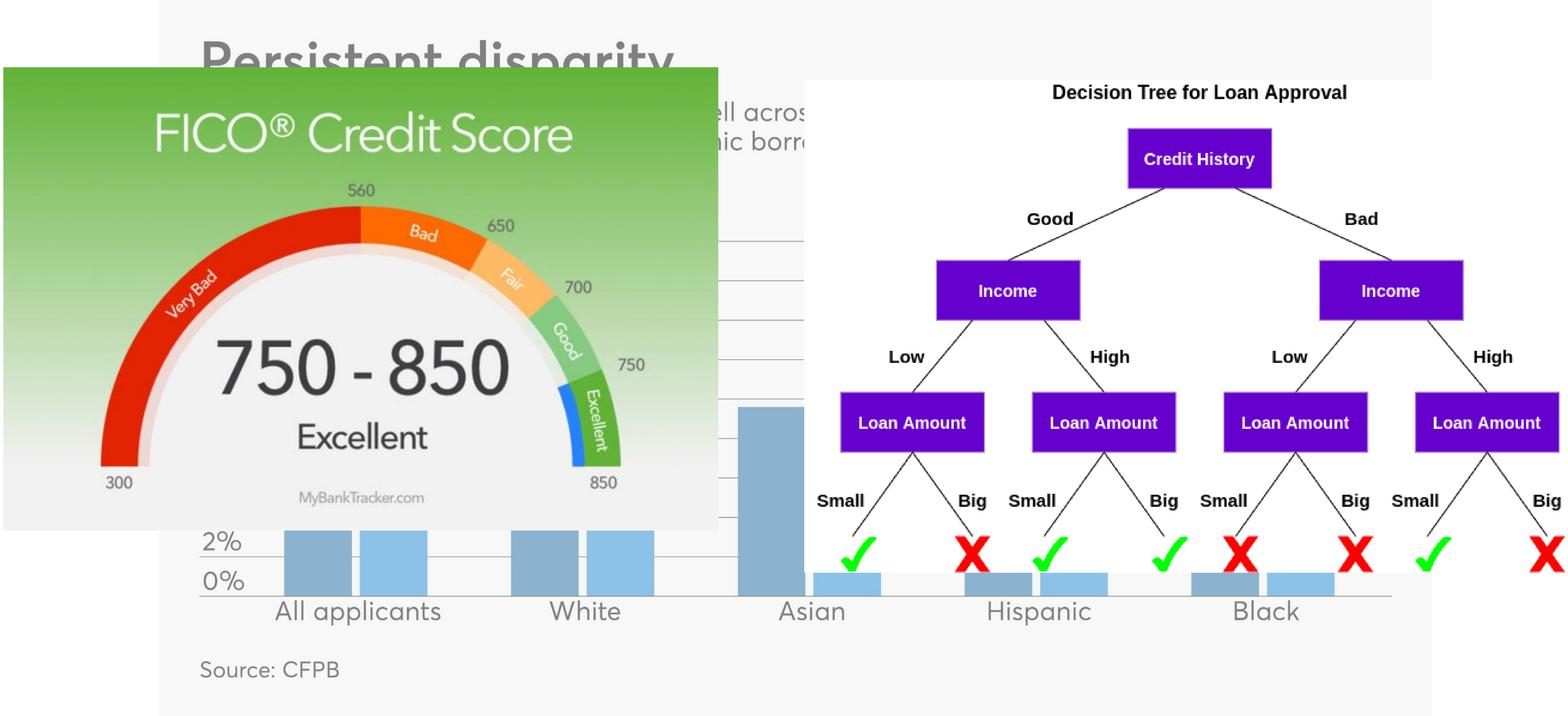
Algorithmic Fairness



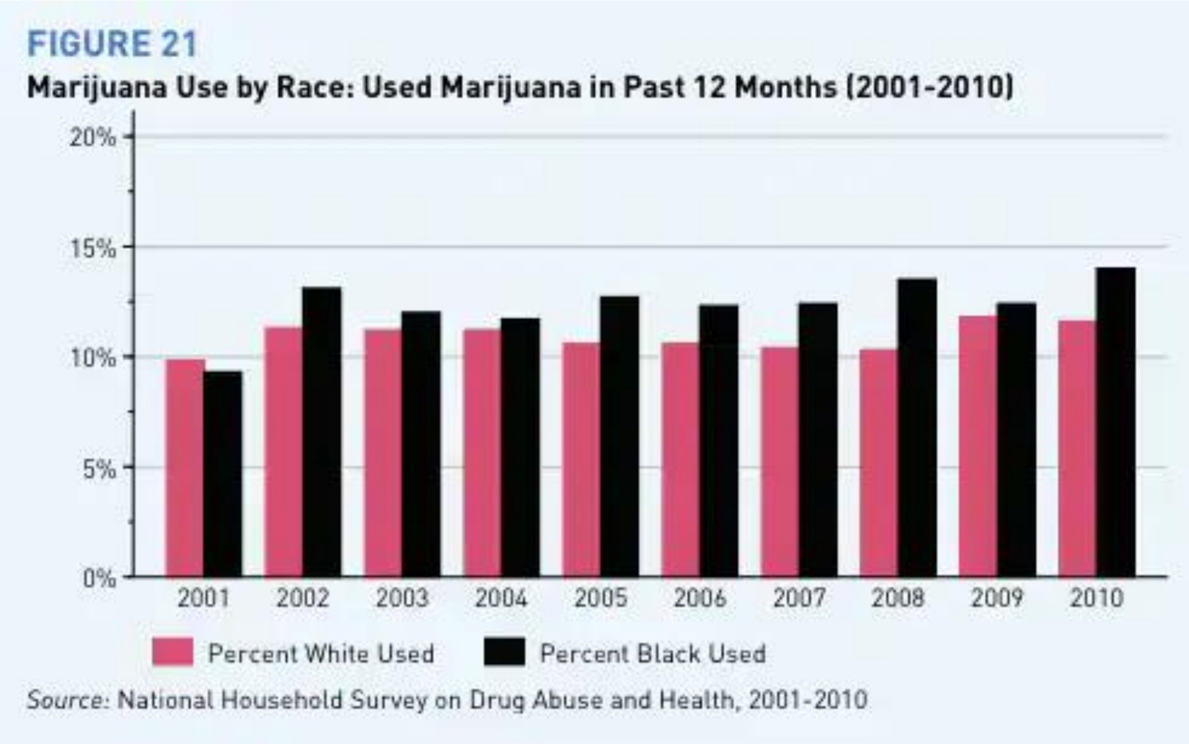
Algorithmic Fairness

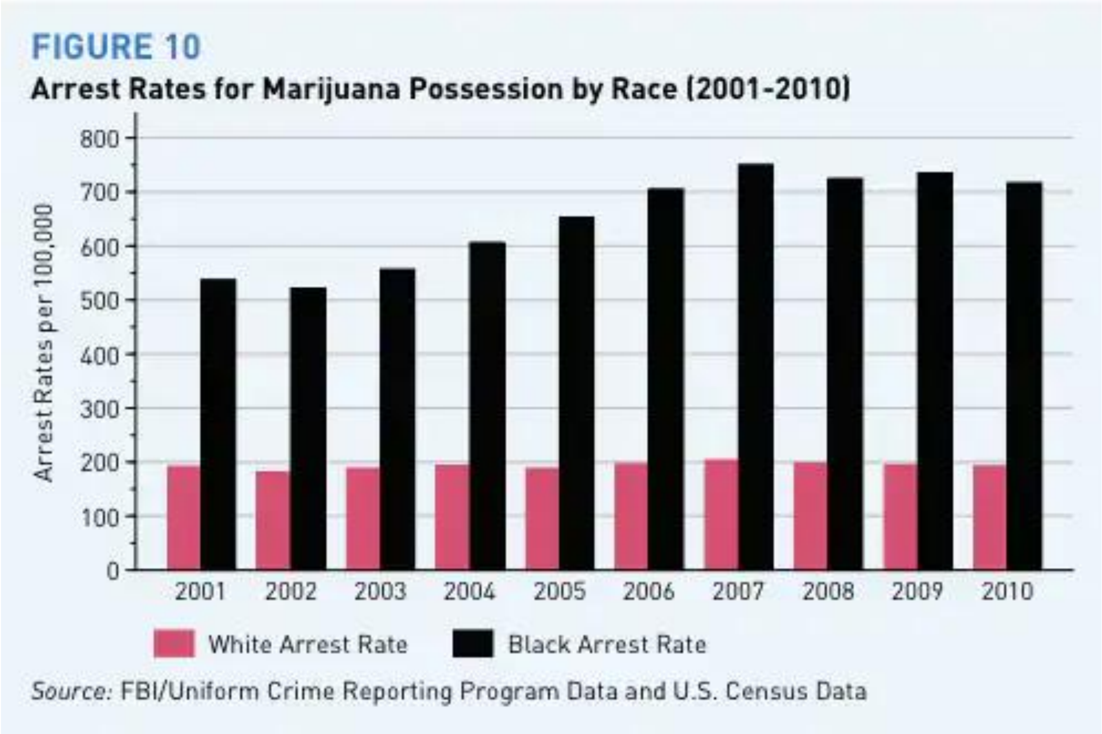


Algorithmic Fairness

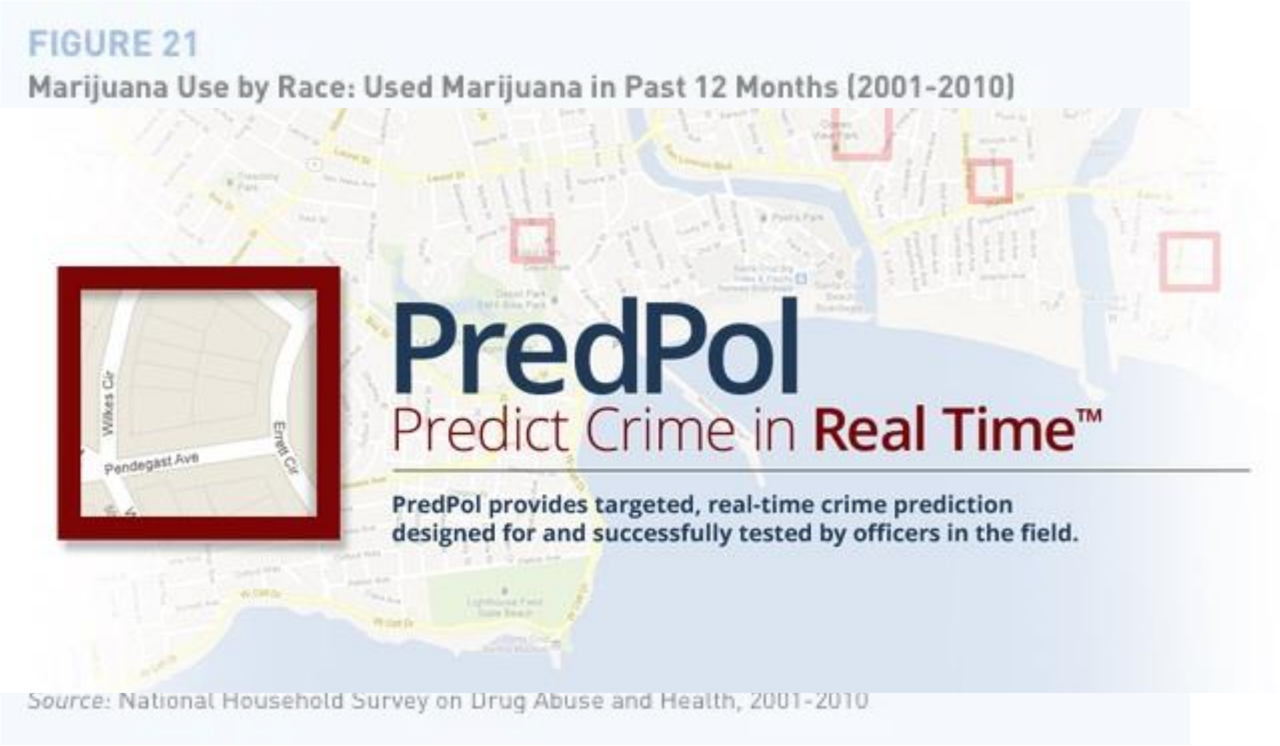


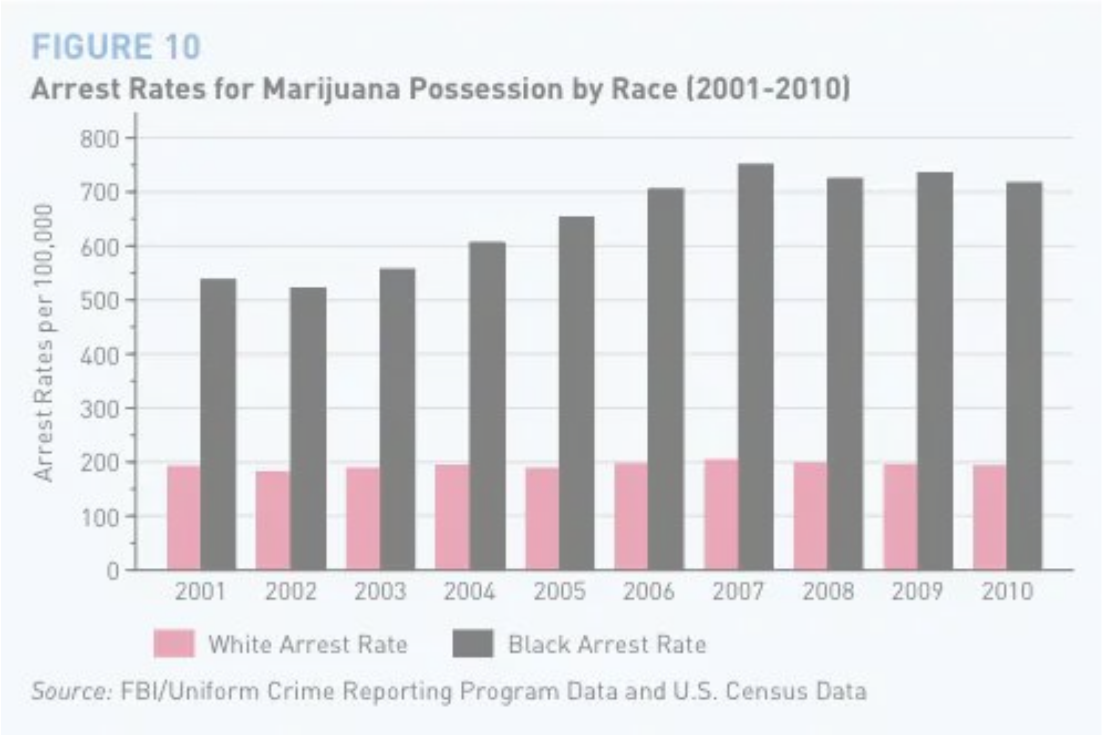
Algorithmic Fairness



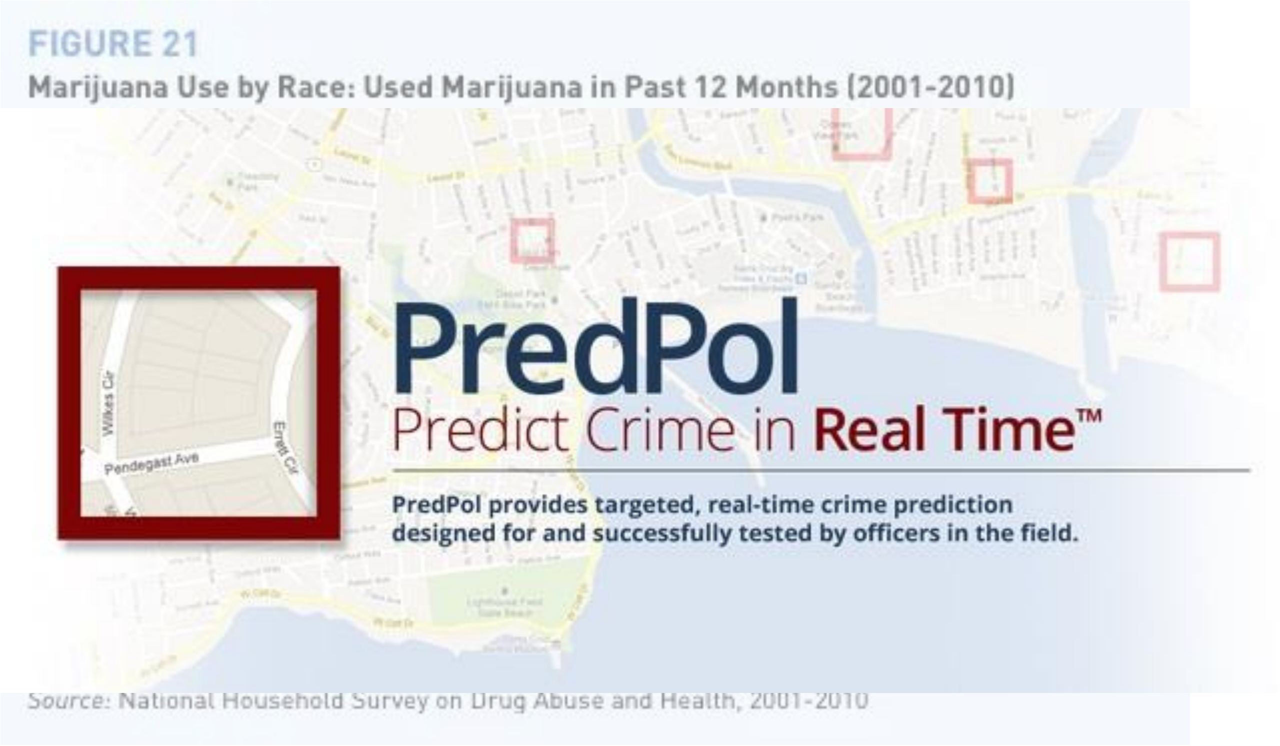


Algorithmic Fairness

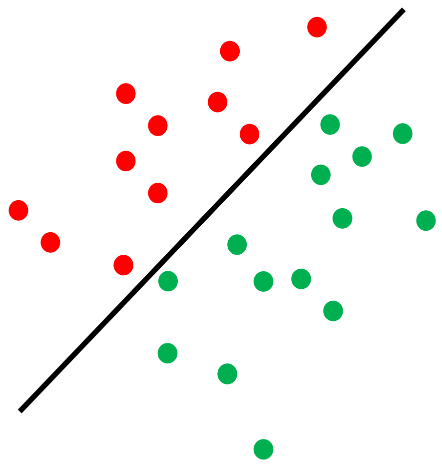




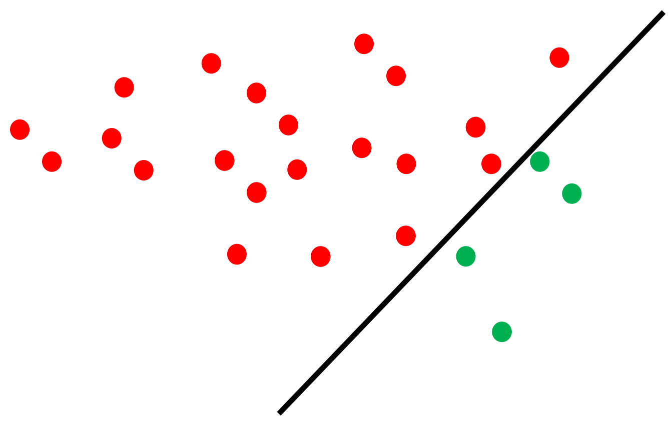
Algorithmic Fairness



Group Fairness



Group A



Group B

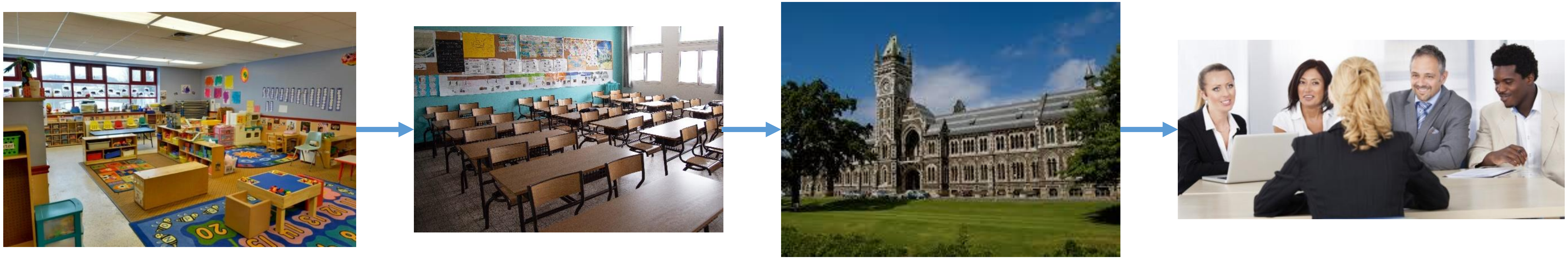
Group Fairness

How do I make this classifier fair?

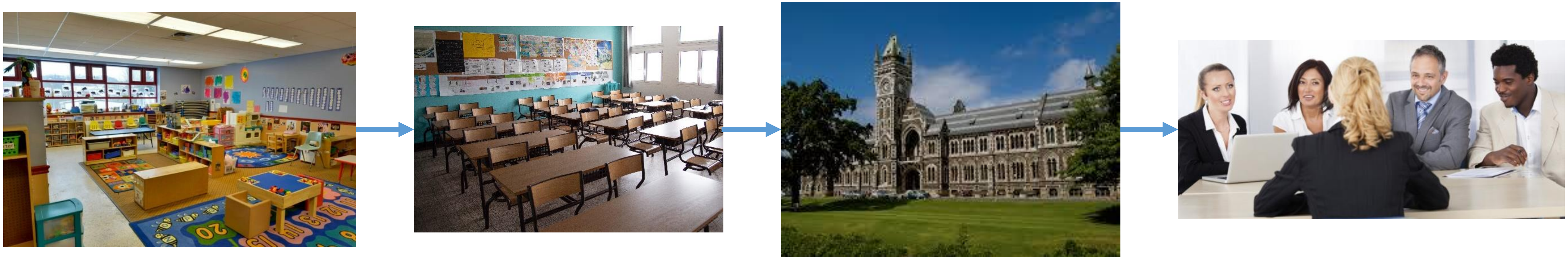
Group A

Group B

Decisions are made along pipelines...



… with disparities at each stage

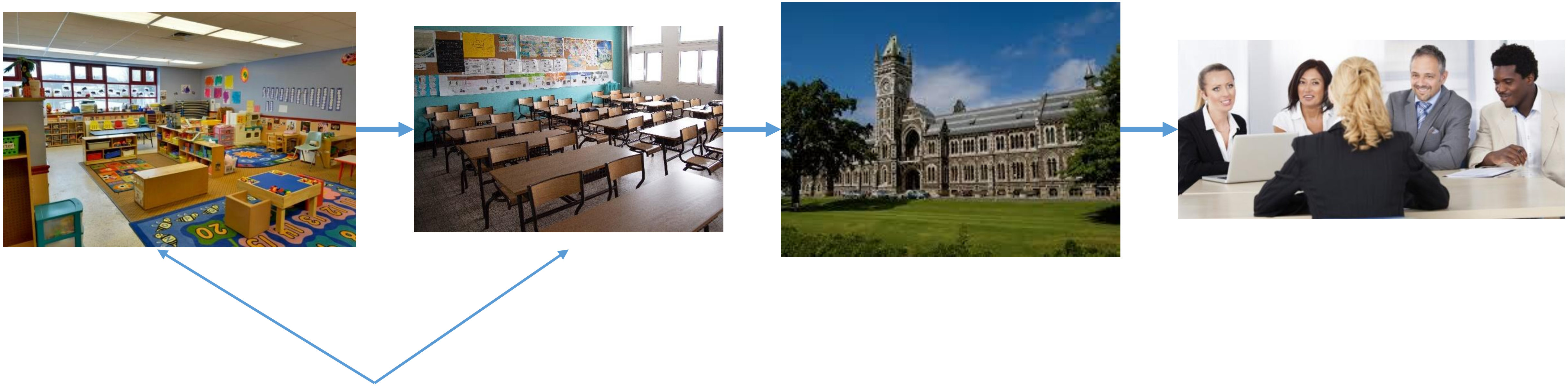


- Inequality of access to opportunities can arise at several stages of such pipelines
- Disparities compose: current opportunities are restricted by previous disparities/disparities have long-term effect on future opportunities
- Disparities can arise even at very early stages, for ex pre-school level

Where to intervene?

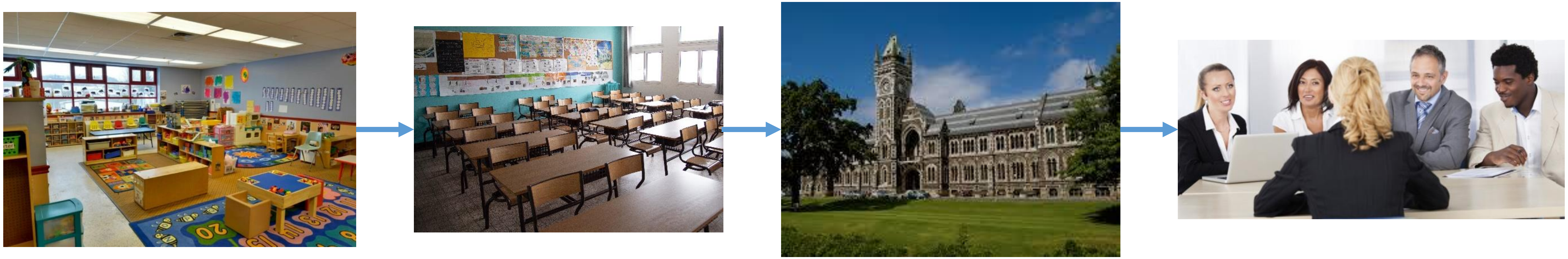
Maybe too late

Where to intervene?



Maybe worth
intervening here
and here too

Where to intervene?



May be valuable to intervene at several levels, rather than myopically/at a single one

Questions:

- **How do interventions at different stages compose?**
- **How this informs the optimal design of interventions at several levels of a pipeline that improve outcomes and reduce disparities across groups?**

If you are interested in composed decisions…

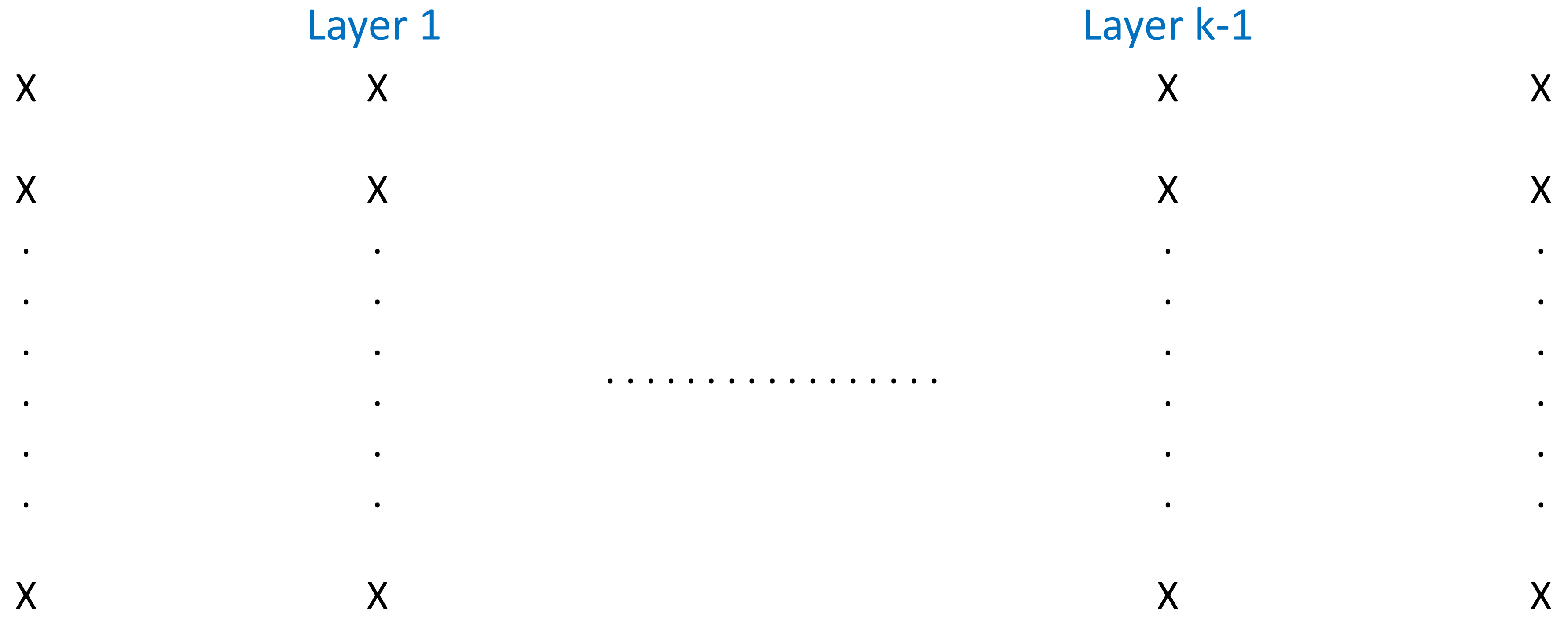
- Dwork and Ilvento: “Fairness under composition”
- Dwork, Ilvento, Jagadeesan: “Individual Fairness in Pipelines”
- Blum, Stangl, Vakilian: “Multi Stage Screening: Enforcing Fairness and Maximizing Efficiency in a Pre-Existing Pipeline”
- Etc.

Contribution 1: *stylized* pipeline intervention model on layered graphs

Starting layer

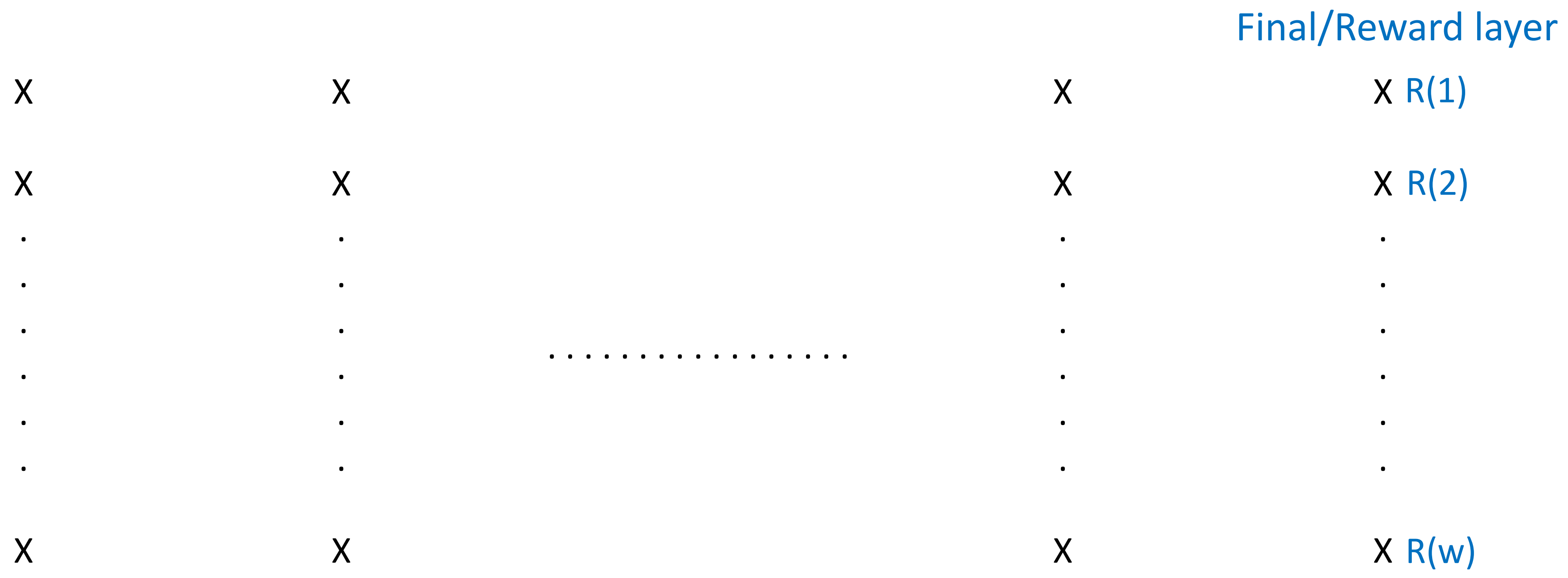
- Different starting nodes ⇔ different starting groups/sub-populations

A stylized pipeline intervention model



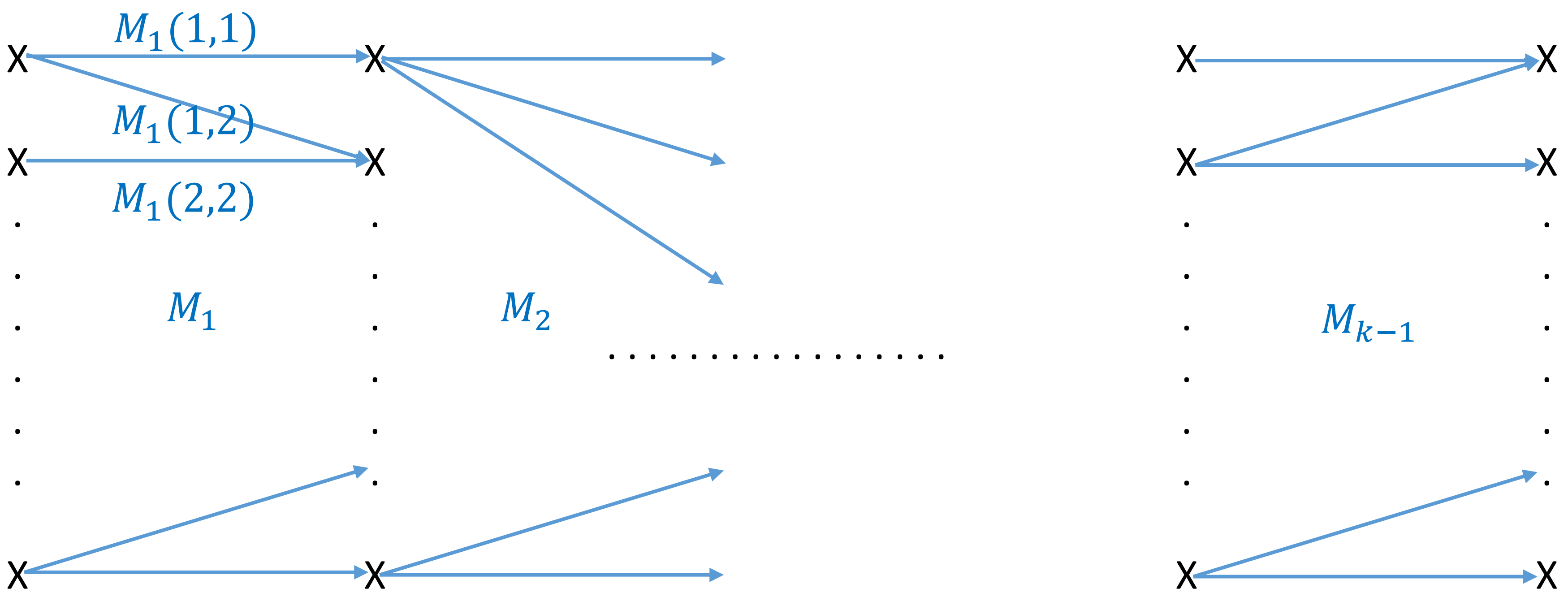
- Subsequent layers: each layer = stage of life, each node = outcome of a given stage
- For example, different educations, etc.

A stylized pipeline intervention model



R(i) = scalar measure of quality of outcome i

A stylized pipeline intervention model



- Stochastic transitions between layers. $M_t(i,j)$ = Pr[node i to node j|layer t -> t+1]
- Can model disparities in access to opportunities. Can give different groups different probabilistic paths to different reward nodes through the graph

A stylized pipeline intervention model

Intervention model:

- Centralized designer, can intervene at any/several stages
- Intervention = change stochastic transitions between layers

Under constraint:

- Incur cost to change transitions between 2 successive layers
- Maximum budget that can be invested across all layers/transitions

Cost function

- Cost from going from initial transition matrix M_t^0 to transition matrix M_t between layers t and t+1:

$$c(M_t^0, M_t)$$

- Main assumption:
  - Convexity in M_t (necessary for optimization)

- Budget constraint:

$$\sum_t c\left(M_t^0, M_t\right) \leq B$$

Contribution 2: DP for near-optimal interventions

Dynamic programming algorithms to find how to approximately optimally:

- Split the budget across different layers
- Use the budget between any two layers to change transitions

What do I mean by optimal here?

Goal #1: Max Social Welfare

Weighted (by population size) sum of the utilities
across the different starting sub-populations

Main caveat:

- Best that can be achieved at the level of the whole population…
- But this says nothing about each sub-population/group
- Potential issue: good outcomes for largest population, but ignore minority populations

Goal #2: Maximin Welfare

Maximize the welfare of an agent in the worst-off population

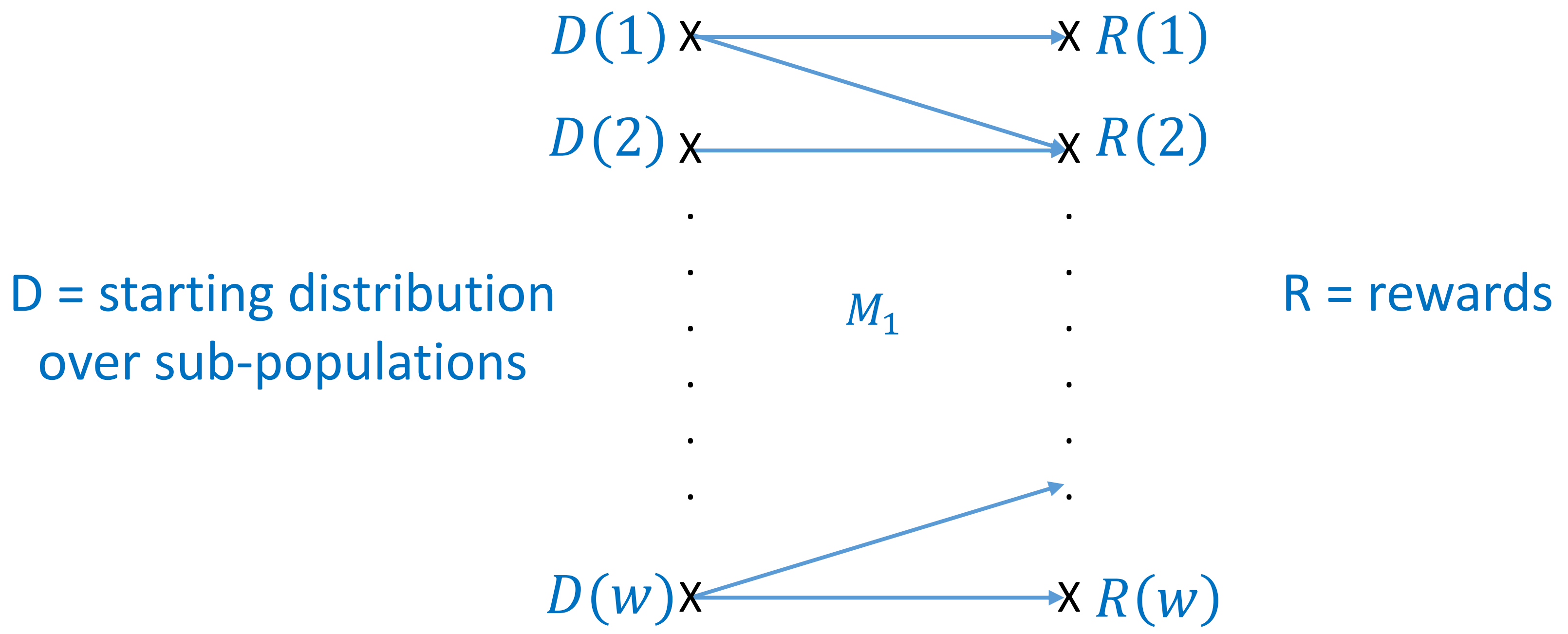
I.e., maximize

$$\min_i u_i(M_1, \ldots, M_{k-1})$$

(i = starting sub-population index)

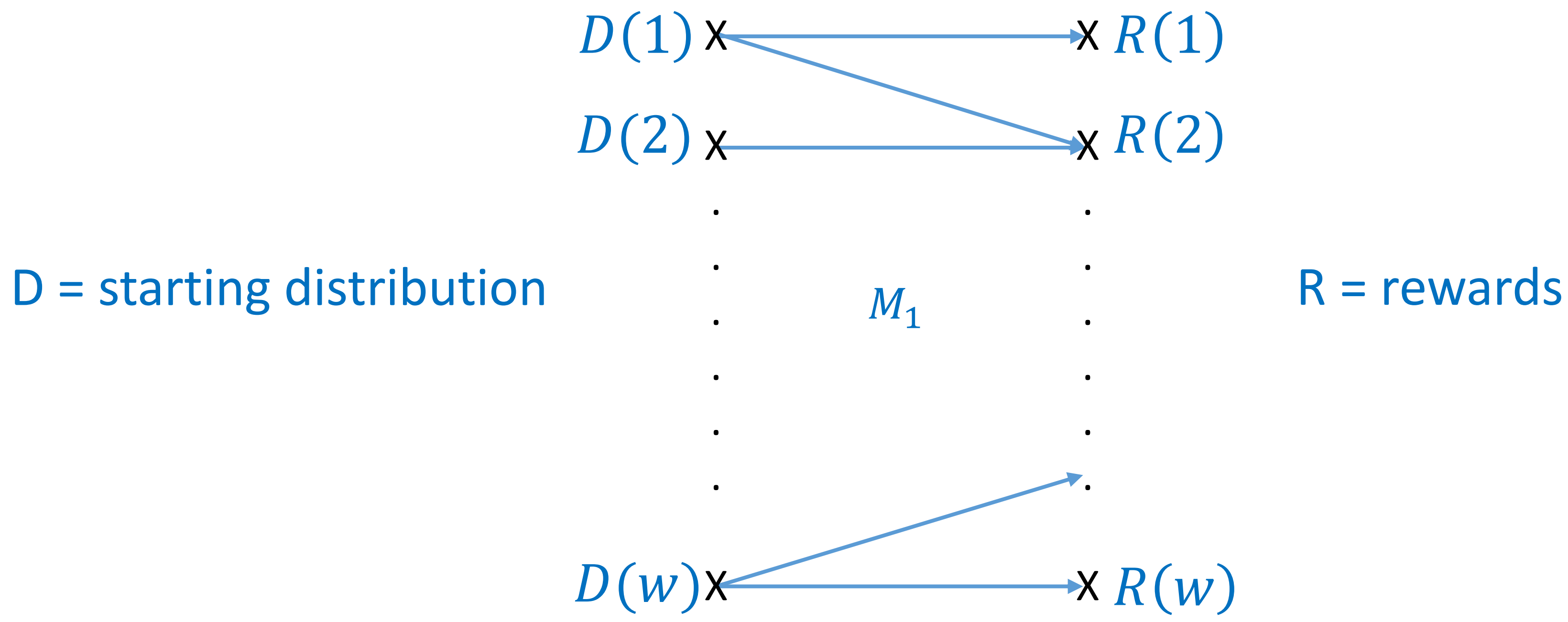
A Dynamic Programming approach for near-optimal SW

Easy case: only 2 layers, single transition matrix



A DP (get it?) approach for near-optimal SW

Easy case: only 2 layers, single transition matrix

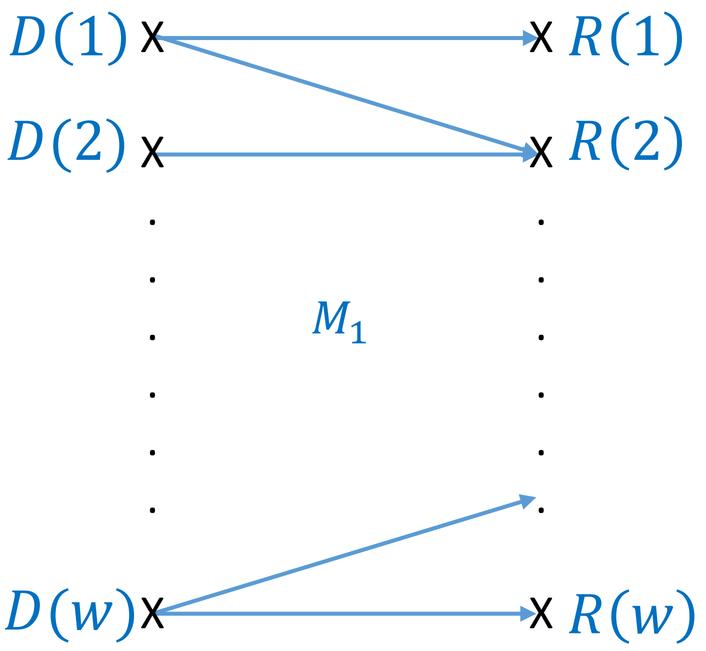


Then, $\max R\ M_1\ D$ such that $c\left(M_1^0, M_1\right) \leq B$.

A DP approach for near-optimal SW

Easy case: only 2 layers, single transition matrix

D = starting distribution



R = rewards

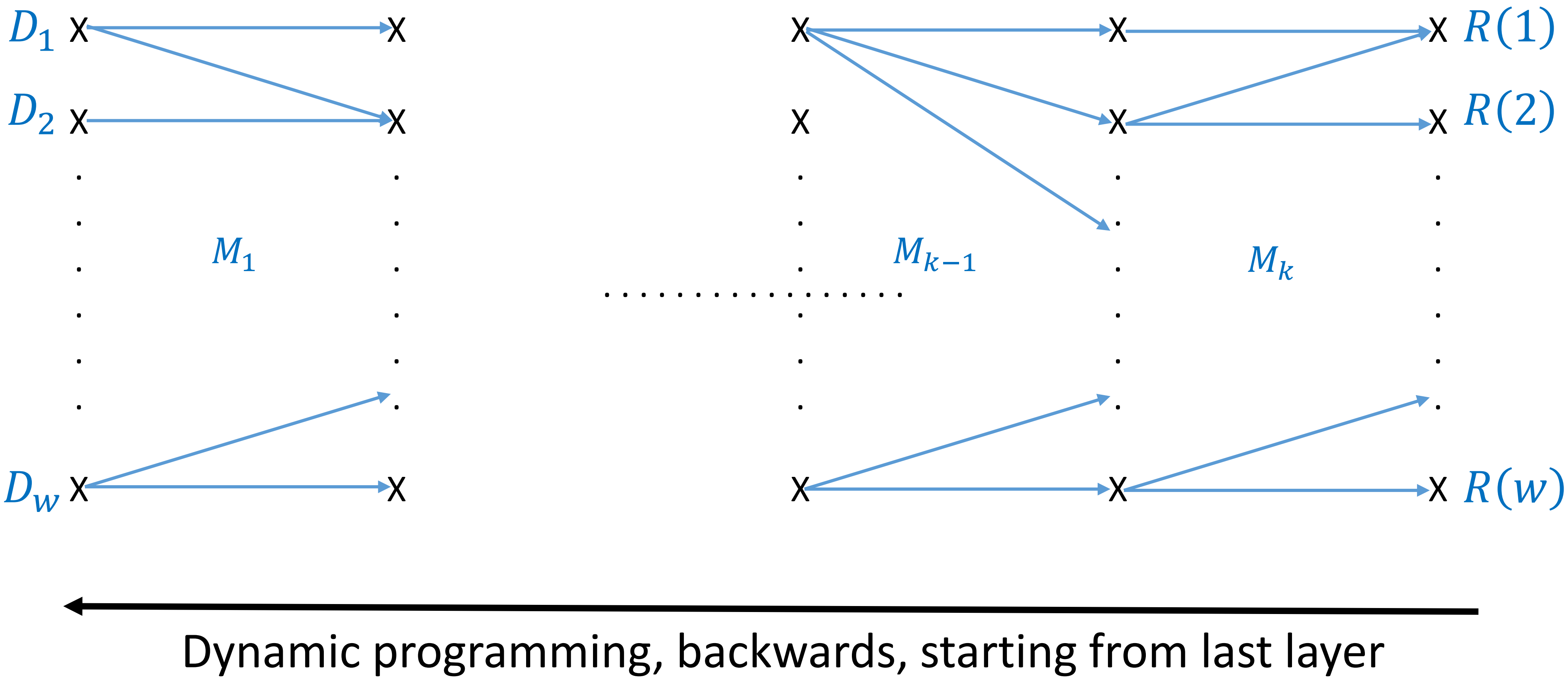
Then, $\max R\ M_1\ D$ such that $c\left(M_1^0, M_1\right) \leq B$.

Linear in M

Convex constraint → Convex optimization!

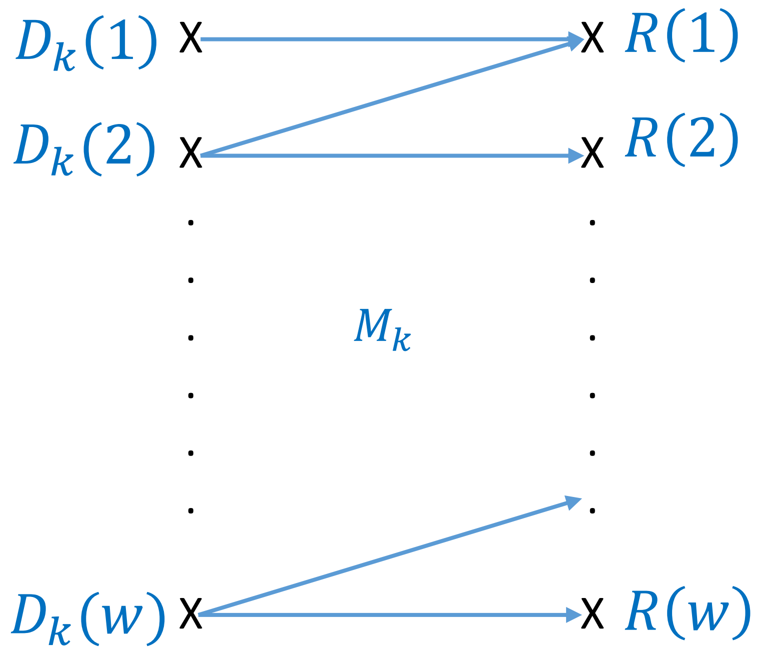
A DP approach for near-optimal SW

General case: many layers



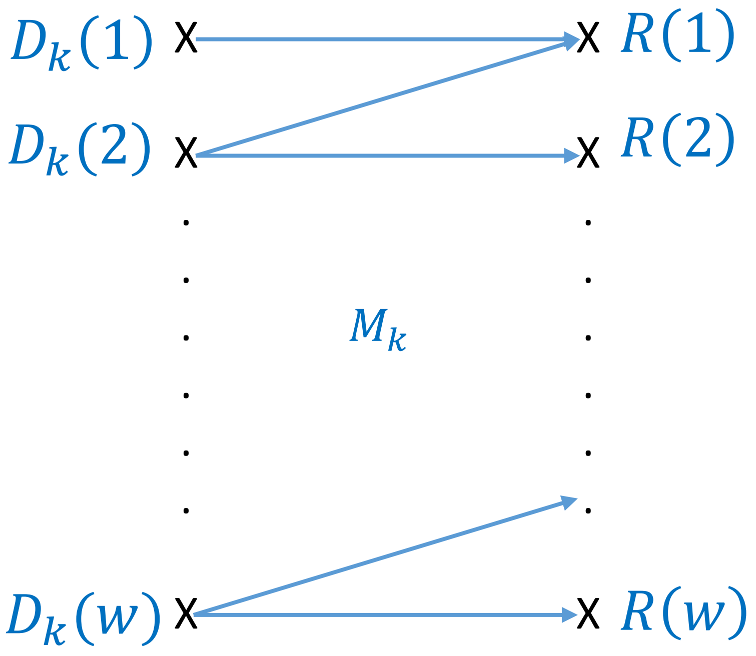
Start with final layer

- Start at the final transition matrix
- Solve $\max R\ M_1^t\ D_k$
  such that $c\left(M_1^0, M_1\right) \leq B_k$



Start with final layer

- Start at the final transition matrix
- Solve $\max R\ M_1^t\ D_k$ such that $c\left(M_1^0, M_1\right) \leq B_k$
- **Difficulty:** what is D_k here? *Depends on early transitions! Unknown: we solve from the end.*



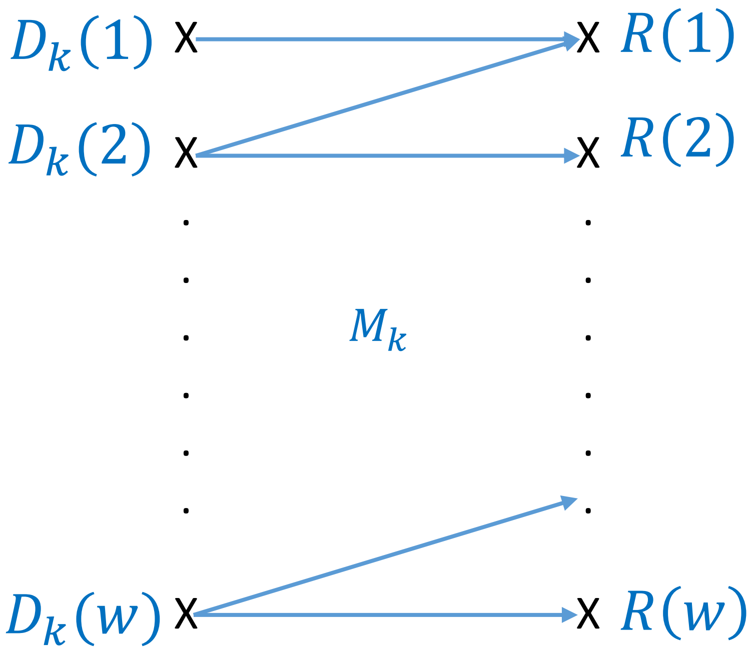
Discretizing D_k

Solution: guess D_k

- How? Try all possible D_k's on an ϵ-net
- Size of net $\sim \left(\frac{1}{\epsilon}\right)^w$

➔ Can only deal with constant w

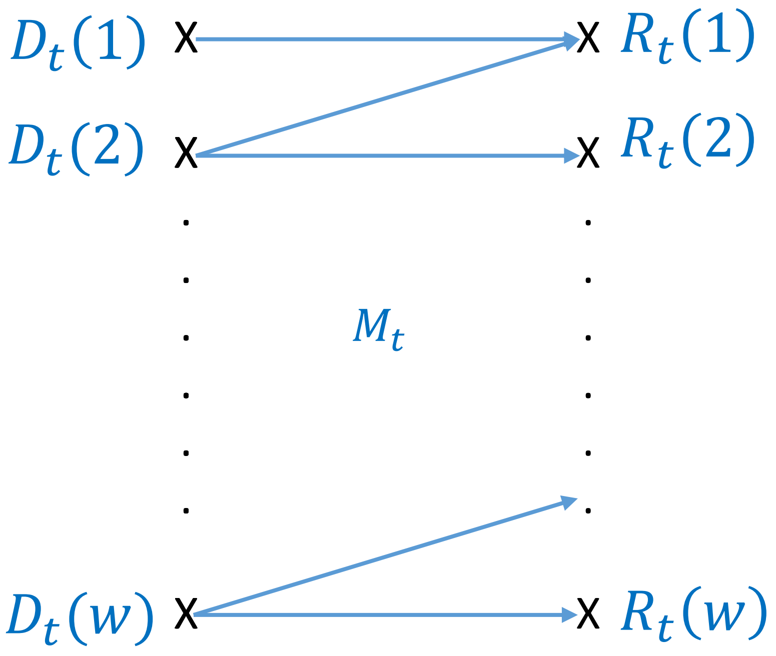
- For each D_k on the net, solve program



A DP approach to finding near-optimal SW

How to iterate on previous layers t -> t+1

- Same idea, solve program for all D_t's on a net

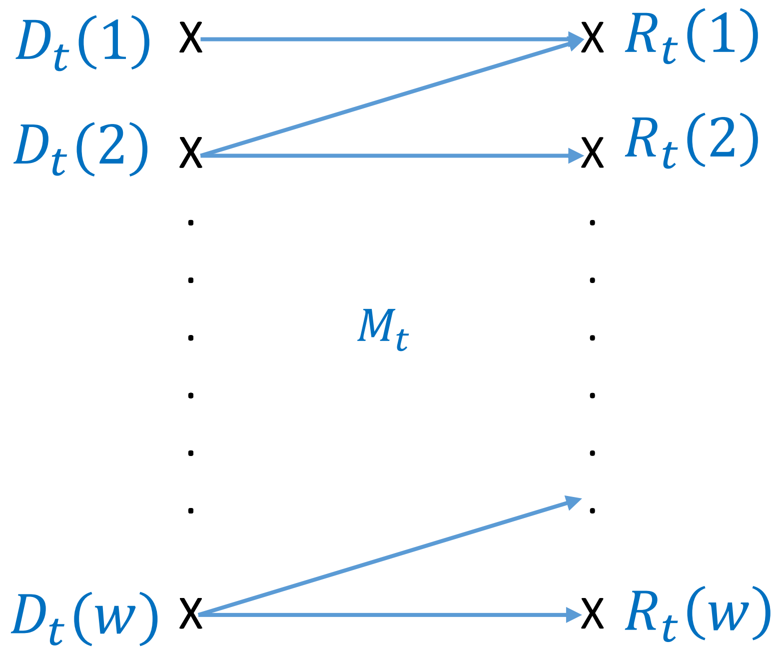


A DP approach to finding near-optimal SW

How to iterate on previous layers t -> t+1

- Same idea, solve program for all D_t's on a net
- How to deal with R_t?

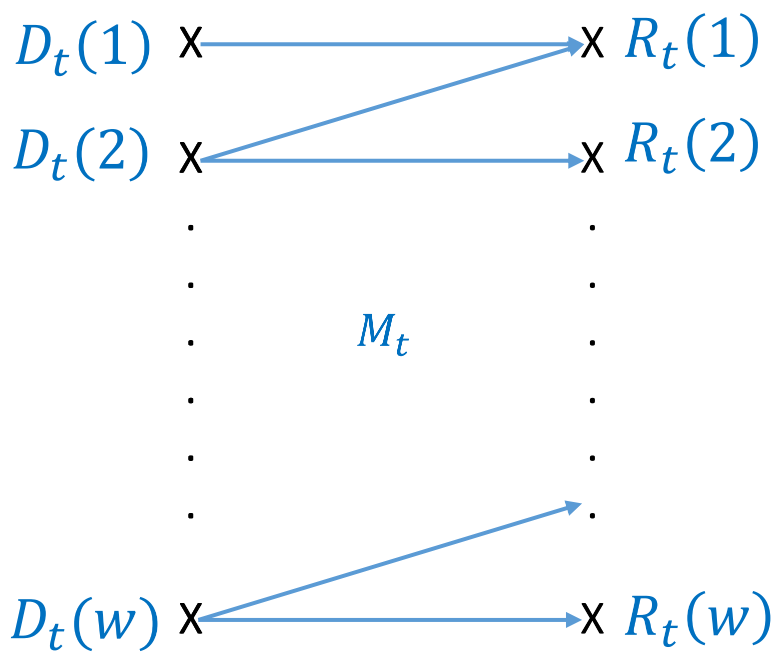
Use solutions of the previous step
Each solution defines a reward vector for t-> t+1



A DP approach to finding near-optimal SW

A quick note on budget:

- Note that we use B_t at each step t. But, OPT budget split across layers is unknown
- Idea: same approach as for D:
  - 1D grid for the budget
  - Try all budget possibilities on each transition



Guarantees of our algorithm

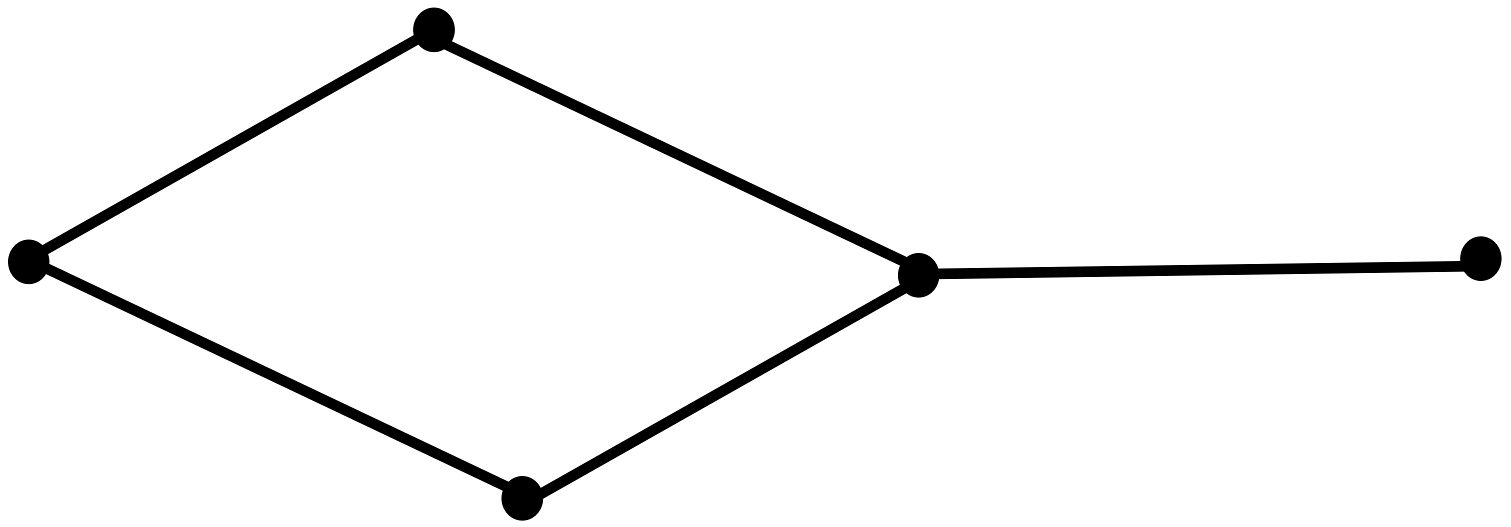
- Welfare guarantee:
  - Net makes us lose $O(\epsilon)$ at each step
  - Get a $\mathrm{k}\epsilon$ approx. to social welfare if k transitions
- Computational efficiency:
  - Each step requires looking at poly($(1/\epsilon)^w$) possibilities due to discretization.
  - Need w constant (think coarse grouping of outcomes in each stage)
  - But need to do this only k times.
- Maximin objective:
  - Instead of keeping track of all possible D_t's at the start of layer t, keep track of more fine-grained $D_{t,i}$ for each starting node i
  - Then, use the same approach

Hardness: Super-polynomial dependencies on width are unavoidable

- Can be seen via reduction to vertex cover

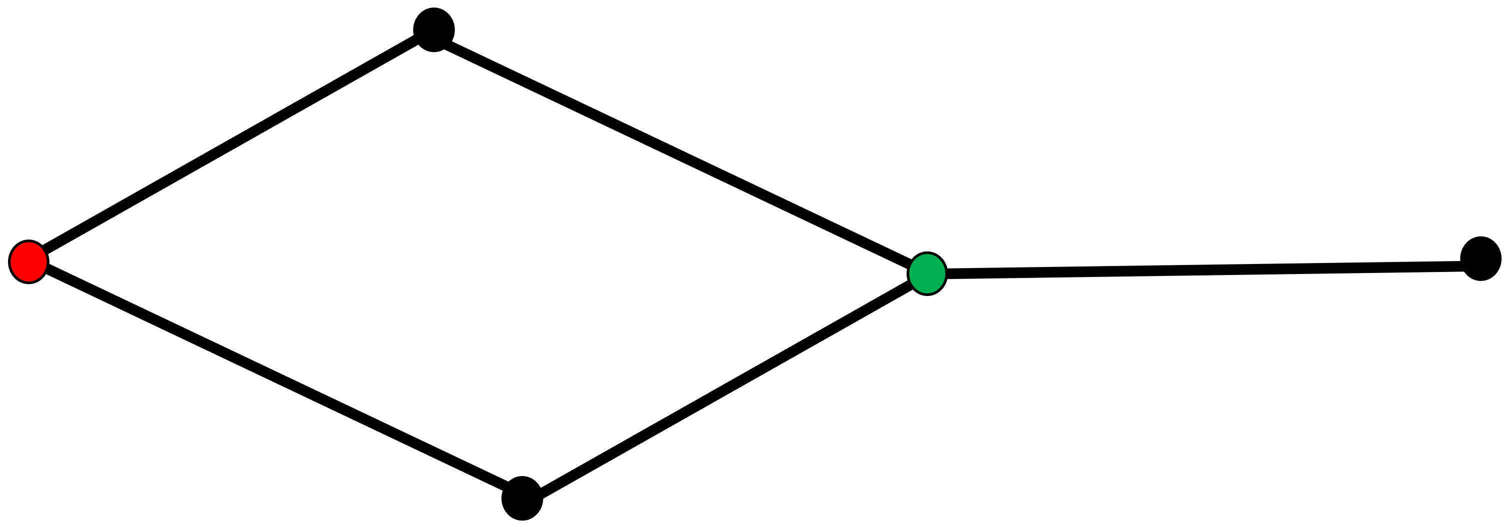
Hardness: Super-polynomial dependencies on width are unavoidable

- Can be seen via reduction to vertex cover



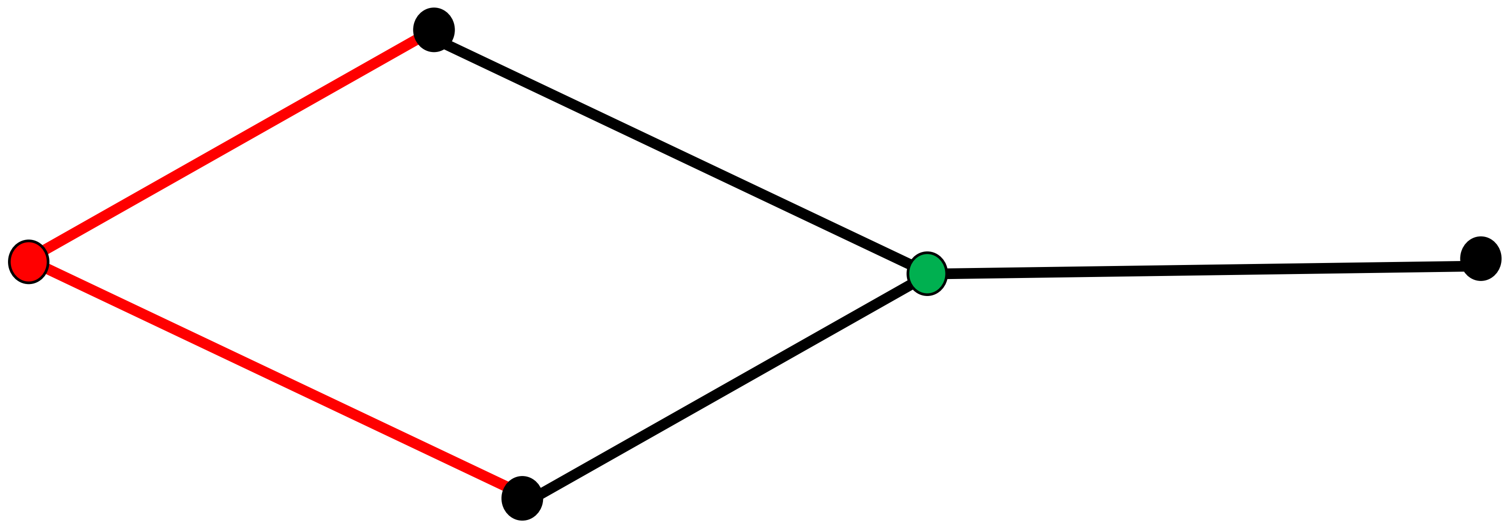
Hardness: Super-polynomial dependencies on width are unavoidable

- Can be seen via reduction to vertex cover



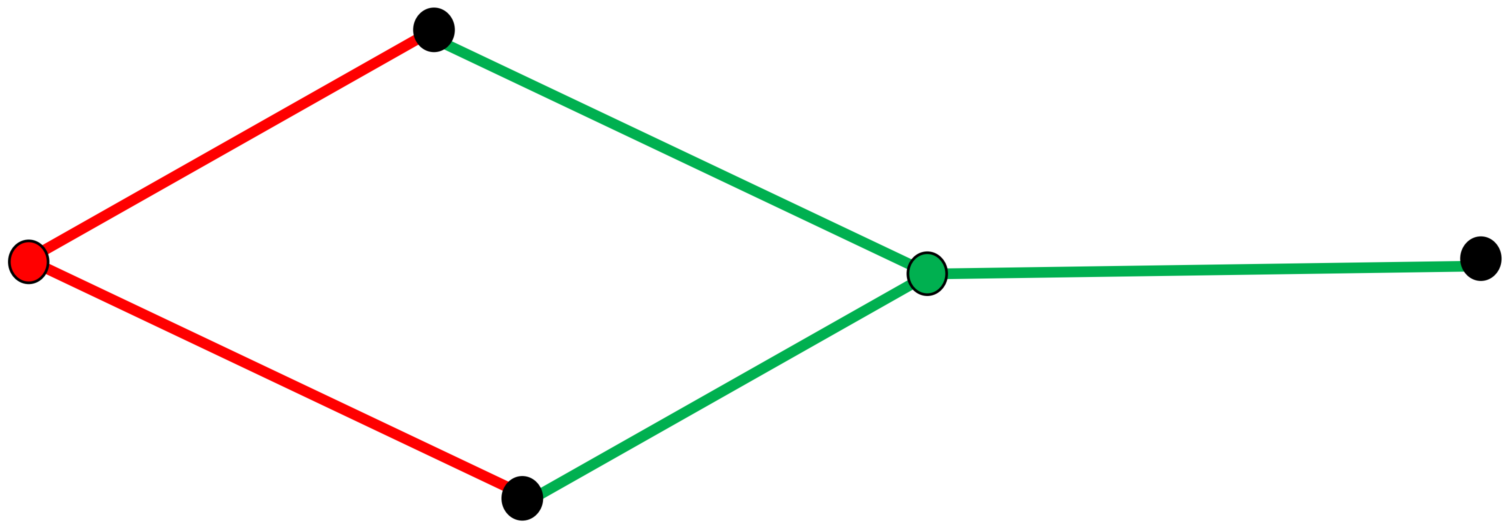
Hardness: Super-polynomial dependencies on width are unavoidable

- Can be seen via reduction to vertex cover



Hardness: Super-polynomial dependencies on width are unavoidable

- Can be seen via reduction to vertex cover



Hardness: Super-polynomial dependencies on width are unavoidable

- Can be seen via reduction to vertex cover

- Why vertex cover again?

Hardness: Super-polynomial dependencies on width are unavoidable

- Can be seen via reduction to vertex cover

- Why vertex cover again?
  - We'll see the reduction in a second…

Hardness: Super-polynomial dependencies on width are unavoidable

- Can be seen via reduction to vertex cover

- Why vertex cover again?
  - We’ll see the reduction in a second…
  - But strong hardness results.

Hardness: Super-polynomial dependencies on width are unavoidable

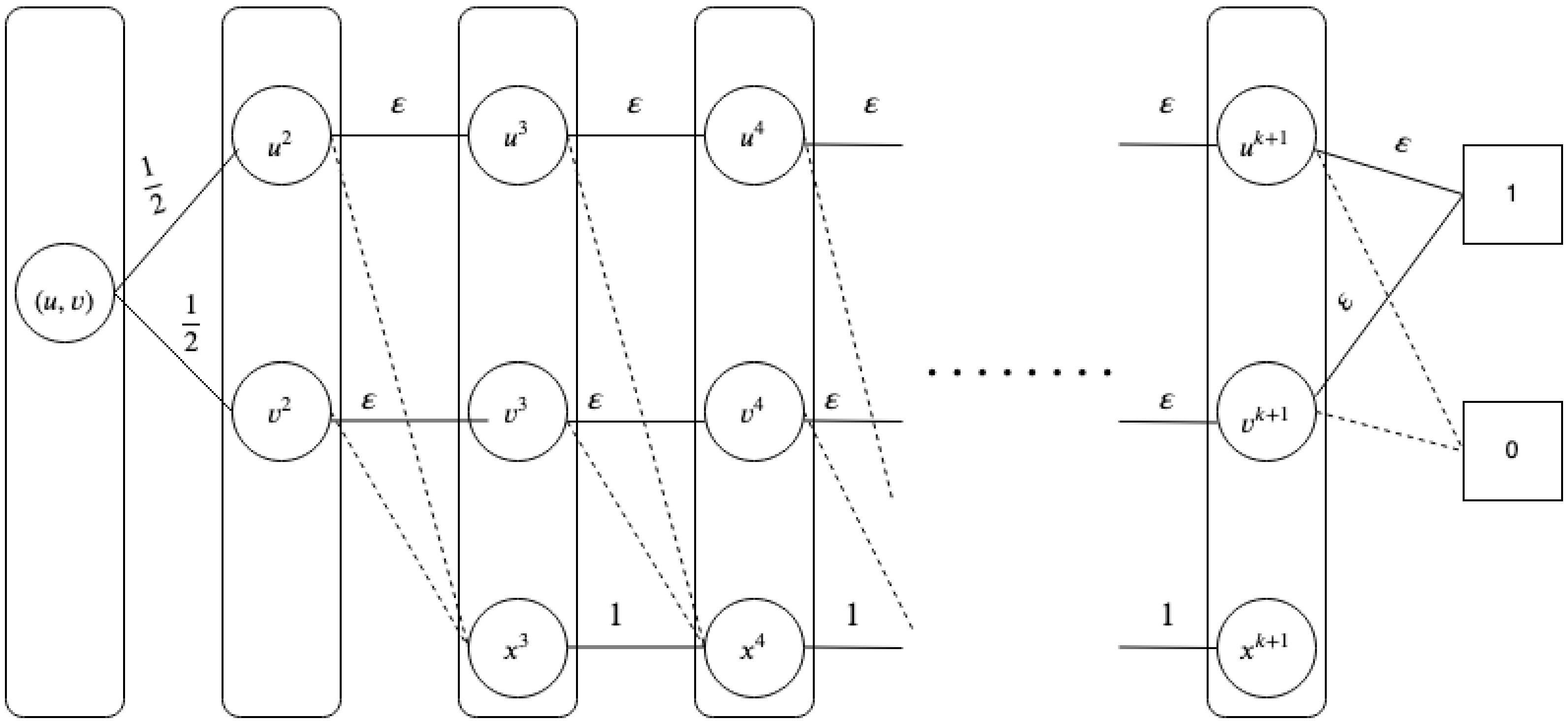
- Can be seen via reduction to vertex cover
- Why vertex cover again?
  - We'll see the reduction in a second…
  - But strong hardness results.
  - Not just NP-complete…

Hardness: Super-polynomial dependencies on width are unavoidable

- Can be seen via reduction to vertex cover
- Why vertex cover again?
  - We'll see the reduction in a second…
  - But strong hardness results.
  - Not just NP-complete…
  - … but also cannot be approximated to a constant factor < 1.3606 [Dinur – Safra 2005]

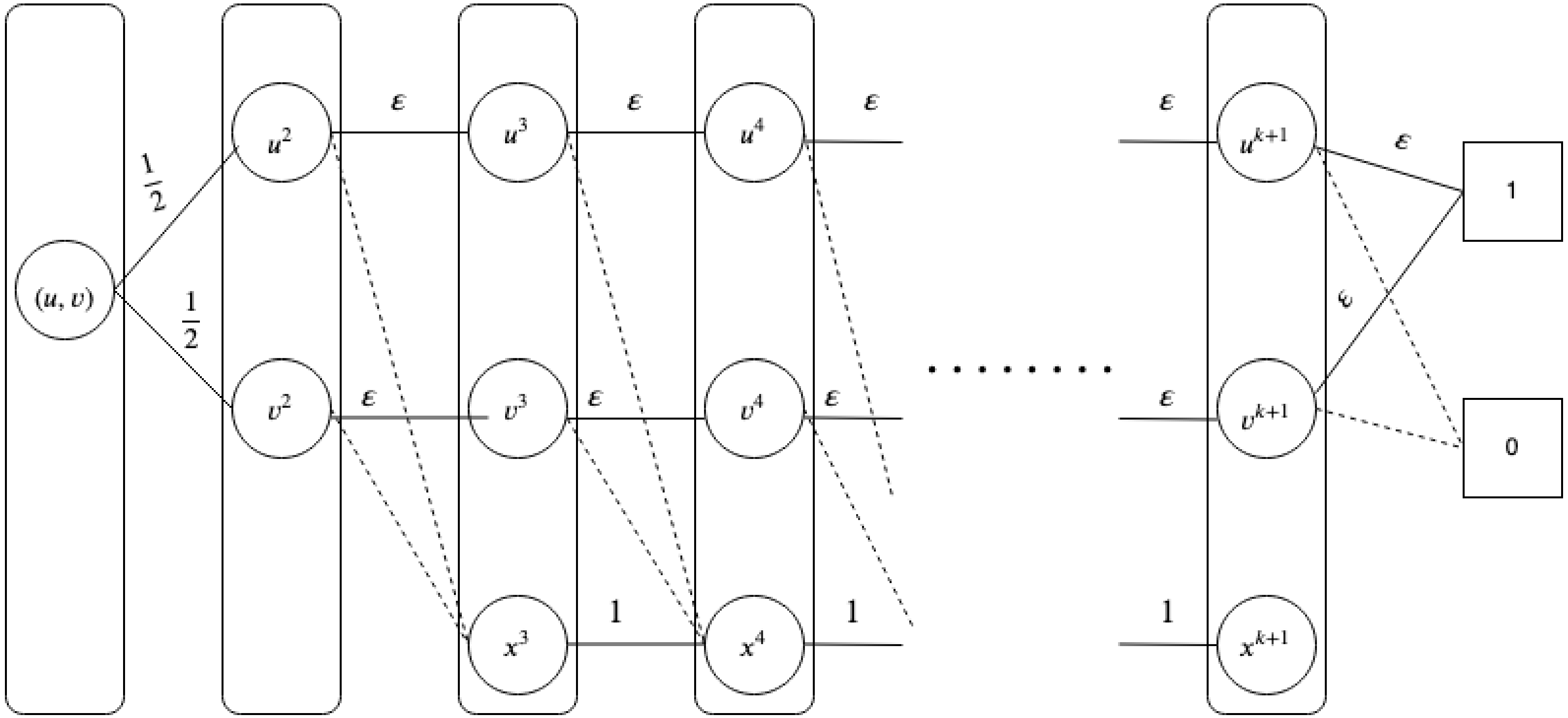
Hardness: Super-polynomial dependencies on width are unavoidable

Take graph G on which we want to solve vertex cover. For each edge (u,v) *in the vertex cover graph*, build:



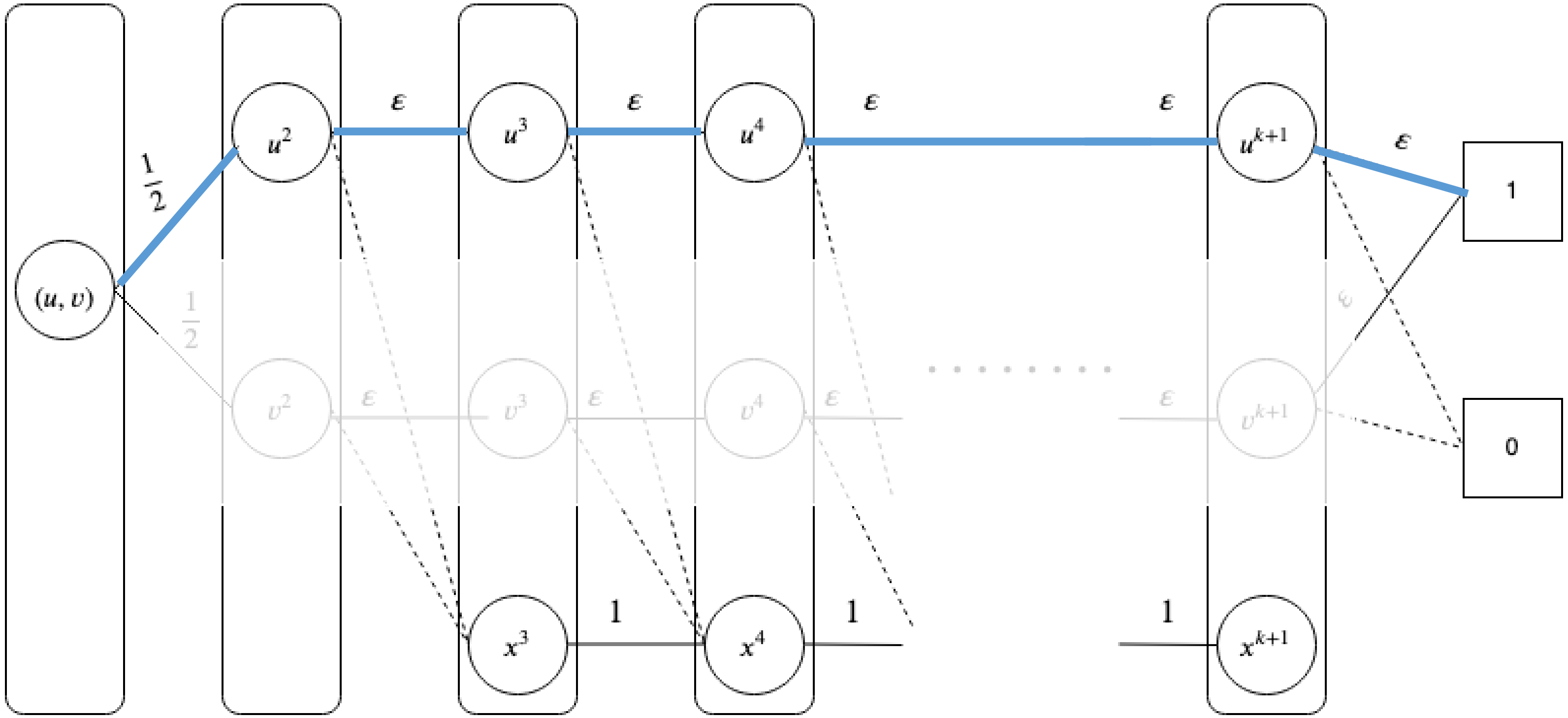
Hardness: Super-polynomial dependencies on width are unavoidable

Idea: most efficient way to get minimax ➔ pick path going to 1 for *each* (u,v) to get welfare, but use as few paths as possible



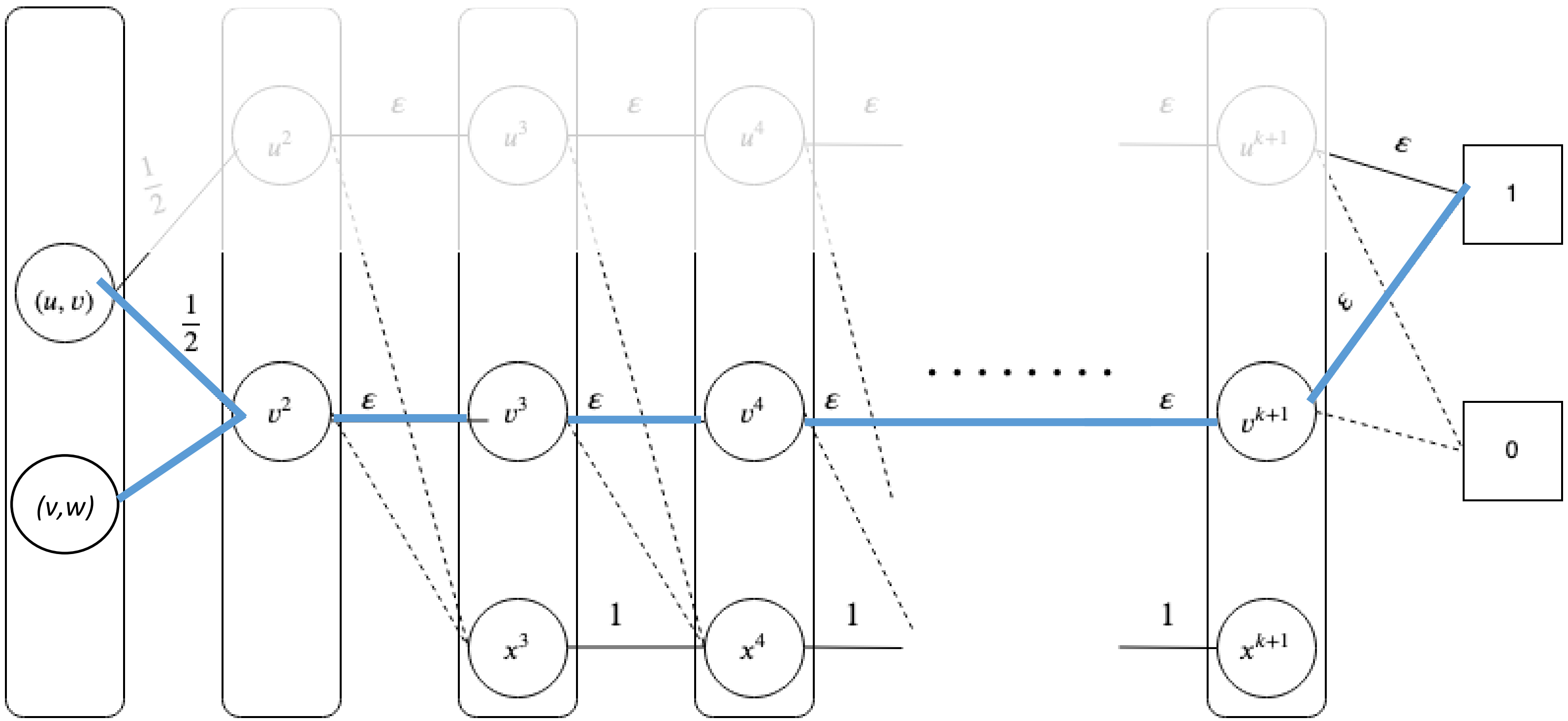
Hardness: Super-polynomial dependencies on width are unavoidable

Idea: most efficient way to get minimax ➔ pick path going to 1 for *each* (u,v) to get welfare, but use as few paths as possible



Hardness: Super-polynomial dependencies on width are unavoidable

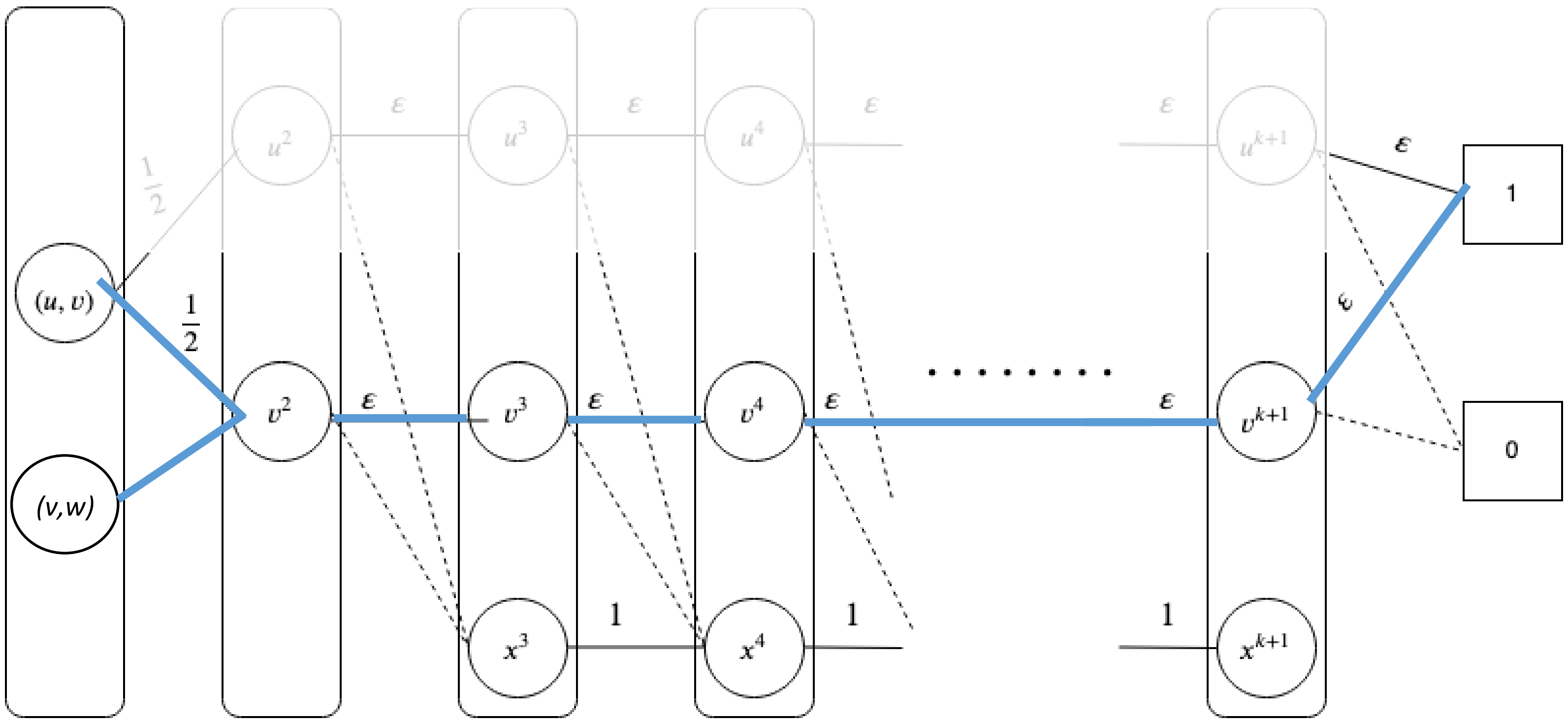
Idea: most efficient way to get minimax ➔ pick path going to 1 for *each* (u,v) to get welfare, but use as few paths as possible



Hardness: Super-polynomial dependencies on width are unavoidable

Idea: most efficient way to get minimax ➔ pick path going to 1 for *each* (u,v) to get welfare, but use as few paths as possible

- But, picking a path = picking a vertex in og graph
- Using as few paths as possible ⇔ using as few vertices as possible in og graph



Contribution 3: Price of fairness

$$P_f = \frac{OPT\ SW}{SW\ of\ maximin\ sol}$$

- Simple case: linear cost 1 for changing transition by 1
- Result: *tight* bounds
  - $P_f = w$ for very very small B
  - $P_f = w/B$ for intermediate B
  - $P_f = 1$ for large B

Contribution 3: Price of fairness

$$P_f = \frac{OPT\ SW}{SW\ of\ maximin\ sol}$$

- Simple case: linear cost 1 for changing transition by 1
- Result: *tight* bounds
  - ~~$P_f = w$ for small B~~ (corner case)
  - $P_f = w/B$ for intermediate B
  - $P_f = 1$ for large B

Contribution 3: Price of fairness

$$P_f = \frac{OPT\ SW}{SW\ of\ maximin\ sol}$$

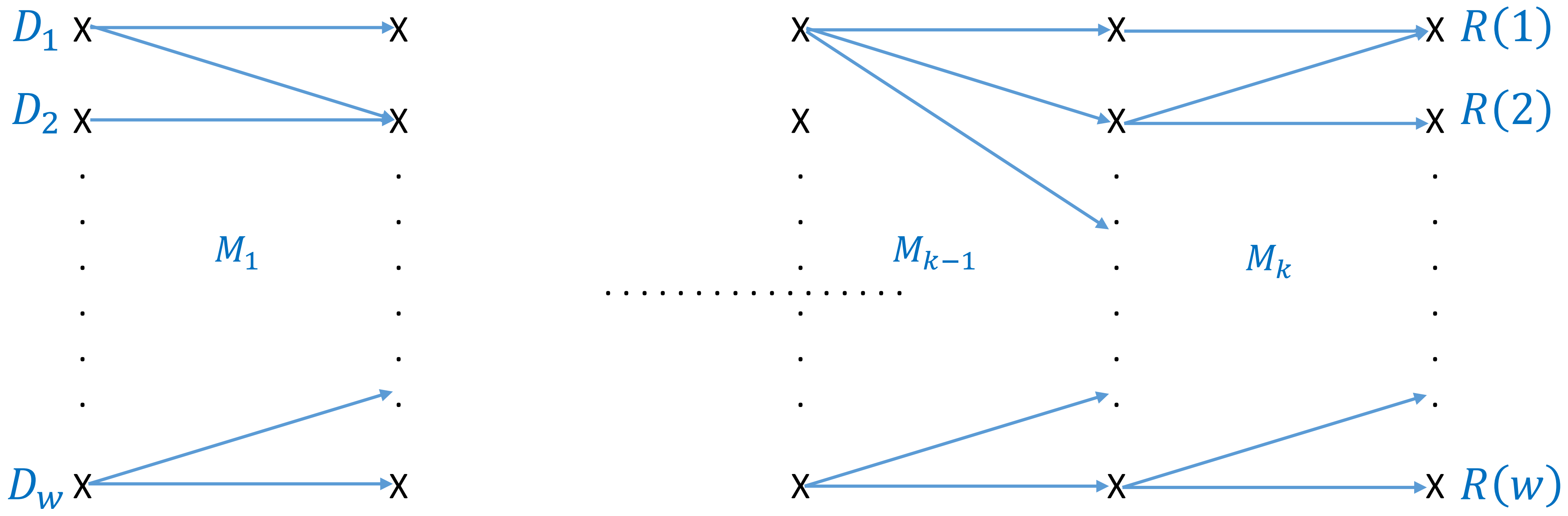
- Simple case: linear cost 1 for changing transition by 1
- Result: *tight* bounds
  - ~~$P_f = w$ for small B~~
  - $P_f = w/B$ for intermediate B
  - ~~$P_f = 1$ for large B~~ “trivial – the proof is left to the reader as an exercise”

Contribution 3: Price of fairness

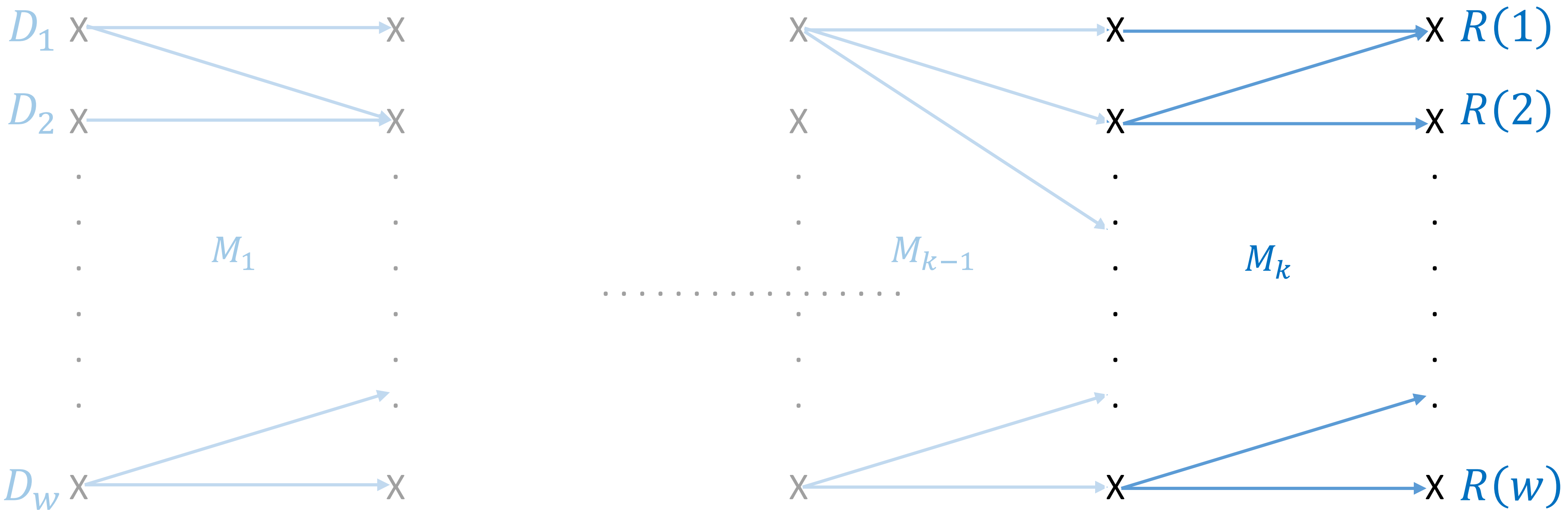
$$P_f = \frac{OPT\ SW}{SW\ of\ maximin\ sol}$$

- Simple case: linear cost 1 for changing transition by 1
- Result: *tight* bounds
  - ~~$P_f = w$ for small B~~
  - **$P_f = w/B$ for intermediate B**
  - ~~$P_f = 1$ for large B~~

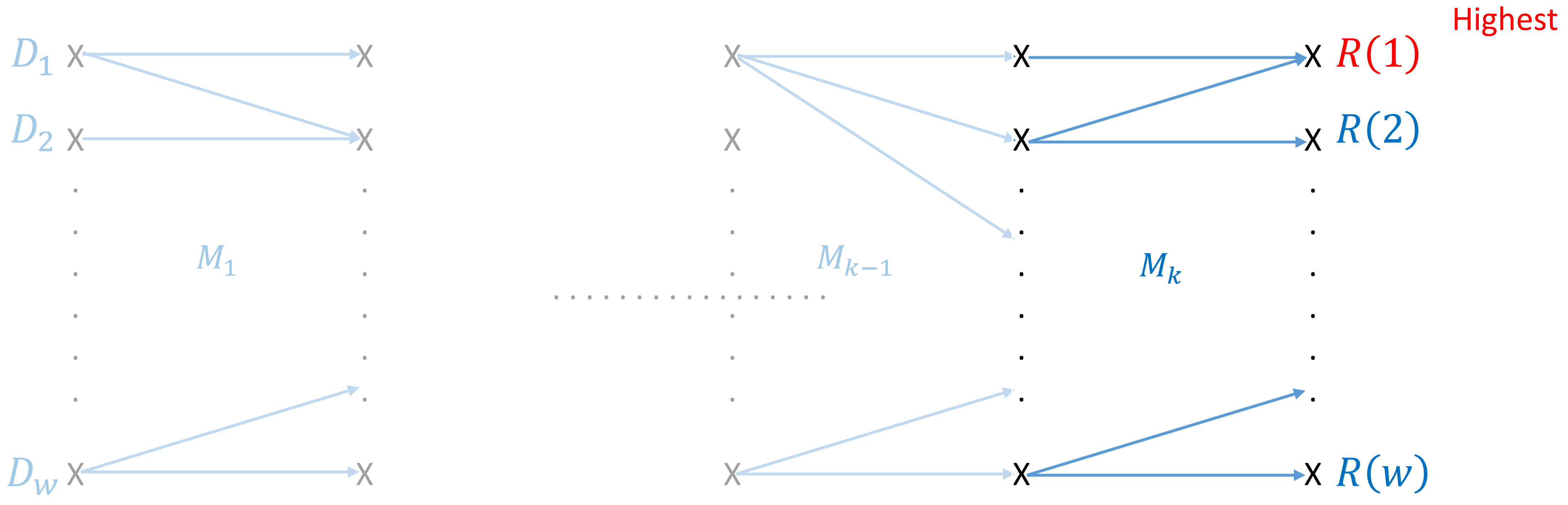
Price of fairness: some intuition



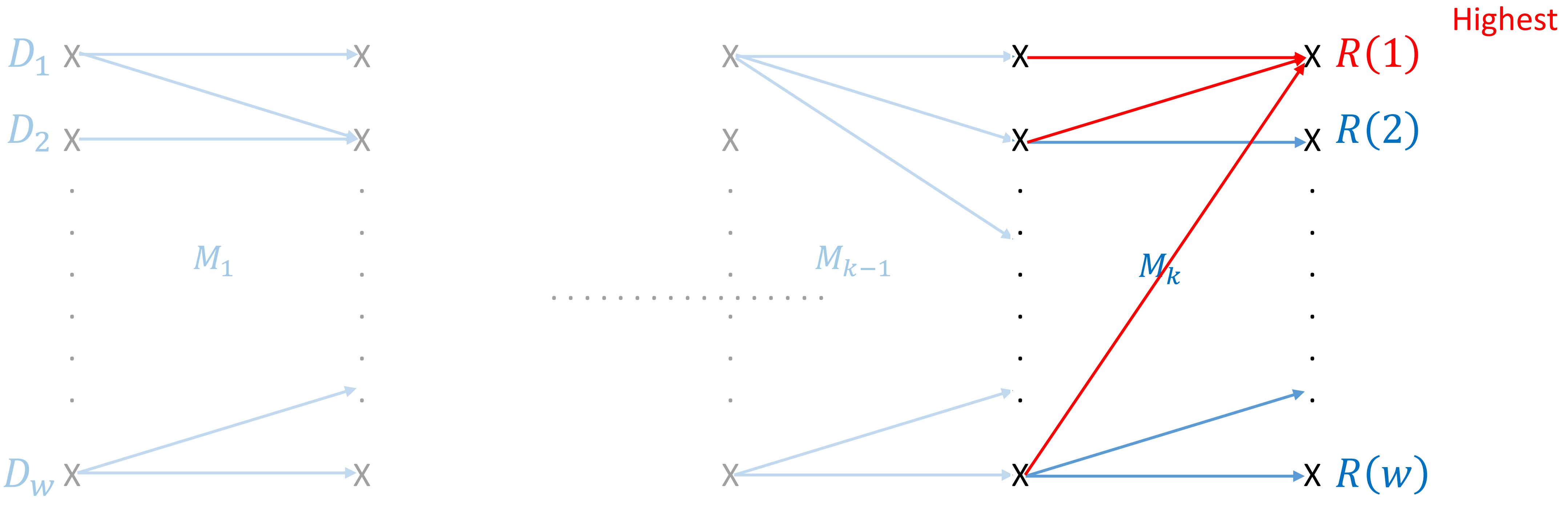
Price of fairness: some intuition



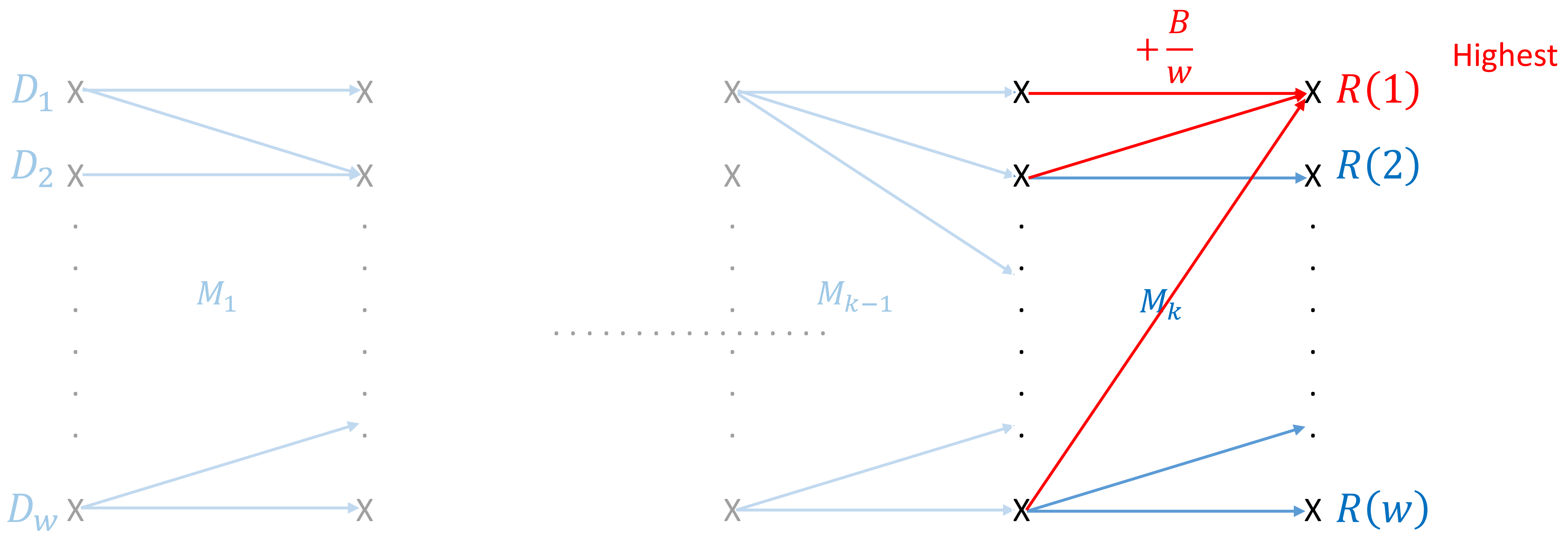
Price of fairness: some intuition



Price of fairness: some intuition

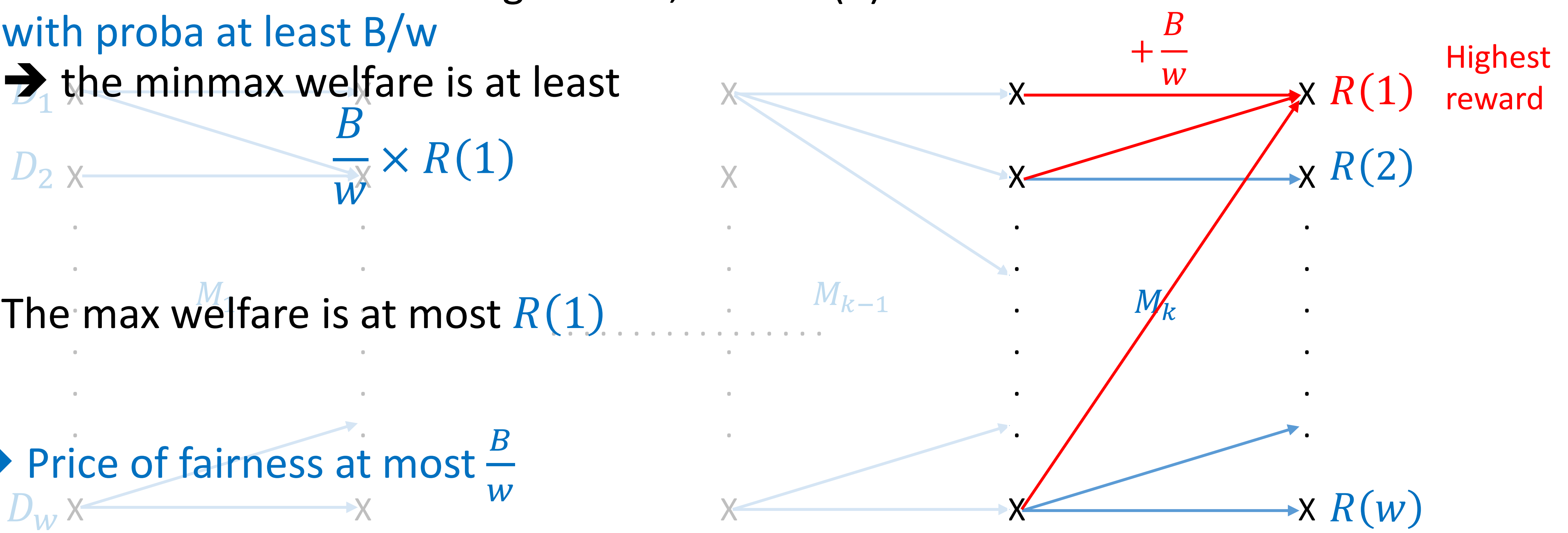


Price of fairness: some intuition



Price of fairness: some intuition

- No matter what the starting node is, reach R(1) with proba at least B/w
  ➔ the minmax welfare is at least $\frac{B}{w} \times R(1)$



- The max welfare is at most $R(1)$

➔ Price of fairness at most $\frac{B}{w}$

Remarks and future directions

Still a first step/stylized model; in practice, important future directions:

- Different populations may face different transitions even if on the same node in the graph

Population-specific transitions

Solution:

- Just duplicate nodes. For each outcome of a layer, there is a corresponding (outcome, starting population) node
- Can correlate effect of interventions across same outcome, different starting populations through cost function.
- E.g., if modify transition for starting population 1, can modify transition for pop 2 by some amount for free.

How does this affect the graph and algorithms?

- Quadratic blow-up w.r.t width
- $w \rightarrow w^2$

Remarks and future directions

Still a first step/stylized model; in practice, important future directions:

- ~~Different populations may face different transitions even if on the same node in the graph~~
- Transitions may not be stochastic, but involve strategic elements; agents make choices
- Acyclic model, does not take feedback loops into account
- Simplified/1D reward model + everyone wants the same outcomes
- What happens if non-centralized designer/different entities intervene at different stages?
- What if we try to estimate transitions/effect of interventions from real data?
- Etc.

Pipeline Interventions

Eshwar Ram Arunachaleswaran*
Sampath Kannan*
Aaron Roth*
Juba Ziani#

**University of Pennsylvania*
#Georgia Tech