INTRODUCTION TO DATA SCIENCE

JOHN P DICKERSON

Lecture #29 – 12/10/2018

CMSC320
Mondays & Wednesdays
2:00pm – 3:15pm

COMPUTER SCIENCE
UNIVERSITY OF MARYLAND

ANNOUNCEMENTS

Please fill out course evaluations!

- https://courseevalum.umd.edu

Last day to fill them out is tomorrow.

- Start Date 2018-11-29 00:00
- End Date 2018-12-11 23:59

Response Rate

	Responded	Invited	% Rate
Students	58	223	26.01%

GRADES & FINAL TUTORIALS

Project 4 is being graded as we speak, should be up in the next day or two

- Generally, people did very well!

Remember to:

- Submit via ELMS the URL for the GitHub Page for your group's final tutorial
- Fill in on the Google Doc your project title, URL, group members, and data source(s)

INCREASING ACCESS TO ORGANS THROUGH MARKET DESIGN & OPTIMIZATION

JOHN P DICKERSON

COMPUTER SCIENCE
UNIVERSITY OF MARYLAND

University of Maryland
CMSC320 – Last Lecture
December 10, 2018

Markets come in many forms …

… some of which don't conform to conventional notions of markets …

… and some in which money may play little or no role.

– excerpt from *Who Gets What – and Why*

MATCHING MARKETS

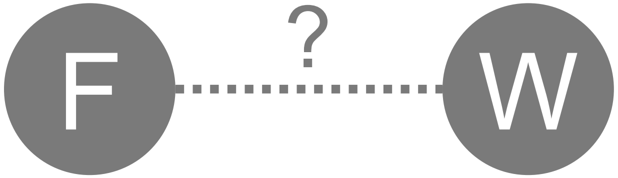
In matching problems, prices do not do all – or any – of the work

Agents are paired with other (groups of) agents, transactions, or contracts

- Workers to firms
- Children to schools
- Residents to hospitals
- Patients to donors
- Advertisements to viewers
- Riders to rideshare drivers

UNCERTAINTY

- Does a matched edge truly exist?
- How valuable is a match?
- Will a better match arrive in the future?



COMPETITION

Rival matching markets compete over the same agents

- **How does this affect global social welfare?**
- **How to differentiate?**

MATCH CADENCE

How quickly do new edges form?

How frequently does a market clear?

Is clearing centralized or decentralized?

Can agents reenter the market?

Use data & optimization – alongside human domain expertise – to learn matching policies

Strong theoretical underpinnings provide design guidance & runtime guarantees

THIS TALK

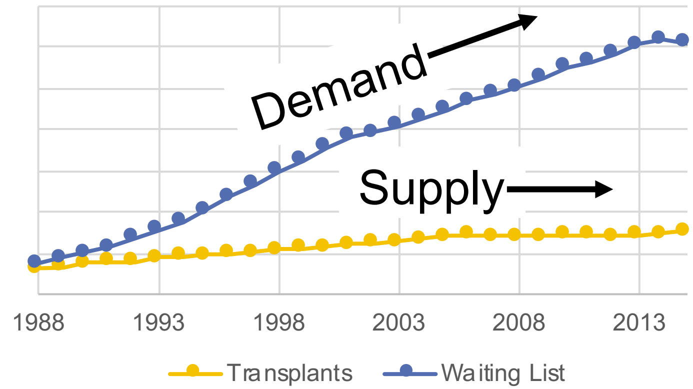
- **Four dimensions of matching market design:**
  - Managing short-term uncertainty
  - Balancing equity & efficiency
  - Combining human input and optimization
  - Incentives & mechanism design
- ***(Each is supported by my work with local and nationwide kidney exchanges)***
- **Also, some open problems!**

Covers recent and ongoing work – talk to me for details!
Publications: jpdickerson.com/pubs.html

KIDNEY EXCHANGE

KIDNEY TRANSPLANTATION

- **US waitlist: about 100,000**
  - 35,587 added in 2017
- **4,044 people died while waiting**
- **14,022 people received a kidney from the deceased donor waitlist**



- **5,794 people received a kidney from a living donor**
  - Some through **kidney exchanges**! [Roth et al. 2004]
  - This talk: experience with UNOS national kidney exchange (and some data from the NHS NLDKSS)

UNOS DONATE LIFE

NHS

TRIED-AND-TRUE: DECEASED-DONOR ALLOCATION

Online bipartite matching problem:

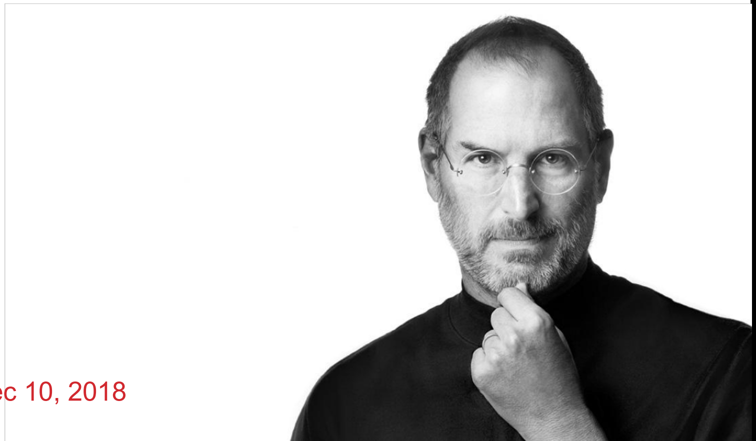
- Set of patients is known (roughly) in advance
- Organs arrive and must be dispatched quickly

Constraints:

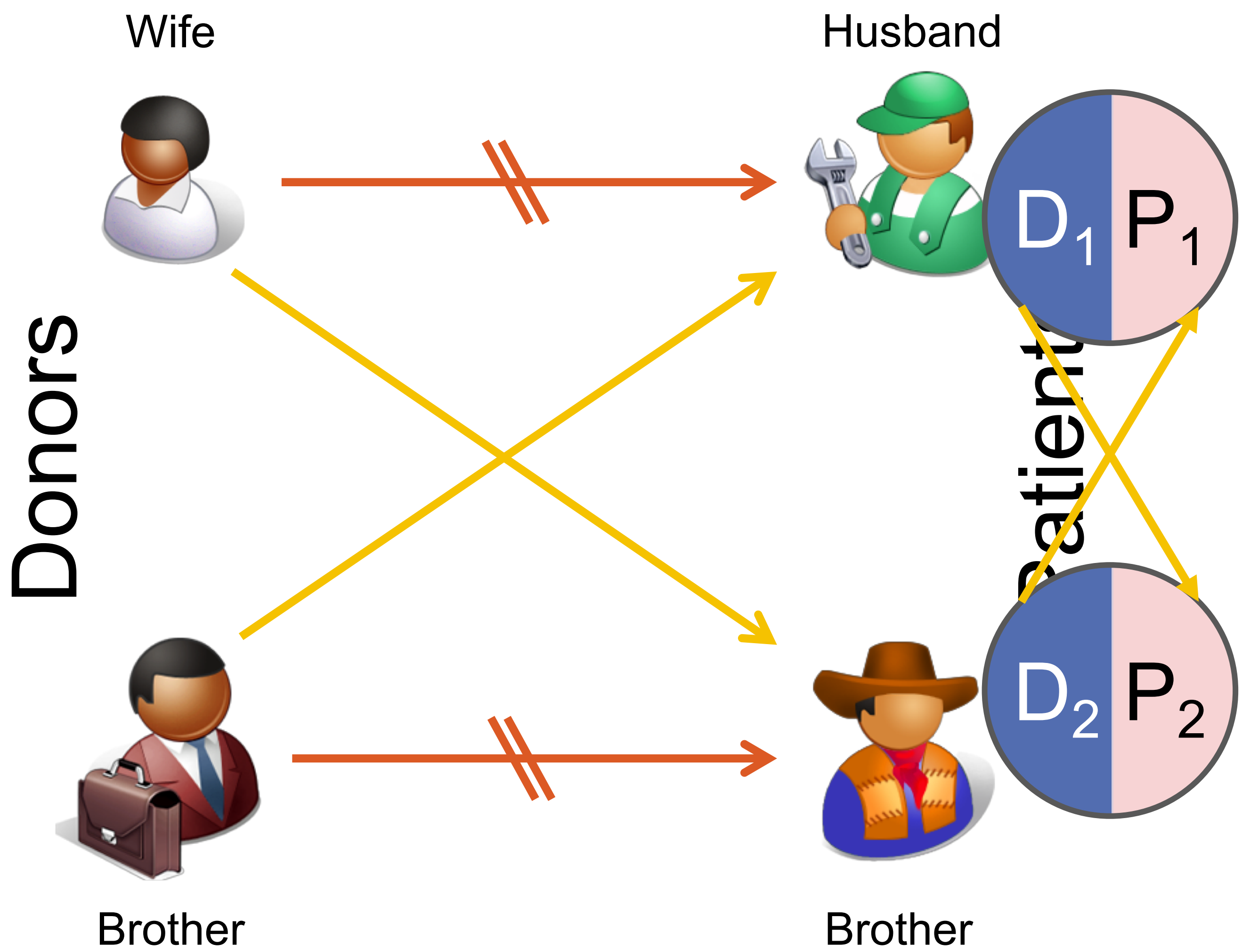
- Locality: organs only stay good for 24 hours
- Blood type, tissue type, etc.

Who gets the organ? Prioritization based on:

- Age?
- QALY maximization?
- Quality of match?
- Time on the waiting list?



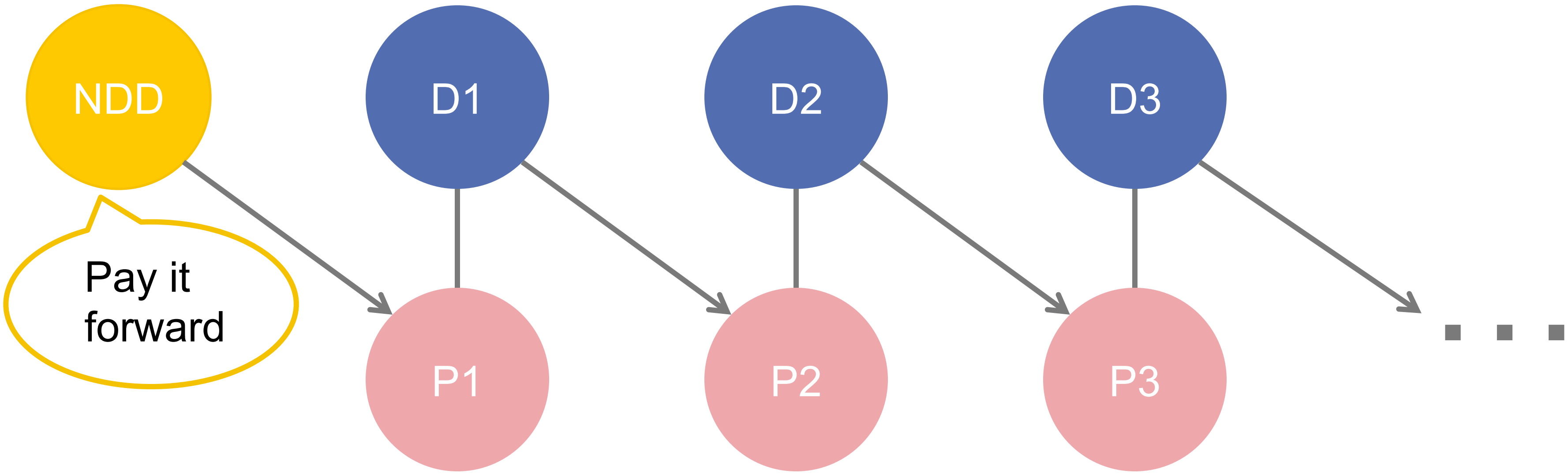
KIDNEY EXCHANGE



(2- and 3-cycles, all surgeries performed simultaneously)

NON-DIRECTED DONORS & CHAINS

[Rees et al. 2009]



Not executed simultaneously, so no length cap required based on logistic concerns …

… **but in practice edges fail, so *some* finite cap is used!**

REAL-WORLD IMPACT

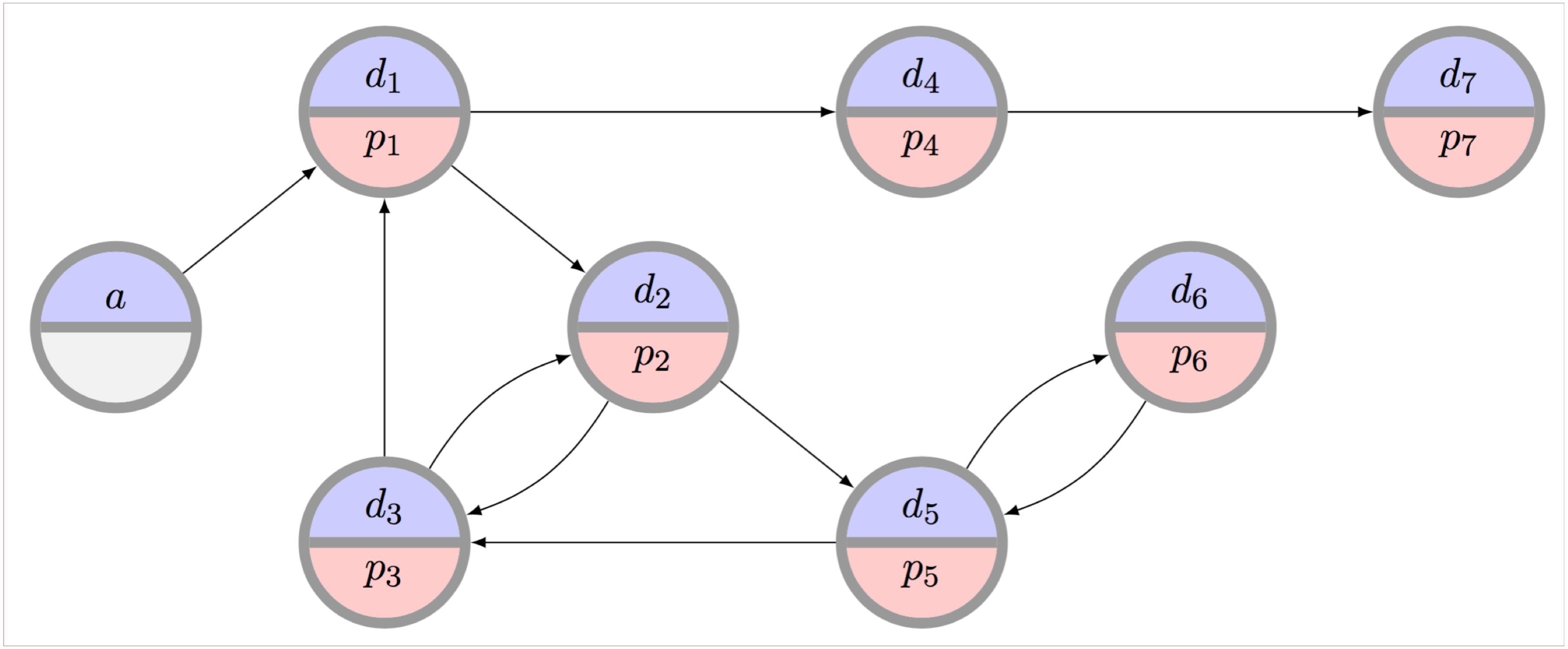
Kidney exchange is only a decade young, but already accounts for >12% of living donations in the United States

- Now a worldwide phenomenon (AU, CA, IL, PT, TR, UK, …)
- (Slowly) moving toward organized international exchange

Extensive experience with, e.g., the United Network for Organ Sharing (UNOS) US nationwide kidney exchange!

- 155+ transplant centers (roughly 69% of the US)
- Completely autonomous biweekly match runs
- Only automated exchange in the US

THE CLEARING PROBLEM



The clearing problem is to find the "best" disjoint set of cycles of length at most *L*, and chains

- Typically, $2 \leq L \leq 5$ for kidneys (e.g., L=3 at UNOS)
- NP-hard (for L>2) in theory, really hard in practice [Glorie et al. 2014, Anderson et al. 2015, Plaut et al. 2016, Dickerson et al. 2016 ...]
  [Abraham et al. 07, Biro et al. 09]

A SIMPLE INTEGER PROGRAM

("Best" = max weight, myopic matching)

[Roth et al. 04, 05, Abraham et al. 07]

Binary variable x_c for each feasible cycle or chain c

Maximize

$$u(M) = \Sigma\, w_c\, x_c$$

Subject to

$$\Sigma_{c\,:\,i \text{ in } c}\, x_c \leq 1 \text{ for each vertex } i$$

"SIMPLE" …?

THE BIG PROBLEM

What is “best”?

- Maximize matches right now or over time?
- Maximize transplants or matches?
- Prioritization schemes (i.e. fairness)?
- Modeling choices?
- Incentives? Ethics? Legality?

Optimization can handle this, but may be inflexible in hard-to-understand ways (for humans)

Want humans in the loop at a **high level**
(and then CS/Opt handles the implementation)

MANAGING SHORT-TERM UNCERTAINTY

[EC-13, EC-15, EC-16, Management Science 18, AAAI-19]

With A. Blum, N. Haghtalab, D. Manlove, D. McElfresh, B. Plaut, A. Procaccia, T. Sandholm, A. Sharma, J. Trimble

MATCHED ≠ TRANSPLANTED

Only around 10-15% of UNOS matched structures result in an actual transplant

- Similarly low % in other exchanges [ATC 2013]

Many **reasons for this. How to handle?**

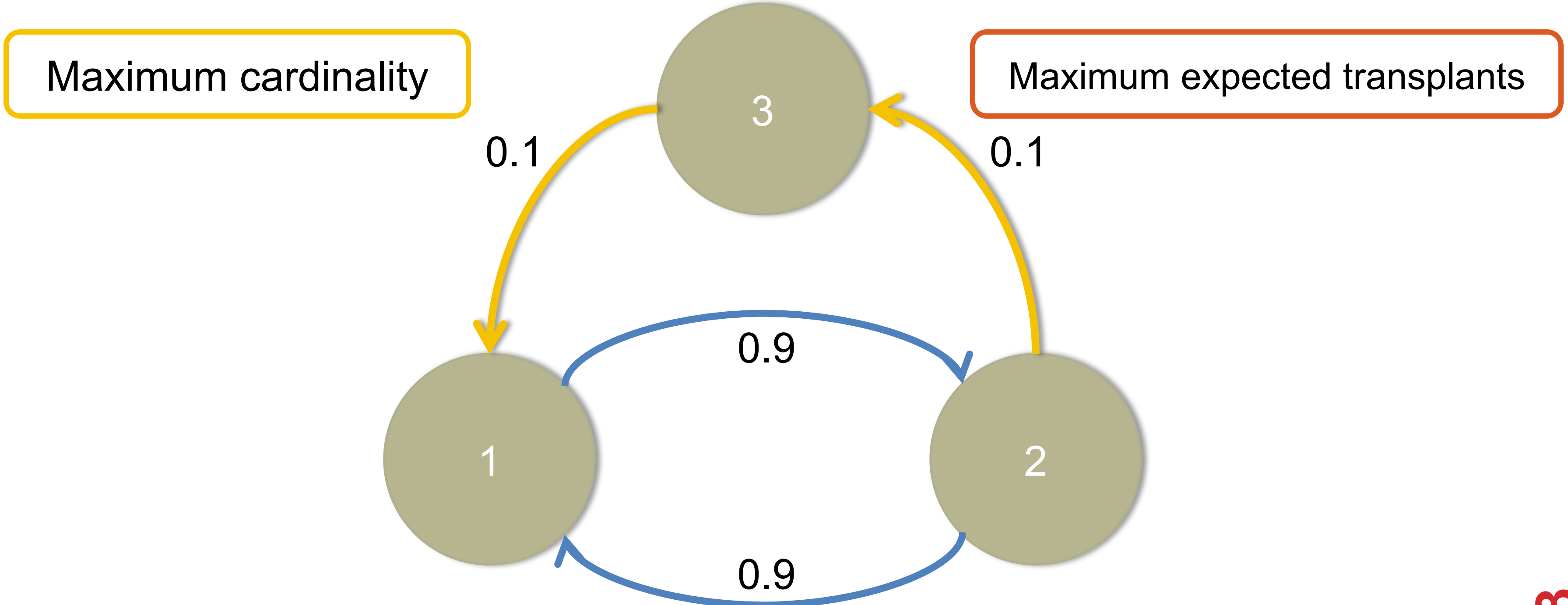
One way: encode *probability of transplantation* rather than just feasibility

- for individuals, cycles, chains, and full matchings

DISCOUNTED CLEARING PROBLEM

("Best" = max expected cardinality | limited recourse)

Find matching M^* with highest discounted utility



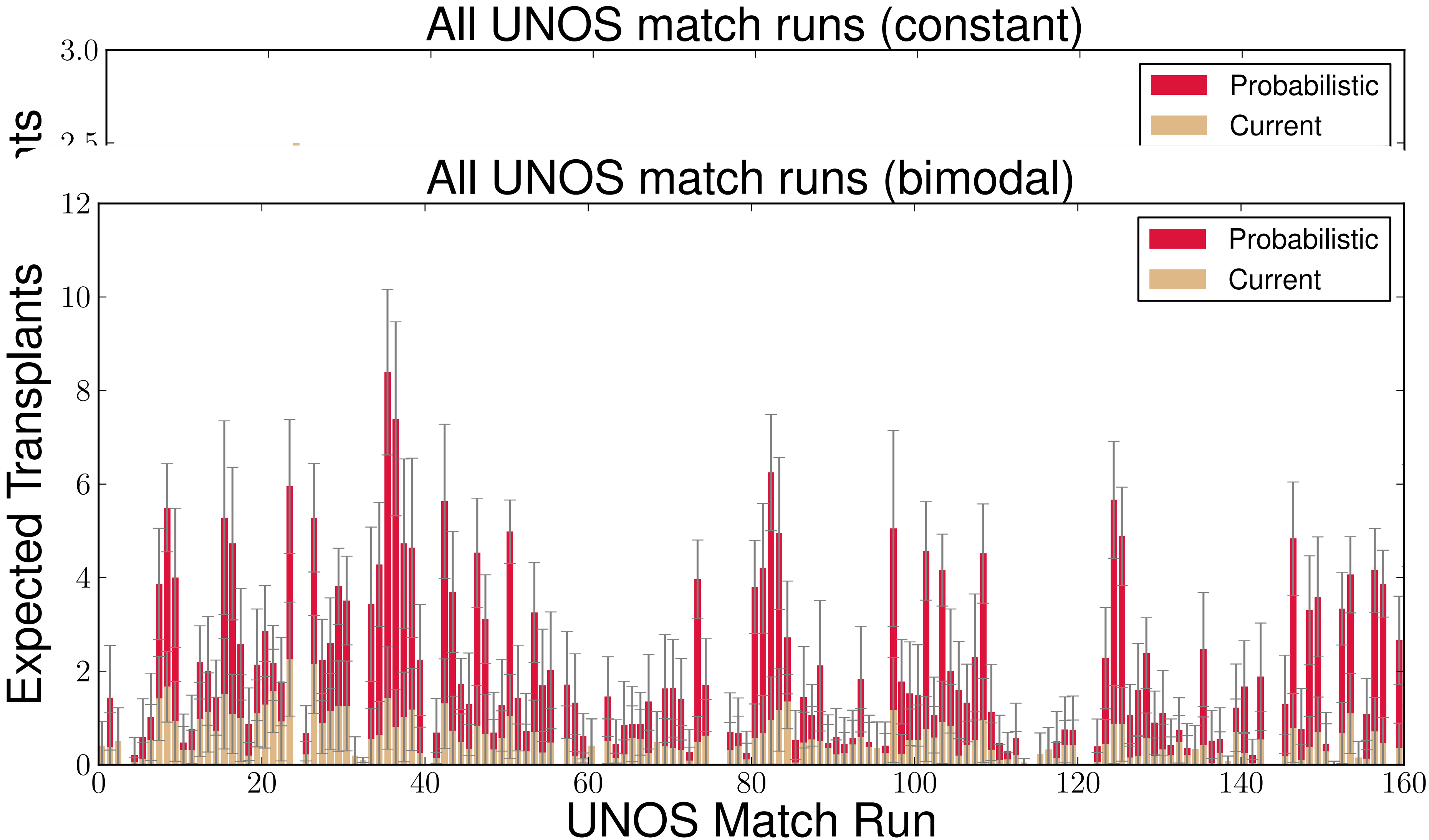
SOLVING THIS NEW PROBLEM

Theorem:
In a sparse random graph model, for any failure probability *p*, w.h.p. there exists a matching that is "linearly better" than *any* max-cardinality matching

Practice: Solved via branch-and-price

- One binary decision variable per cycle/chain
- Upper-bounding is now NP-hard ❌
- Pricing problem is (empirically) much easier ✅

Maybe this is a good idea …



Under discussion for implementation at UNOS

PRE-MATCH EDGE TESTING

Idea: perform a *small amount* of costly testing before a match run to test for (non)existence of edges

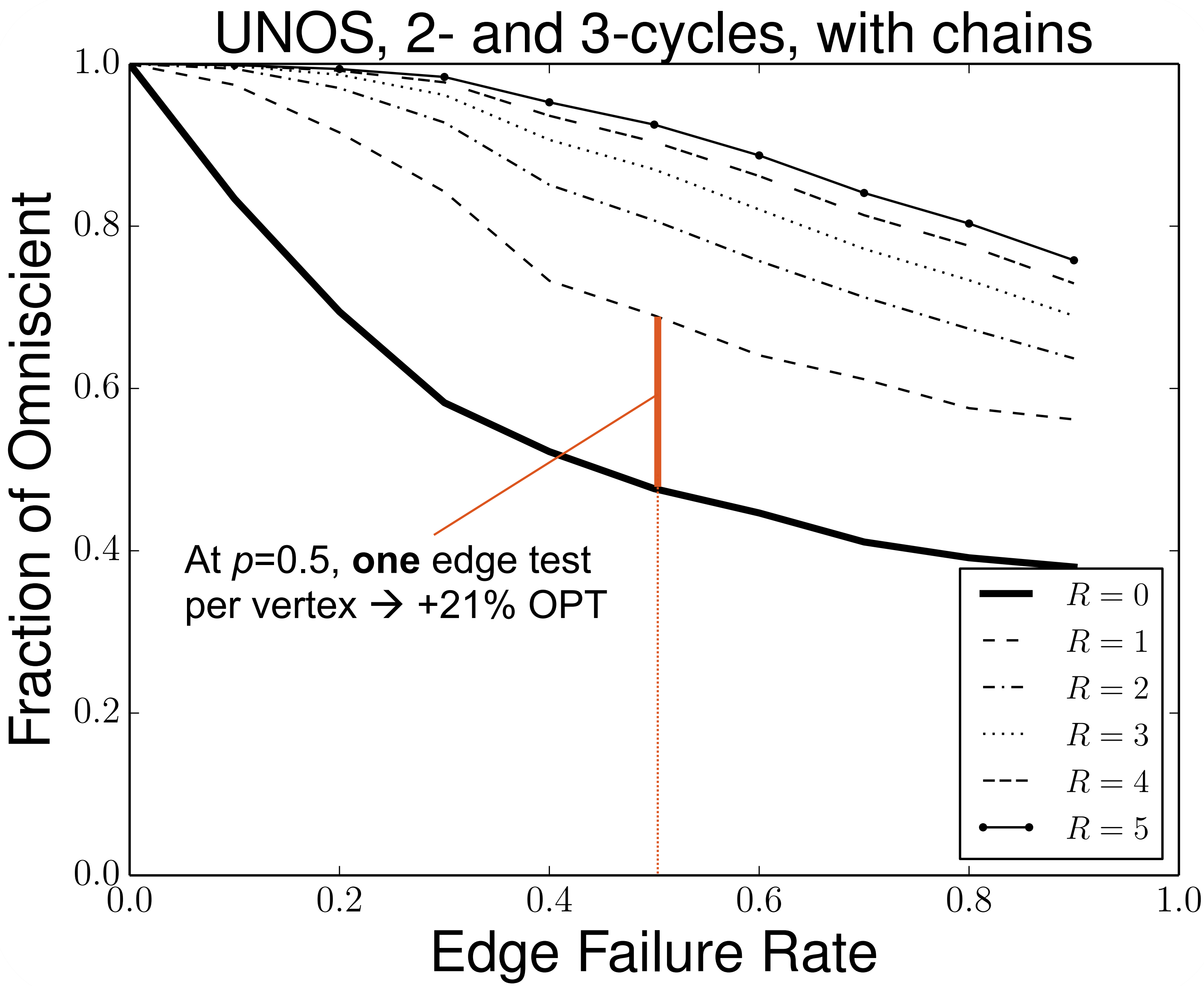
- E.g., more extensive medical testing, donor interviews, surgeon interviews, …

Cast as a *stochastic matching* problem:

Given a graph $G(V,E)$, choose subset of edges S such that:

$$|M(S)| \geq (1-\varepsilon)\ |M(E)|$$

Need: “sparse” S, where every vertex has O(1) incident tested edges



Even 1 or 2 extra tests would result in a huge lift

In theory and practice, we're helping the global bottom line by considering post-match failure …

… But can this hurt some individuals?

BALANCING FAIRNESS AND EFFICIENCY

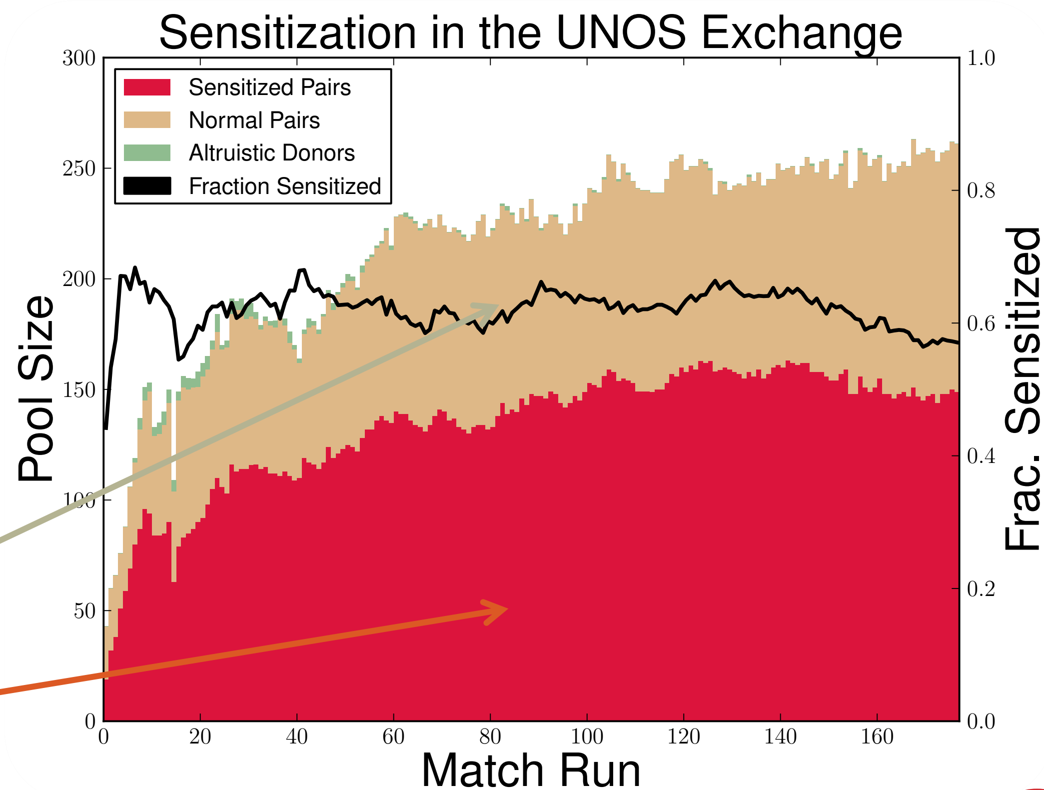
[AAMAS-14, AAAI-15, AAAI-18, Invited to AIJ, u.r. 2018]

With D. McElfresh, A. Procaccia and T. Sandholm

SENSITIZATION IN KIDNEY TRANSPLANTATION

Highly-sensitized patients: unlikely to be compatible with a random donor

- Deceased donor waitlist: 17%
- Kidney exchanges: **much** higher (60%+)



"Easy to match" patients

"Hard to match" patients

PRICE OF FAIRNESS

Efficiency vs. fairness:

- **Utilitarian** objectives may favor certain classes at the expense of marginalizing others
- **Fair** objectives may sacrifice efficiency in the name of egalitarianism

Price of fairness: relative system efficiency loss under a fair allocation [Bertismas, Farias, Trichakis 2011] [Caragiannis et al. 2009]

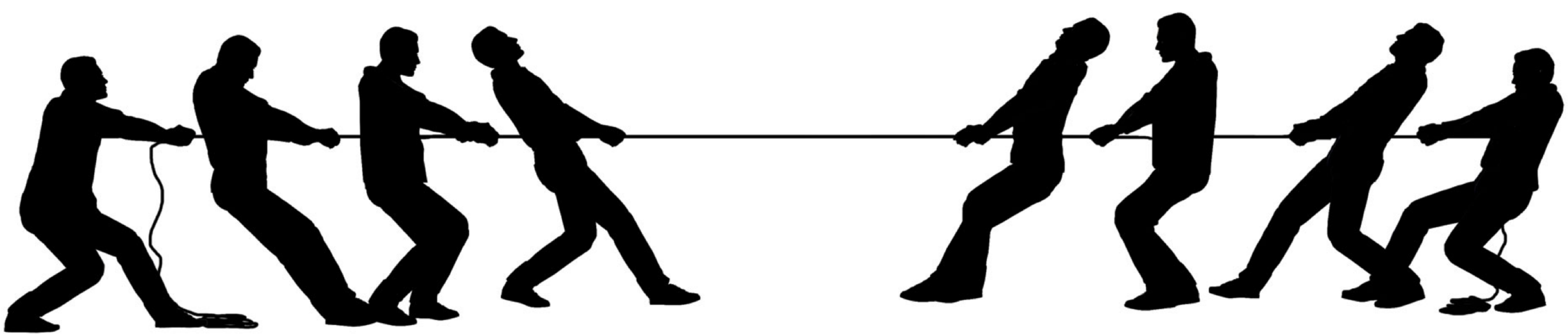
- Very applicable to kidney exchange!

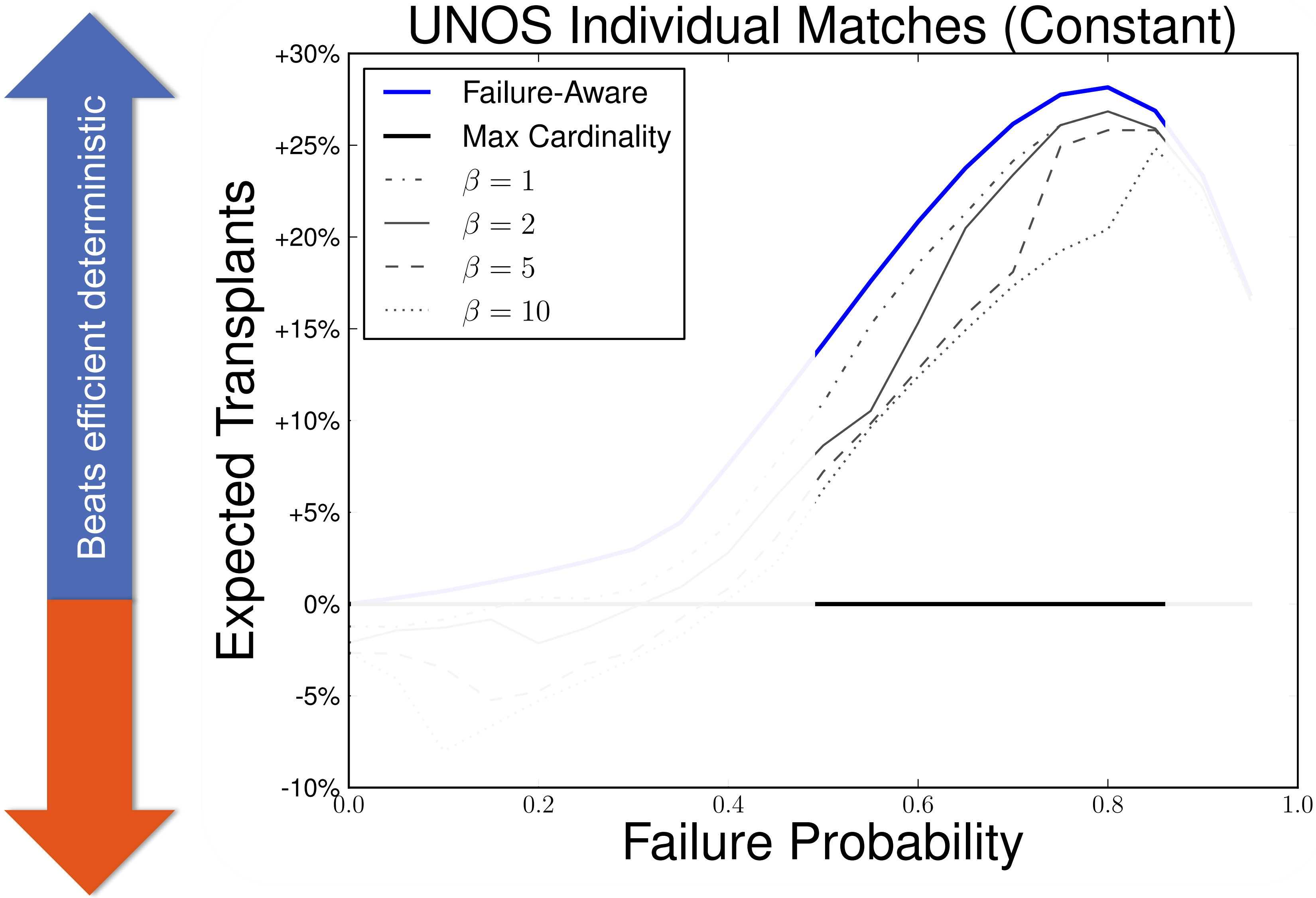
CONTRADICTORY GOALS

Earlier, we saw failure-aware matching results in tremendous gains in #expected transplants

Gain comes at a price – may further marginalize hard-to-match patients because:

- Highly-sensitized patients tend to be matched in chains
- Highly-sensitized patients may have higher failure rates (in, e.g., APD data, not in UNOS data)





UNOS runs, weighted fairness, constant probability of failure (x-axis), increase in expected transplants over deterministic matching (y-axis)

Fairness vs. efficiency can be balanced in theory and in practice in a static model …

… But how should we match over time?

LEARNING TO MATCH IN A DYNAMIC ENVIRONMENT

[AAAI-12, AAAI-15, NIPS-15 MLHC, w.p. 2018]

With M. Curry, D. McElfresh, C. Moy, A. Procaccia, and T. Sandholm

DYNAMIC KIDNEY EXCHANGE

Kidney exchange is a naturally dynamic event

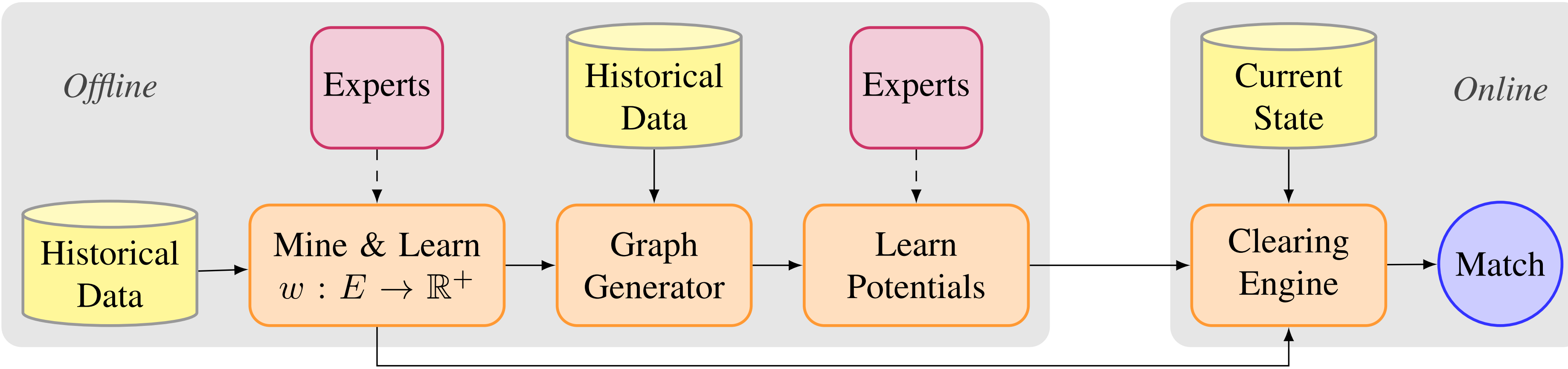
Can be described by the evolution of its graph

- Additions, removals of edges and vertices

Vertex Removal	Edge Removal	Vertex/Edge Add
Transplant, this exchange	Matched, positive crossmatch	Normal entrance
Transplant, deceased donor waitlist	Matched, candidate refuses donor	
Transplant, other exchange ("sniped")	Matched, donor refuses candidate	
Death or illness	Pregnancy, sickness changes HLA	
Altruist runs out of patience		
Bridge donor reneges		

How should we balance matching now versus waiting to match?

FUTUREMATCH: LEARNING TO MATCH IN DYNAMIC ENVIRONMENTS



Offline (run once or periodically)

1. Domain expert describes overall goal
2. Take historical data and policy input to learn a weight function *w* for match quality
3. Take historical data and create a graph generator with edge weights set by *w*
4. Using this generator and a realistic exchange simulator, learn potentials for graph elements as a function of the exchange dynamics

Online (run every match)

1. Combine *w* and potentials to form new edge weights on real input graphs
2. Solve maximum weighted matching and return match

PITTSBURGH SUPERCOMPUTING CENTER

Best Data-Intensive Application/Use

HPCwire Readers' Choice Awards 2014

Presented at Supercomputing

Tied with IBM Watson

EXPERIMENTAL RESULTS & IMPACT

We show it is possible to:

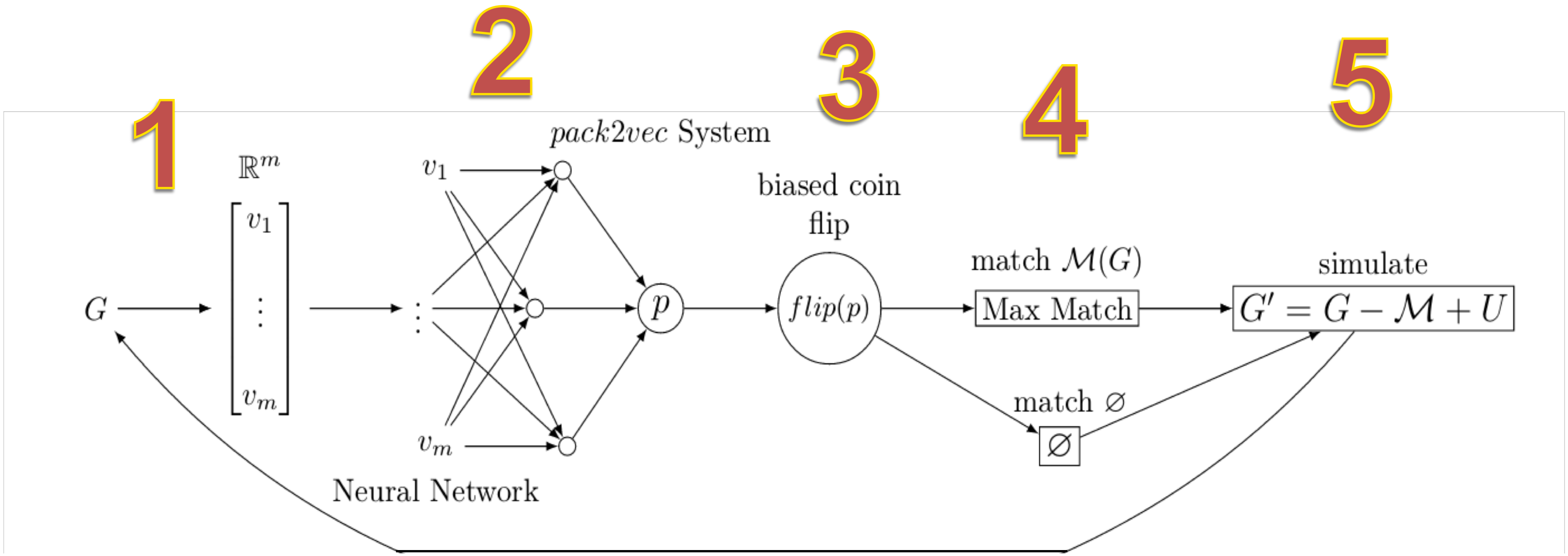
- Increase overall #transplants a lot at a (much) smaller decrease in #marginalized transplants
- Increase #marginalized transplants a lot at no or very low decrease in overall #transplants
- Increase both #transplants and #marginalized

Sweet spot depends on distribution:

- Luckily, we can generate – and learn from – realistic families of graphs!

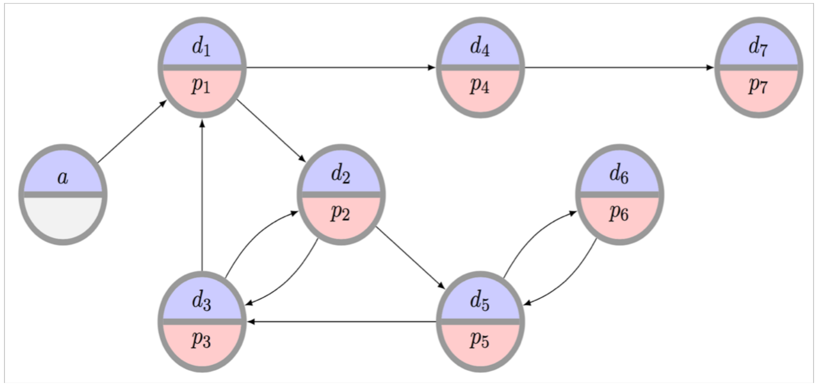
FutureMatch now used for policy recommendations at UNOS

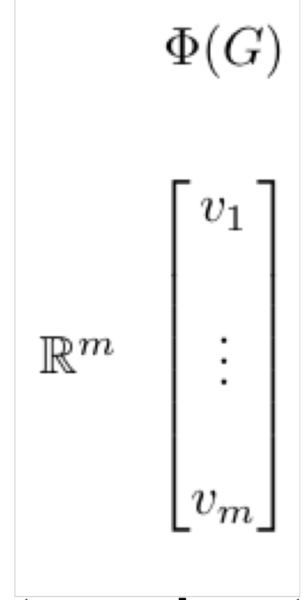
LEARNING & DYNAMIC KIDNEY EXCHANGE



1. **Embed** current compatibility graph into fixed-dimensional space
2. **Neural network** uses those vectors to learn appropriate policy
3. Flip a **biased coin**
4. If heads: find and **match** maximum cardinality matching
5. **Simulate** kidney exchange environment and grow the graph

1. EMBEDDING





Neural networks generally **take a fixed-sized vector as input**

- Our state space: graphs of any size
- Need: embed the graph as a vector and still maintain certain properties, such as node neighborhood structure. We use random walks to do so [Li, Campbell, Caceres 2017]

Use random walk on a carefully selected initial distribution to capture temporal changes in probability distribution

- Encode distance between pairs of probability distributions
- Empirically, this approach can distinguish between, e.g., **Erdős–Rényi** and **Stochastic Block Model** graphs

2. EMBEDDING TO NEURAL NET

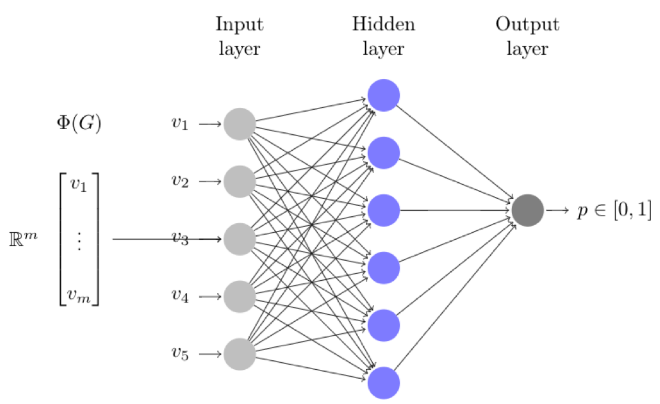
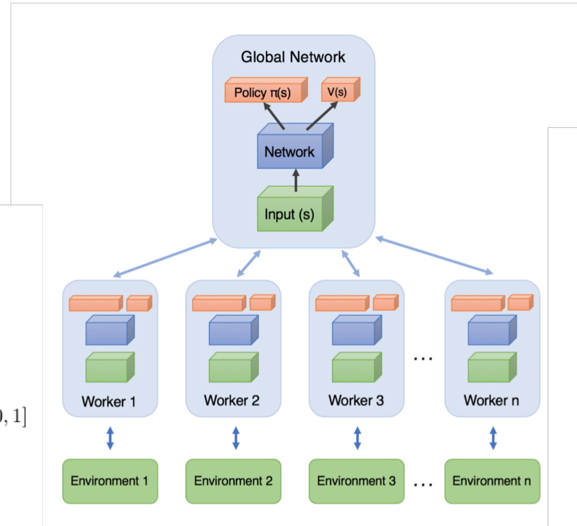
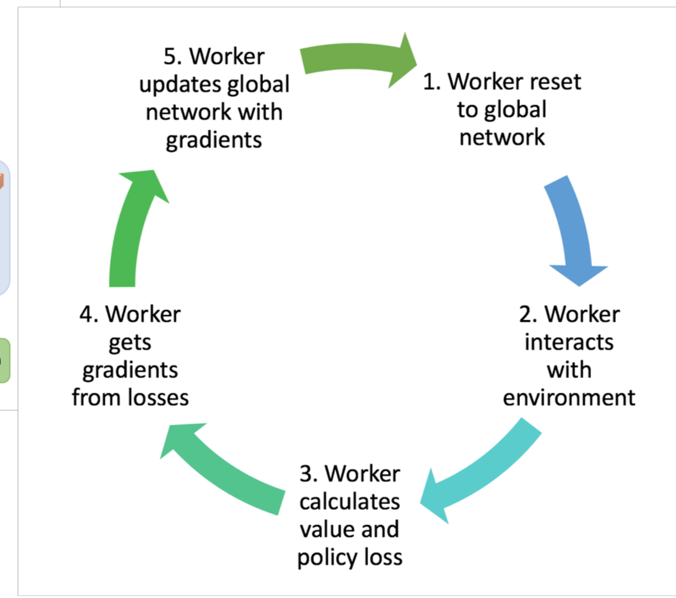


Figure 1: High-level neural network architecture.



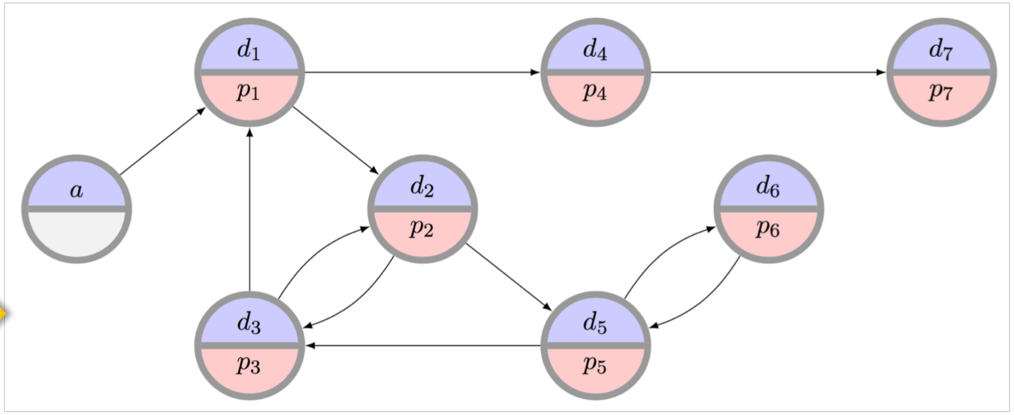


Feed an embedded graph into, e.g., a neural network to output a learned probability for our biased coin flip

- (Currently, using an adaptation of Asynchronous Advantage Actor-Critic (A3C) method [Mnih 2016])

3. BIASED COIN FLIP W/LEARNED PROBABILITY

1



MAX MATCH

0

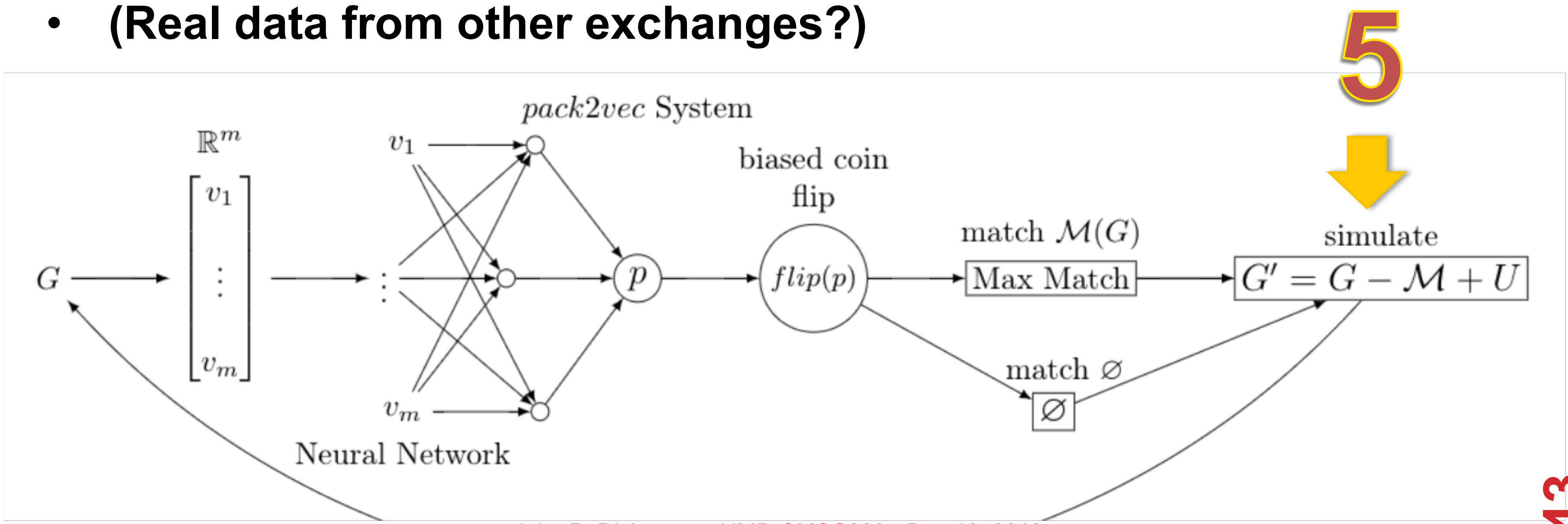
MATCH NOTHING (wait)

4. MAX MATCHING (THE CLEARING PROBLEM)—OR NOT

5. KIDNEY EXCHANGE SIMULATION – CHANGING THE INPUT GRAPH

To train the neural network, we must be able to simulate kidney exchange (graphs). We use several evolution models.

- Homogeneous Erdős–Rényi graphs [Akbarpour et al. 2017+]
- Heterogeneous Erdős–Rényi graphs [Ashlagi et al. 2013+]
- Real data from the UNOS exchange
- **(Real data from other exchanges?)**



EARLY RESULTS

We replicate results from prior theory papers:

- In some models, dynamic matching helps
- In some models, dynamic matching does not help

Still iterating on:

- Neural net structure
- Action space (binary coin flip vs. multiple match types)
- Learning method (A3C vs. DQN vs. more standard methods)

But …

- **Seems promising.** Can learn matching policies beyond simply batching for T time periods; can realize gains over greedy.
- **Policies depend on graph structure.**

COMPETITION WITHIN, AND BETWEEN, KIDNEY EXCHANGES

[AAAI-15, AMMA-15, IJCAI-18, w.p. 2018]

With S. Das, N. Gupta, C. Hajaj, A. Hassidim, Z. Li, T. Sandholm, and D. Sarne

MANAGING INCENTIVES

Clearinghouse cares about global welfare:

- How many patients received kidneys (over time)?

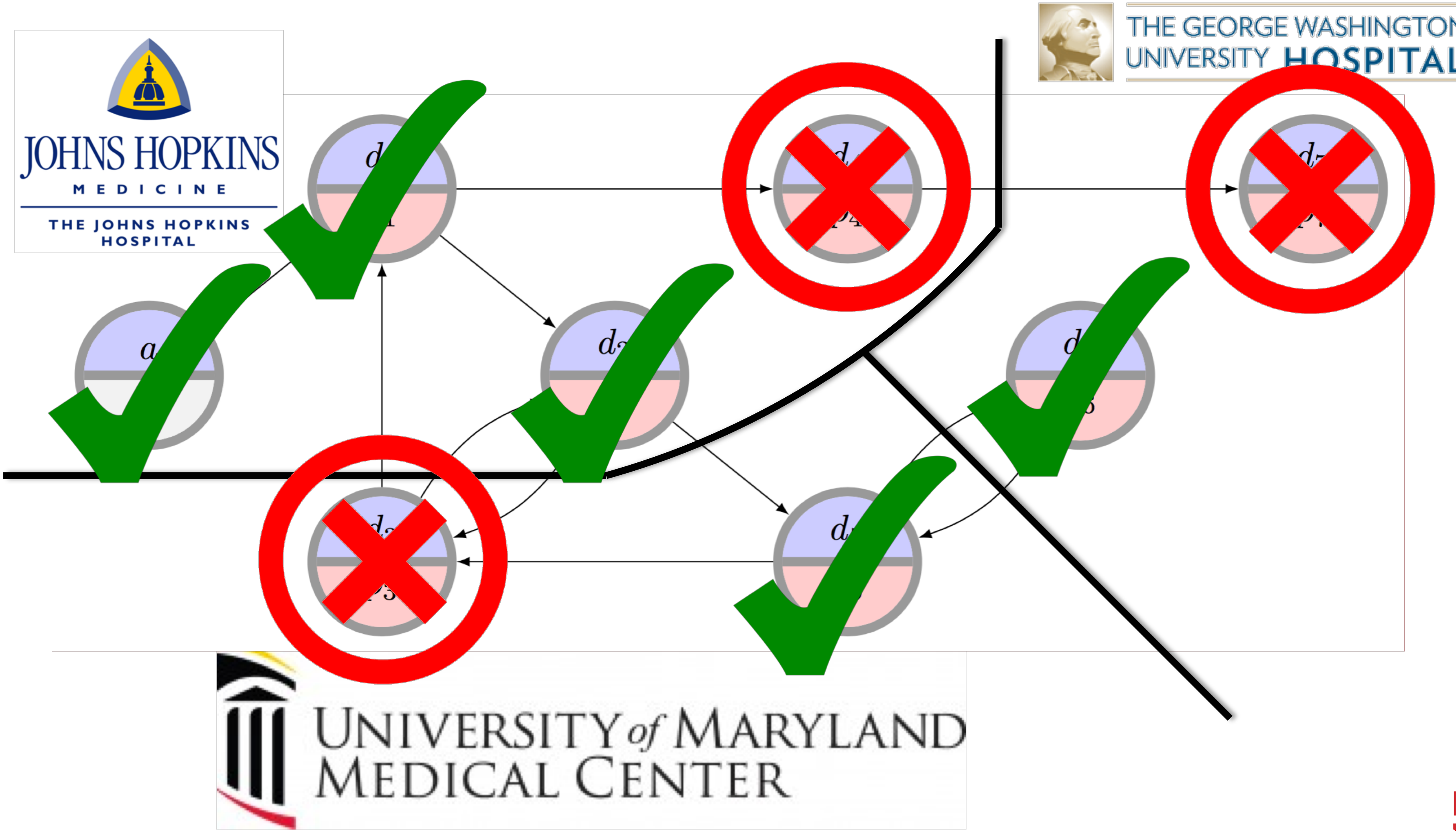
Transplant centers care about their individual welfare:

- How many of my own patients received kidneys?

Patient-donor pairs care about their individual welfare:

- Did I receive a kidney?
- (Most work considers just clearinghouse and centers)

PRIVATE VS GLOBAL MATCHING



0%

RACE TO THE BOTTOM

In the US, there are multiple competing exchanges:

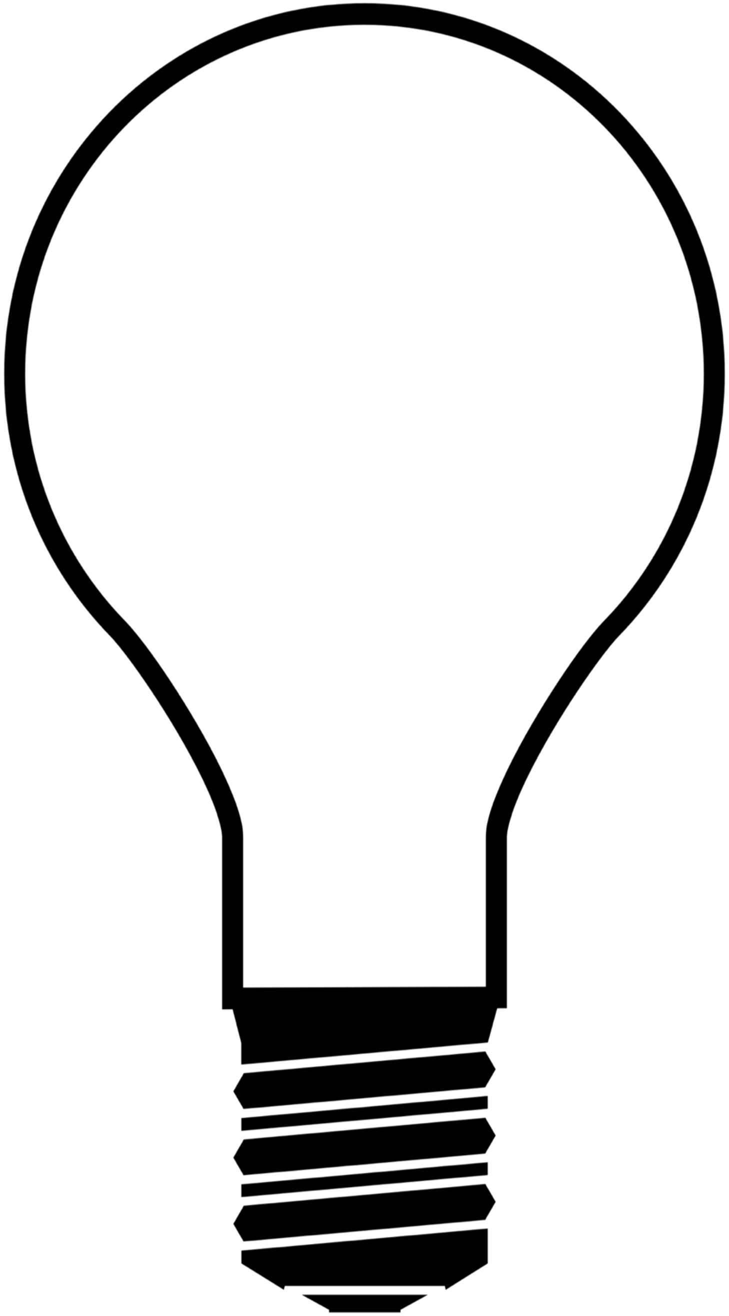
- UNOS, NKR, APD, …
- Single-center "centralized exchanges"

What about international exchange?

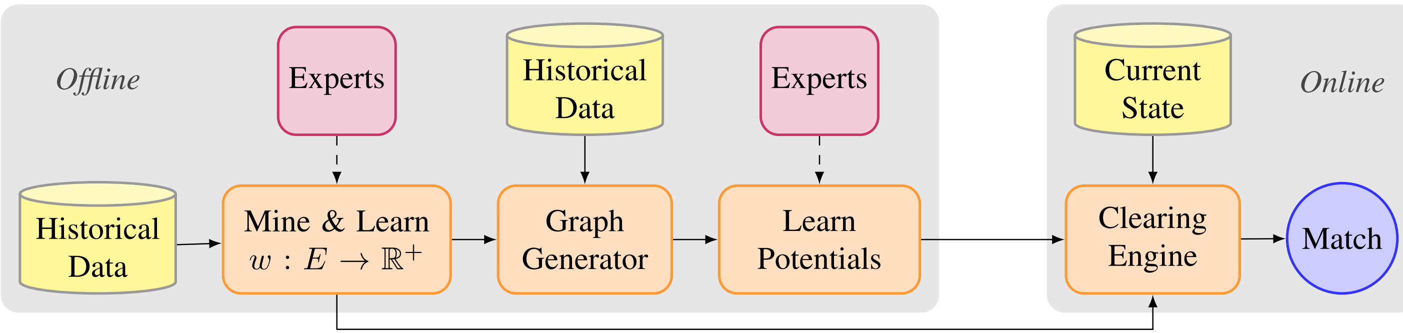
- EU COST Action to investigate connection of exchanges

Fragmenting the market results in:

- Higher short-term failure rates
- Fewer matching opportunities
- Higher (aka greedier, "myopic") match speed
- Overall efficiency loss (in theory, simulation, and reality)



QUESTIONS?



More information:

http://jpdickerson.com

Code:

/JohnDickerson/KidneyExchange

Joint work with:

Funding & support:

NDSEG Fellowship

Facebook Fellowship

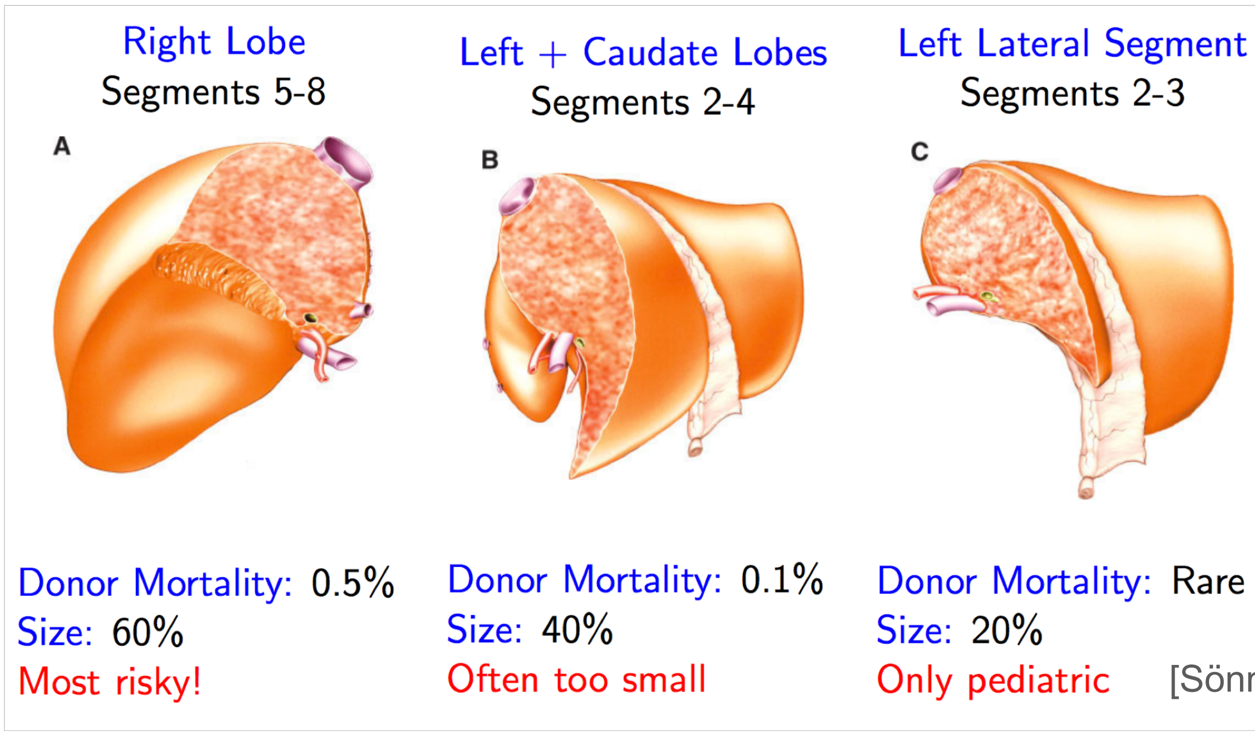
BACKUP SLIDES

JOHN P DICKERSON

THE CUTTING EDGE

MOVING BEYOND KIDNEYS: LIVERS [Ergin, Sönmez, Ünver *w.p. 2015*]

Similar matching problem (mathematically)



[Sönmez 2014]

Right lobe is biggest but riskiest; exchange may reduce right lobe usage and increase transplants

MOVING BEYOND KIDNEYS: MULTI-ORGAN EXCHANGE

[Dickerson Sandholm AAAI-14, JAIR-17]

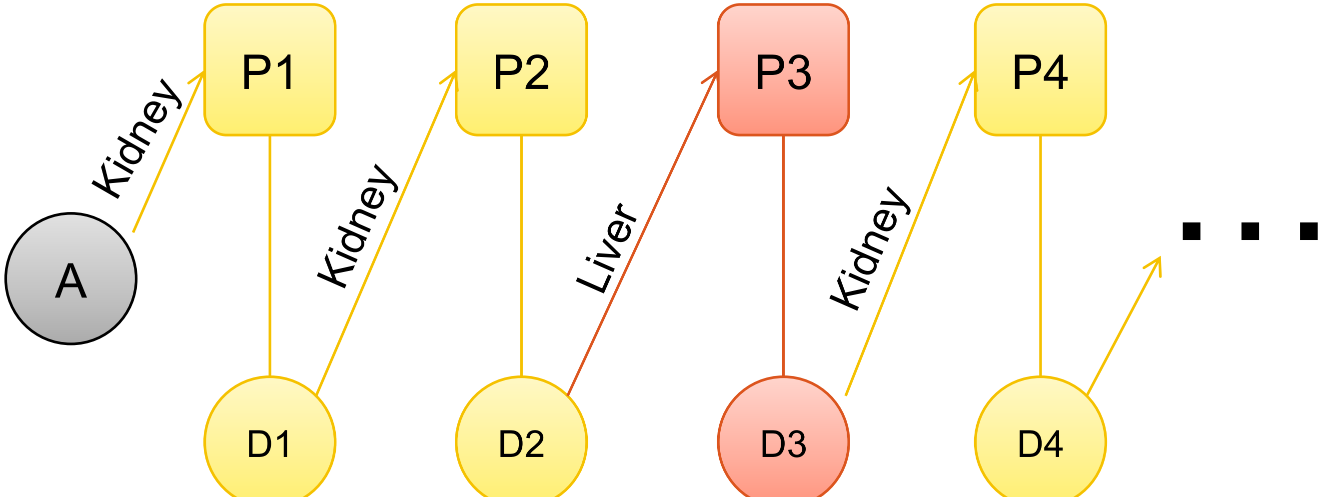
Chains are great! [Anderson et al. 2015, Ashlagi et al. 2014, Rees et al. 2009]

Kidney transplants are "easy" and popular:

- Many altruistic donors

Liver transplants: higher mortality, morbidity:

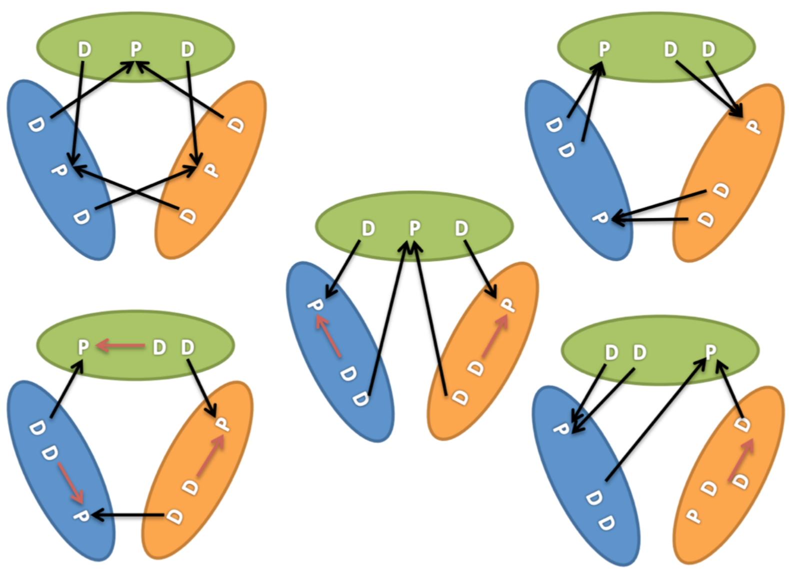
- (Essentially) no altruistic donors



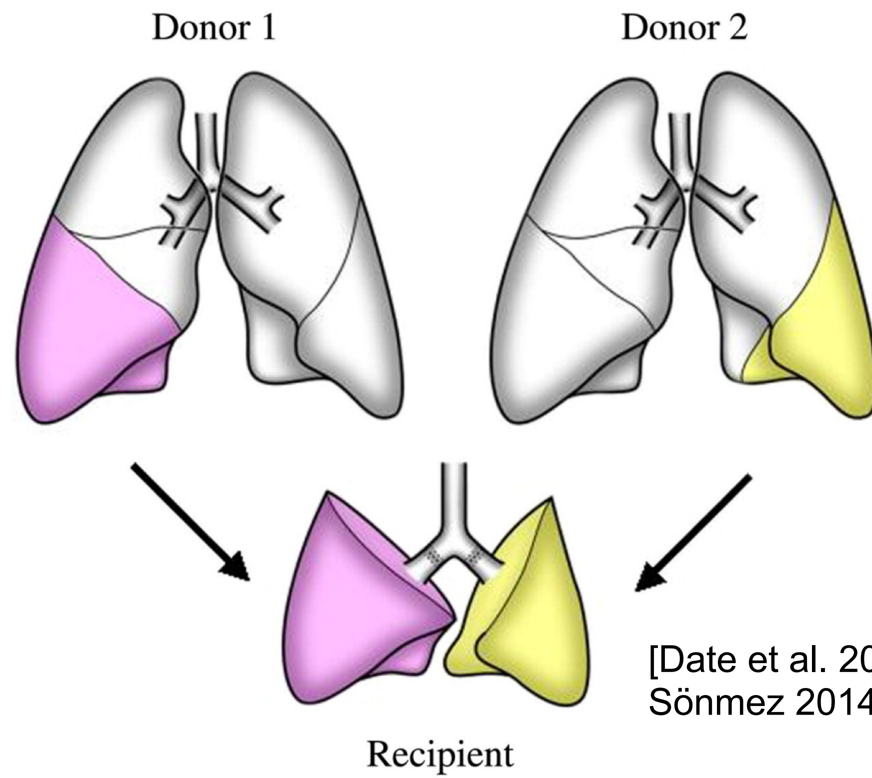
MOVING BEYOND KIDNEYS: LUNGS [Ergin, Sönmez, Ünver *w.p. 2014*]

Fundamentally different matching problem

- **Two** donors needed



3-way lung exchange configurations



[Date et al. 2005; Sönmez 2014]

(Compare to the single configuration for a “3-cycle” in kidney exchange.)

FAILURE-AWARE MODEL

Compatibility graph *G*

- Edge (v_i, v_j) if v_i's donor can donate to v_j's patient
- Weight w_e on each edge e

Success probability q_e for each edge e

Discounted utility of cycle *c*

$$u(c) = \sum w_e \cdot \prod q_e$$

Value of successful cycle

Probability of success

FAILURE-AWARE MODEL

Discounted utility of a *k*-chain *c*

$$u(c) = \left[\sum_{i=1}^{k-1} (1 - q_i) i \prod_{j=0}^{i-1} q_j \right] + \left[k \prod_{i=0}^{k-1} q_i \right]$$

Exactly first *i* transplants

Chain executes in entirety

Cannot simply "reweight by failure probability"

Utility of a match *M*: ***u*(*M*) = ∑ *u*(*c*)**

INCREMENTALLY SOLVING VERY LARGE IPS

#Decision variables grows linearly with #cycles and #chains in the pool

- Millions, billions of variables
- Too large to fit in memory

Branch-and-price incrementally brings variables into a reduced model [Barnhart et al. 1998]

Solves the "pricing problem" – each variable gets a real-valued price

- Positive price → resp. constraint in full model violated
- No positive price cycles → optimality at this node

CONSIDERING ONLY "GOOD" CHAINS

Theorem:
Given a chain *c*, any extension *c'* will not be needed in an optimal solution if the infinite extension has non-positive value.

$$\left(\frac{q_{max}}{1-q_{max}}\prod_{i=0}^{k-1} q_i\right) + u(c) + \ell - \left(d_{min} + \sum_{i=0}^{k} d_i\right) \leq 0$$

- Optimistic future value of infinite extension
- Discounted utility of current chain
- Donation to waitlist
- Pessimistic sum of LP dual values in model

$G(n, t(n), p)$: random graph with

- n patient-donor pairs
- $t(n)$ altruistic donors
- Probability $\Theta(1/n)$ of incoming edges

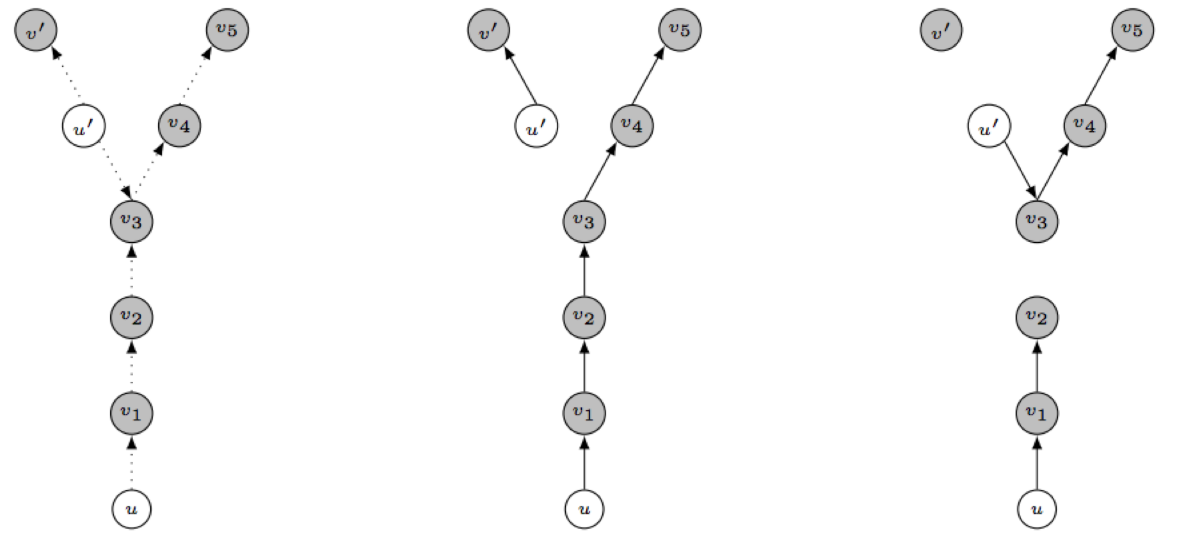
Constant transplant success probability q

Theorem

For all $q \in (0,1)$ and α, $\beta > 0$, given a large $G(n, \alpha n, \beta/n)$, w.h.p. there exists some matching M' s.t. for every maximum cardinality matching M,

$$u_q(M') \geq u_q(M) + \Omega(n)$$

BRIEF INTUITION: COUNTING Y-GADGETS



(a) A Y gadget. (b) The maximum cardinality matching M_Y. (c) The matching M'_Y.

For every structure *X* of constant size, w.h.p. can find $\Omega(n)$ structures isomorphic to *X and isolated from the rest of the graph*

Label them (alt vs. pair): flip weighted coins, constant fraction are labeled correctly → constant × $\Omega(n)$ = $\Omega(n)$

Direct the edges: flip 50/50 coins, constant fraction are entirely directed correctly → constant × $\Omega(n)$ = $\Omega(n)$

Under the “most stringent” fairness rule:

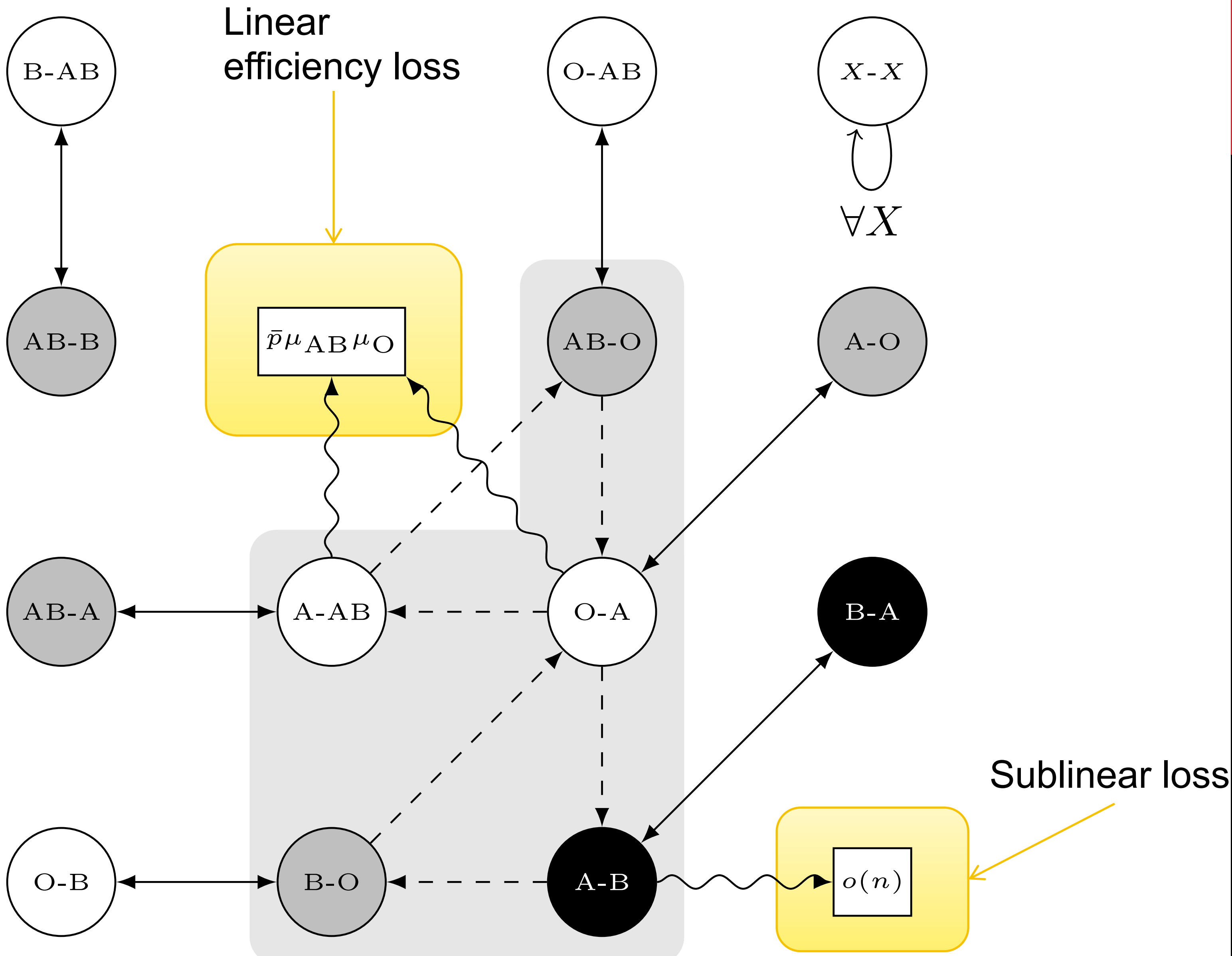
$$u_{H \succ L}(M) = \begin{cases} u(M) & \text{if } |M_H| = \max_{M' \in \mathcal{M}} |M'_H| \\ 0 & \text{otherwise} \end{cases}$$

Theorem

Assume “reasonable” level of sensitization and “reasonable” distribution of blood types. Then, almost surely as $n \to \infty$,

$$\text{POF}(\mathcal{M}, u_{H \succ L}) \leq \frac{2}{33}.$$

(And this is achieved using cycles of length at most 3.)


Linear
efficiency loss
B-AB
O-AB
X-X
∀X
AB-B
$\bar{p}\mu_{AB}\mu_{O}$
AB-O
A-O
AB-A
A-AB
O-A
B-A
Sublinear loss
O-B
B-O
A-B
$o(n)$

BETTER STATIC OPTIMIZATION METHODS

Recall two main methods for solving big IPs for kidney exchange:

- Branch-and-price = B&B + column generation
- Constraint generation

Many different ways to do these:

- E.g., how do I solve the pricing problem?
- E.g., which constraints should I add to the model?

Big runtime changes [Anderson et al. PNAS-2015, Glorie et al. MSOM-2014]

BASIC EDGE FORMULATION

[Abraham et al. 07]

Binary variable x_{ij} for each edge from *i* to *j*

Maximize

$$u(M) = \Sigma\, w_{ij}\, x_{ij}$$

Flow constraint

Subject to

$\Sigma_j\, x_{ij} = \Sigma_j\, x_{ji}$ for each vertex *i*

$\Sigma_j\, x_{ij} \leq 1$ for each vertex *i*

$\Sigma_{1 \leq k \leq L}\, x_{i(k)i(k+1)} \leq L\text{-}1$ for paths i(1)…i(L+1)

(no path of length L that doesn't end where it started – cycle cap)

STATE OF THE ART FOR EDGE FORMULATION

[Anderson et al. PNAS-2015]

Builds on the prize-collecting traveling salesperson problem [Balas Networks-89]

- PC-TSP: visit each city (patient-donor pair) exactly once, but with the additional option to pay some penalty to skip a city (penalized for leaving pairs unmatched)

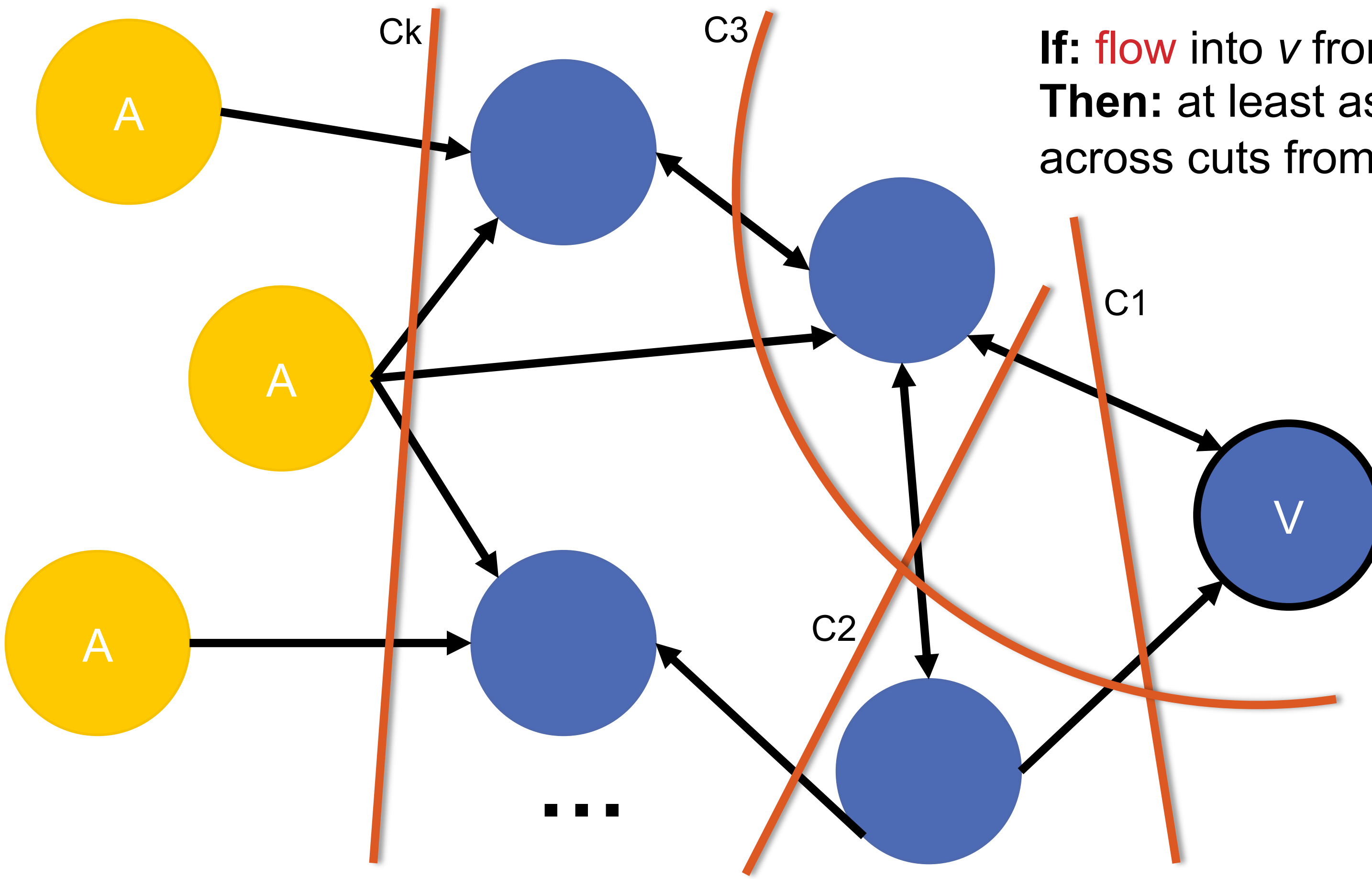
They maintain decision variables for all cycles of length at most *L*, but build chains in the final solution from decision variables associated with individual edges

Then, an exponential number of constraints could be required to prevent the solver from including chains of length greater than *K*; these are generated incrementally until optimality is proved.

- Leverage cut generation from PC-TSP literature to provide stronger (i.e. tighter) IP formulation

BEST EDGE FORMULATION

[Anderson et al. 15]



If: flow into *v* from a chain
Then: at least as much flow across cuts from {A}

REVIEW: CYCLE FORMULATION

Objective = maximum cardinality

Binary variable x_c for each cycle/chain c of length at most L

Maximize

$$\Sigma\ |c|x_c$$

Subject to

$$\Sigma_{c\ :\ i \text{ in } c}\ x_c \leq 1 \quad \text{for each vertex } i$$

DFS TO SOLVE PRICING PROBLEM *[Abraham et al. PNAS-2015]*

Pricing problem:

- Optimal dual solution $\boldsymbol{\pi}^*$ to reduced model
- Find non-basic variables with **positive price** (for a maximization problem)
  - 0 < weight of cycle – sum of duals in $\boldsymbol{\pi}^*$ of constituent vertices

First approach [Abraham et al. EC-2007] *explicitly* prices all feasible cycles and chains through a DFS

- Can speed this up in various ways, but proving **no positive price cycles exist** still takes time poly in chain/cycle cap = bad for even reasonable caps

THE RIGHT IDEA

Idea: solve structured optimization problem that implicitly prices variables

Price: $w_c - \Sigma_{v \text{ in } c}\, \delta_v \;=\; \Sigma_{e \text{ in } c}\, w_e - \Sigma_{v \text{ in } c}\, \delta_v \;=\; \Sigma_{(u,v) \text{ in } c}\, [w_{(u,v)} - \delta_v]$

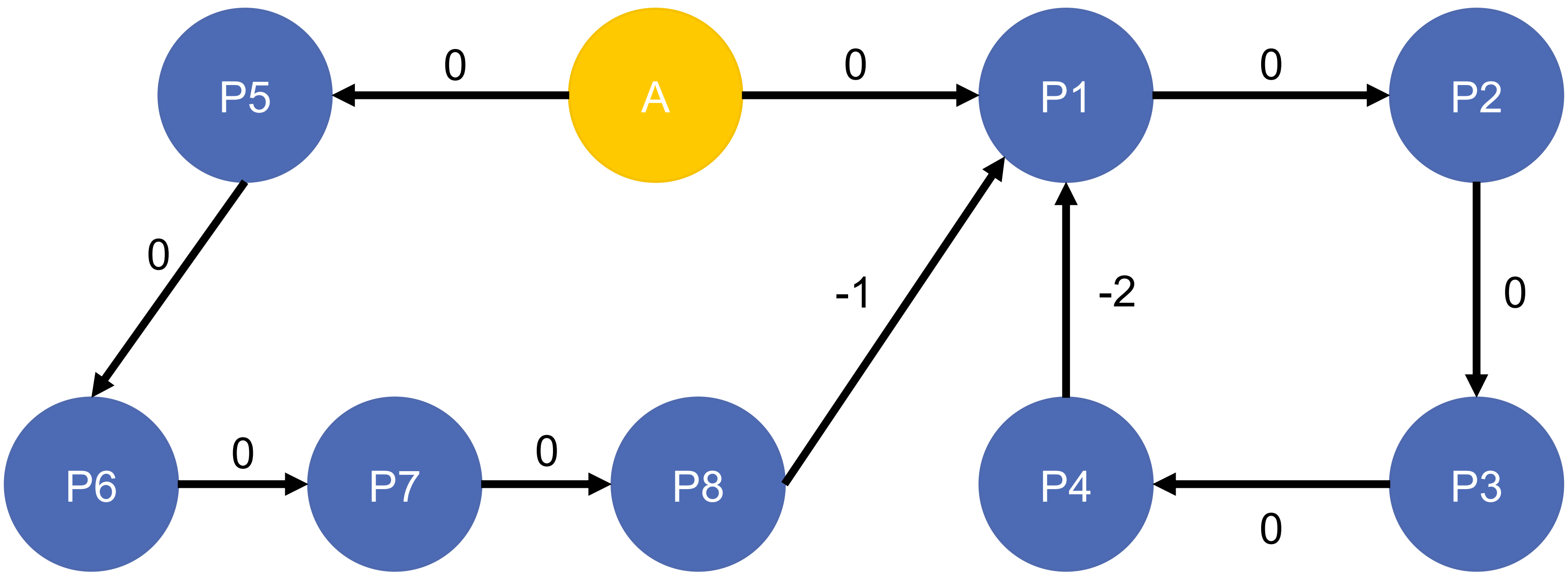
Take *G*, create *G'* s.t. all edges *e* = (*u*,*v*) are reweighted $r_{(u,v)} = \delta_v - w_{(u,v)}$

- Positive price cycles in *G* = negative weight cycles in *G'*

Bellman-Ford finds shortest paths

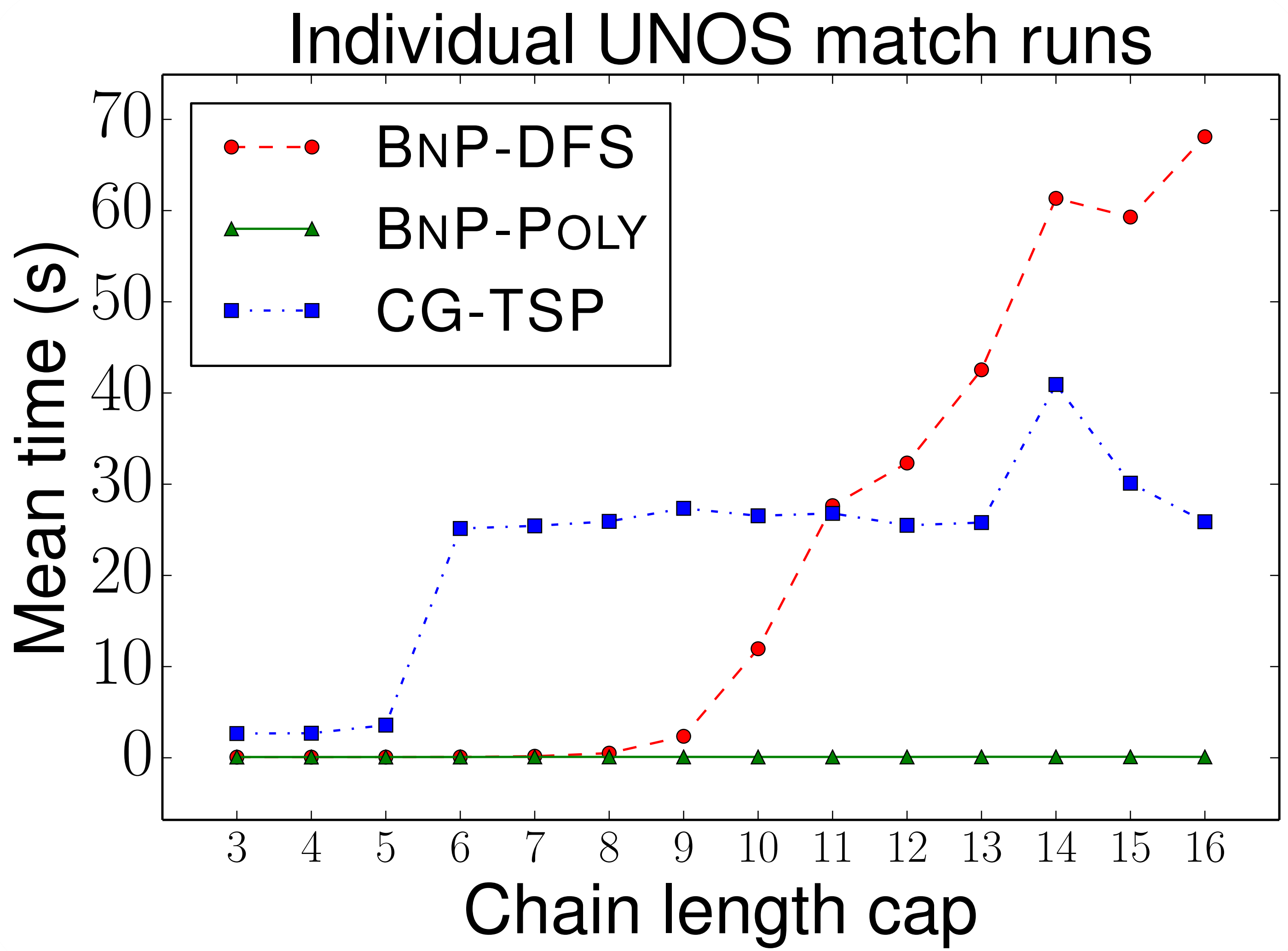
- Undefined in graphs with negative weight
- Adapt B-F to prevent internal looping during the traversal
  - *Shortest* path is NP-hard (reduce from Hamiltonian path:
    - Set edge weights to -1, given edge (*u*,*v*) in *E*, ask if shortest path from *u* to *v* is weight $1-|V|$ → visits each vertex exactly once
  - We only need *some* short path (or proof that no negative cycle exists)
- Now pricing runs in time $O(|V||E|\text{cap}^2)$

LOOP BLOCKING MUST BE DURING TRAVERSAL



(cycle cap = 3, chain cap = 6)

EXPERIMENTAL RESULTS



Note: Anderson et al.'s algorithm (CG-TSP) is *very strong* for uncapped aka "infinite-length" chains, but a chain cap is often imposed in practice