# APPLIED MECHANISM DESIGN FOR SOCIAL GOOD

JOHN P DICKERSON

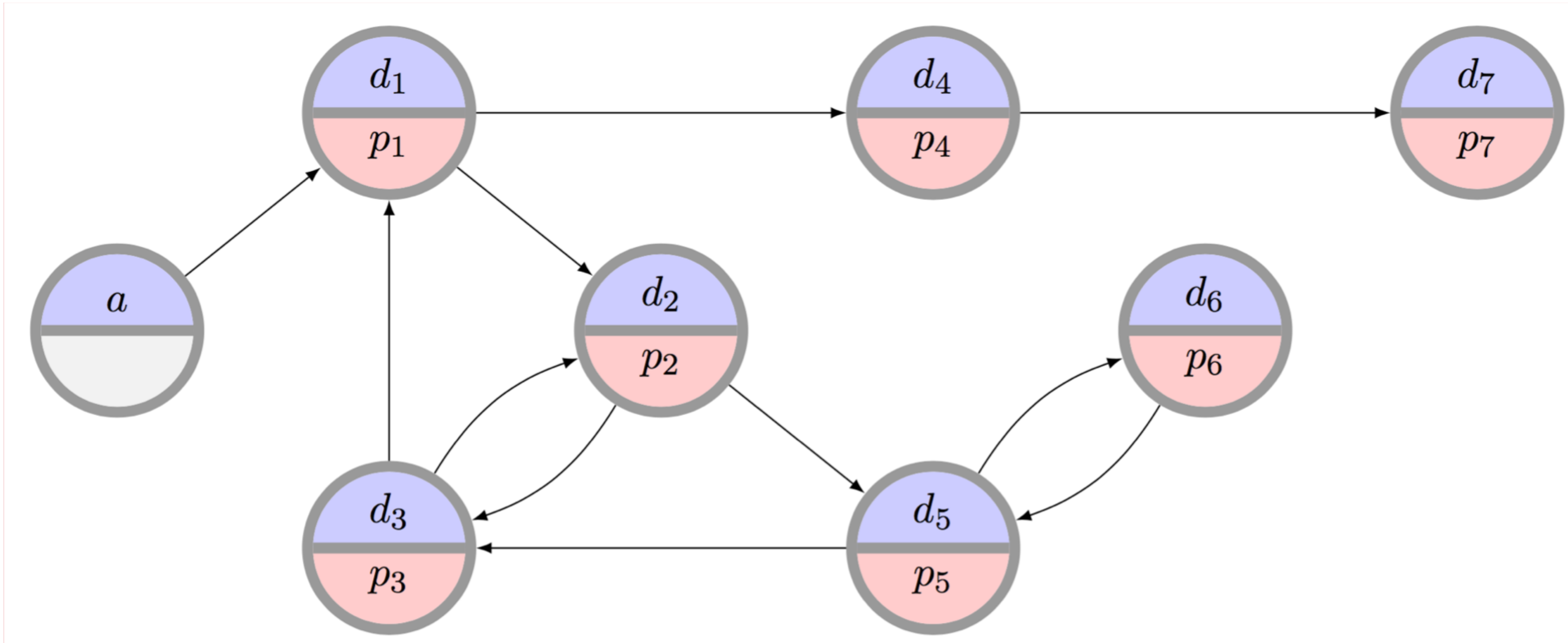
Lecture #14 – 03/13/2018

CMSC828M
Tuesdays & Thursdays
9:30am – 10:45am

COMPUTER SCIENCE
UNIVERSITY OF MARYLAND

# THIS CLASS:
# MANAGING SHORT-TERM UNCERTAINTY IN EXCHANGES (WITH SOME FAIRNESS)

# THE CLEARING PROBLEM




**The clearing problem is to find the "best" disjoint set of cycles of length at most *L*, and chains (maybe with a cap *K*)**

- Last class: only considered static deterministic matching
- This class: matching under short-term uncertainty
- Next class: general long-term dynamic matching over time

# MATCHED ≠ TRANSPLANTED

**Only around 10% of UNOS matches resulted 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

# 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***

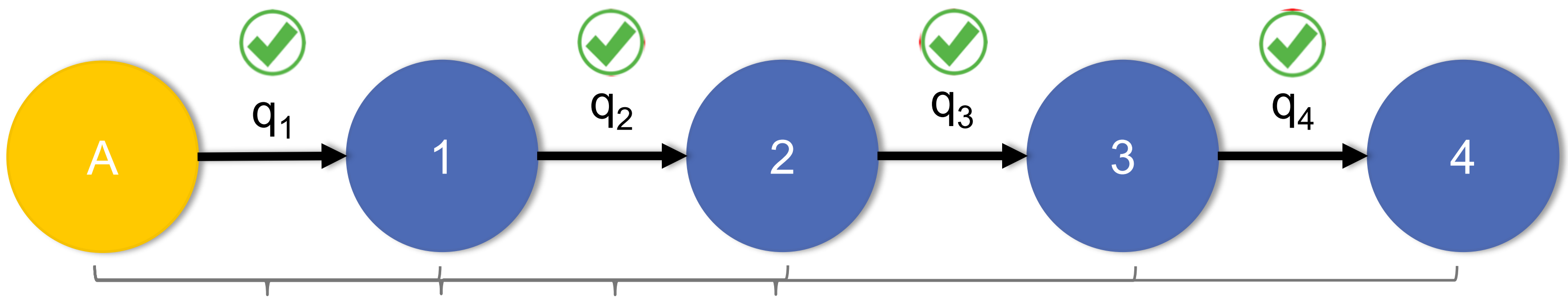
$$u(c) = \sum w_e \cdot \prod q_e$$

Value of successful cycle

Probability of success

# FAILURE-AWARE MODEL

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




$1q_1(1-q_2) + 2q_1q_2(1-q_3) + 3q_1q_2q_3(1-q_4) + 4q_1q_2q_3q_4$

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

Exactly first *i* transplants

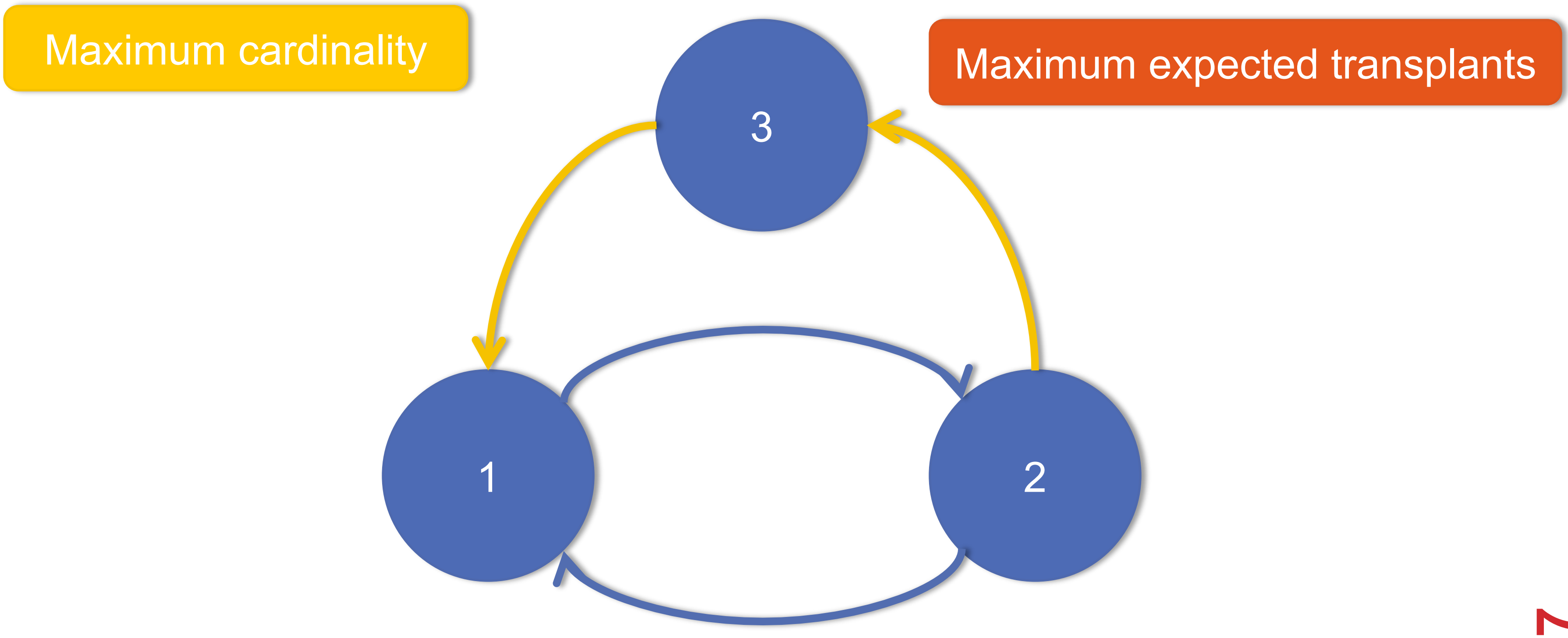
Chain executes in entirety

**Cannot simply “reweight by failure probability”**

# OUR PROBLEM

***Discounted clearing problem* is to find matching $M^*$ with highest discounted utility**

- Utility of a match *M*: $u(M) = \sum u(c)$



# "IN THE LARGE"

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

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

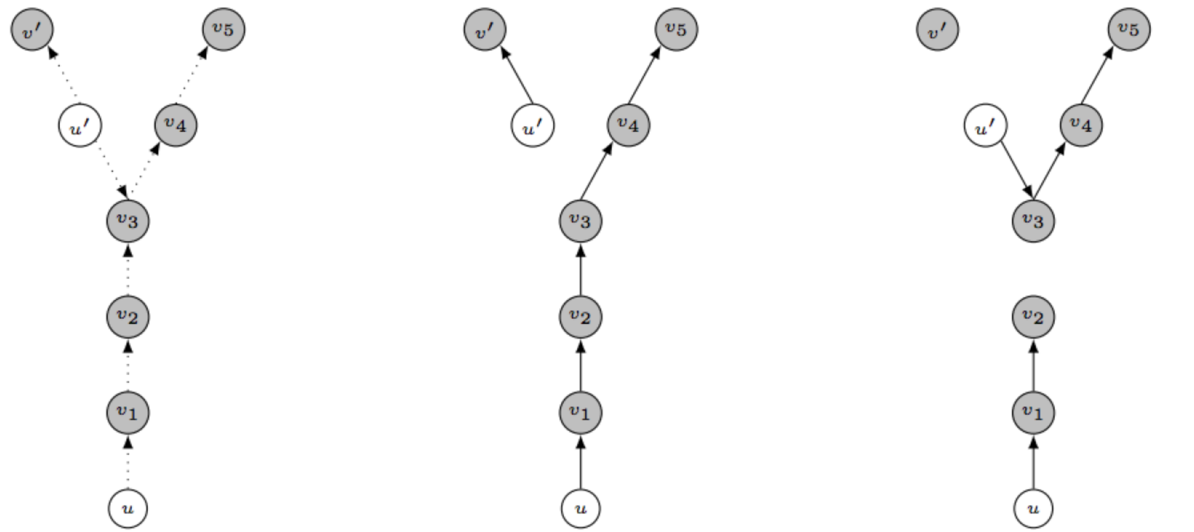
**Constant transplant success probability q**

THEOREM

For all $q \in (0,1)$ and $\alpha$, $\beta > 0$, 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 *Ω*(*n*) structures isomorphic to *X and isolated from the rest of the graph***

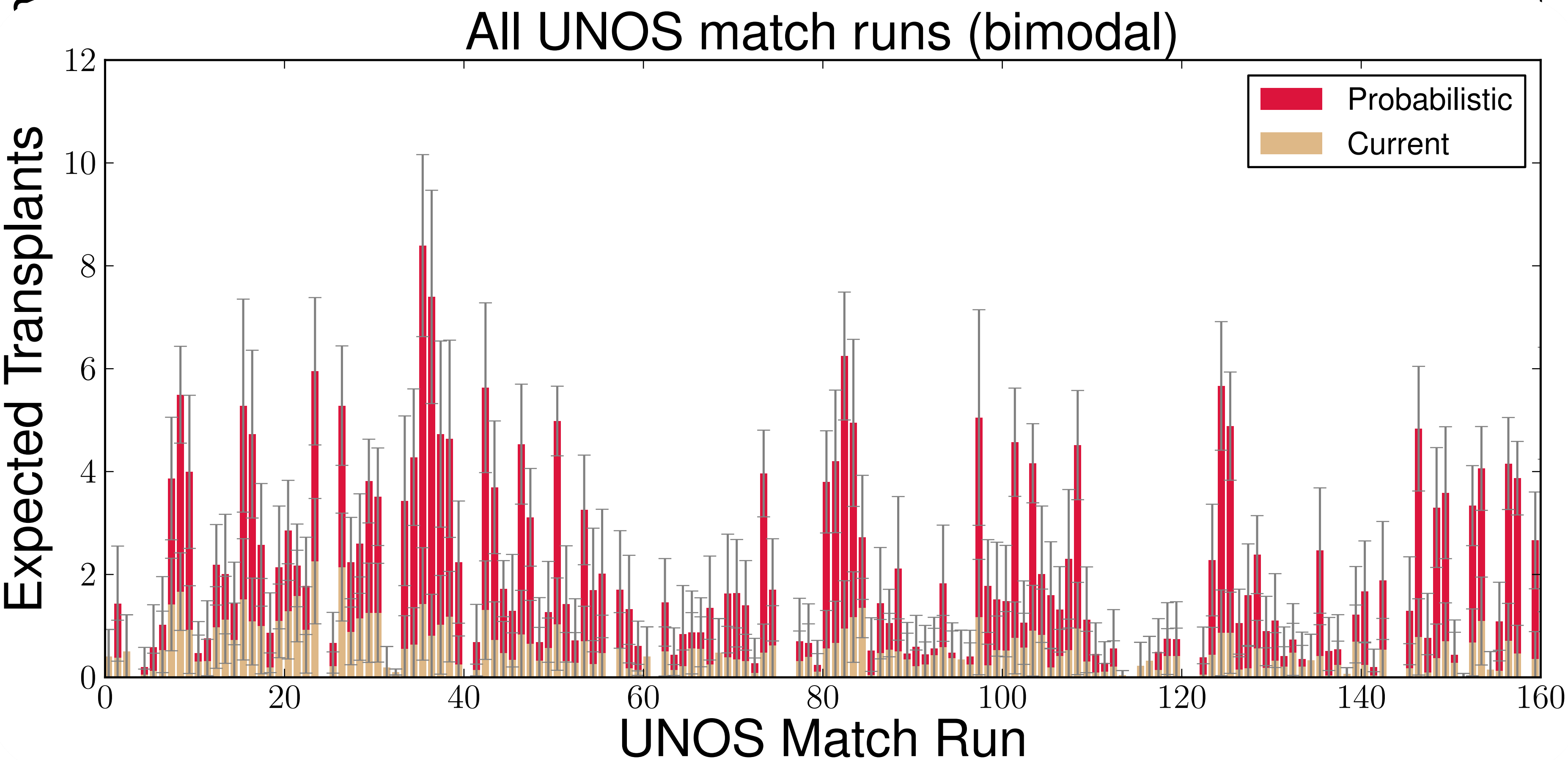
**Label them (alt vs. pair): flip weighted coins, constant fraction are labeled correctly → constant × *Ω*(*n*) = *Ω*(*n*)**

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

**In theory, we're losing out on *expected actual transplants* by maximizing match cardinality.**

**… What about in practice?**

All UNOS match runs (constant)
3.0
Probabilistic
Current

All UNOS match runs (bimodal)
Expected Transplants
0
2
4
6
8
10
12
Probabilistic
Current
0
20
40
60
80
100
120
140
160
UNOS Match Run

# SOLVING THIS NEW PROBLEM

**Real-world kidney exchanges are still small**

- Mostly around 300 donors and 300 patients

THEOREM

The restricted **discounted** maximum cycle cover problem is NP-hard.

***Un*discounted clearing problem is NP-hard when cycle/chain cap $L \geq 3$ (reduction from 3D-matching last class)**

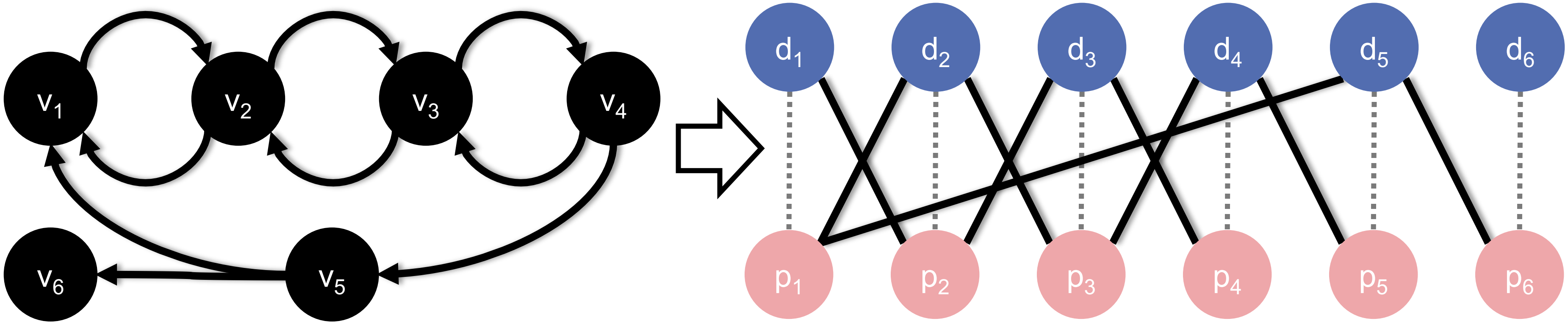
- Special case of our problem
- (Set success rate q = 1 for all edges)

# WE CAN'T USE THE CURRENT SOLVER

**Branch-and-bound IP solvers use upper and lower bounds to prune subtrees during search**

**Upper bound: cycle cover with no length cap**

- (Last class: PTIME through max weighted perfect matching)




**But now it is better to use shorter cycles instead of longer cycles to cover the same set of vertices …**

# WE CAN'T USE THE CURRENT SOLVER

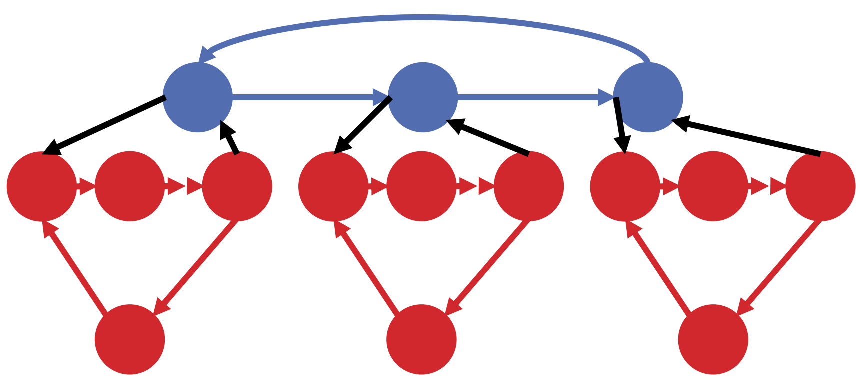
B&B upper bound

THEOREM

The unrestricted **discounted** maximum cycle cover problem is NP-hard.

**Reduce from 3D-matching, like last class. Intuition:**

- **3-cycles are better than L-cycles, for L>3**
- **Want the top (blue) vertices matched in 3-cycles, not L-cycles**
- **We showed this happens iff there is a 3D-matching**



# WE CAN USE PARTS OF THE CURRENT SOLVER

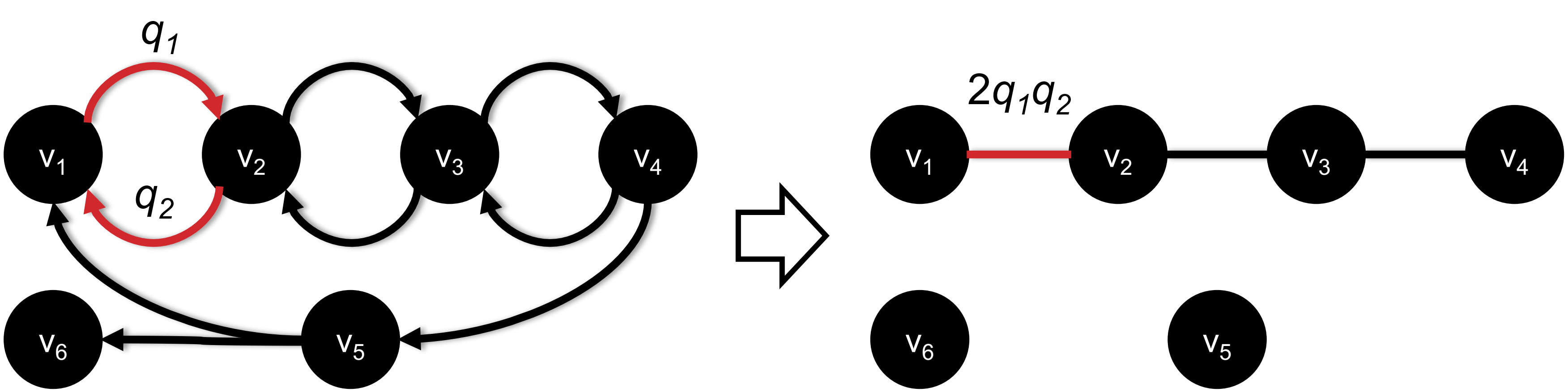
B&B lower bound

THEOREM

The restricted **discounted** maximum cycle cover problem is solvable in PTIME for *L*=2.

**For all 2-cyles between *u* and *v* in the original graph, set corresponding undirected edge weight in translated graph to:**

- $w_{e'} = q_{(u,v)} \cdot q_{(v,u)} \cdot ( w_{(u,v)} + w_{(v,u)} )$



# PRICING: 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

# SCALABILITY EXPERIMENTS

| \|V\| | CPLEX | Ours | Ours without chain curtailing |
|---|---|---|---|
| 10 | 127 / 128 | 128 / 128 | 128 / 128 |
| 25 | 125 / 128 | 128 / 128 | 128 / 128 |
| 50 | 105 / 128 | 128 / 128 | 125 / 128 |
| 75 | 91 / 128 | 126 / 128 | 123 / 128 |
| 100 | 1 / 128 | 121 / 128 | 121 / 128 |
| 150 | | 114 / 128 | 95 / 128 |
| 200 | | 113 / 128 | 76 / 128 |
| 250 | | 94 / 128 | 48 / 128 |
| 500 | | 107 / 128 | 1 / 128 |
| 700 | | 115 / 128 | |
| 900 | | 38 / 128 | |
| 1000 | | | |

- Runtime limited to 60 minutes; each instance given 8GB of RAM.
- |V| represents #patient-donor pairs; additionally, 0.1|V| altruistic donors are present.

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

**… But can this hurt some *individuals*?**

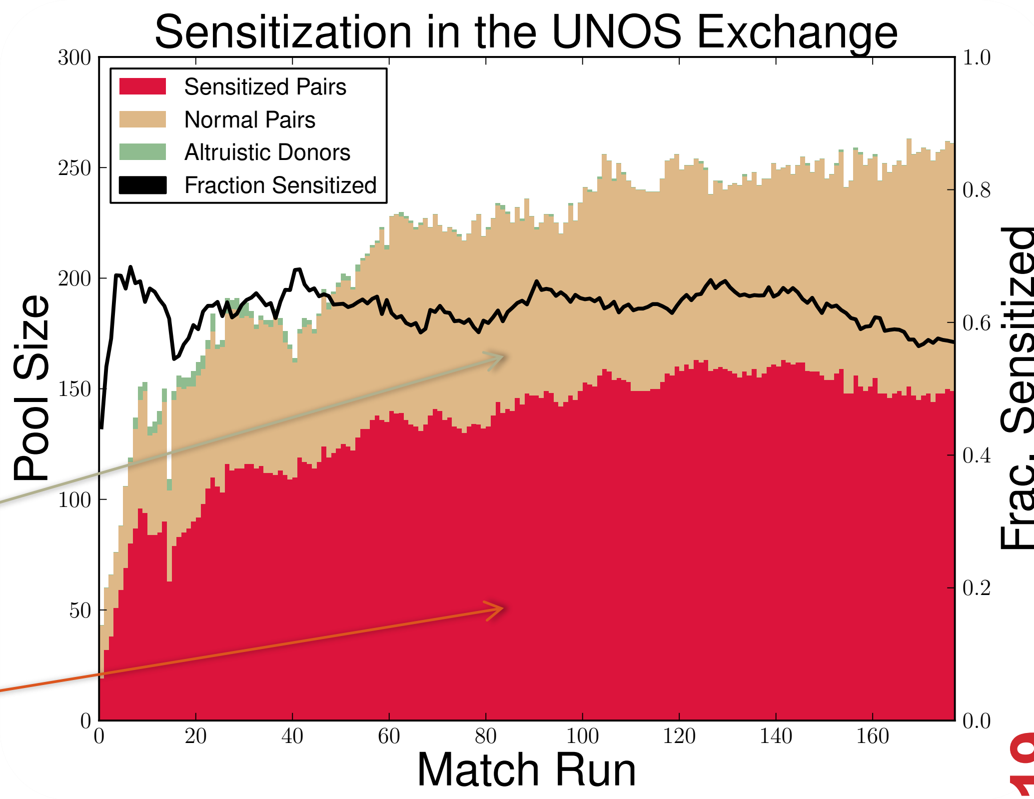
# SENSITIZATION AT UNOS

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

- Deceased donor waitlist: 15%-20%
- Kidney exchanges: **much** higher (60%+)

“Easy to match” patients

“Hard to match” patients



# RECALL:
# 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]

# PRICE OF FAIRNESS IN KIDNEY EXCHANGE

**Want a matching $M^*$ that maximizes utility function $u\colon \mathcal{M} \rightarrow \mathbb{R}$**

$$M^* = \underset{M \in \mathcal{M}}{\operatorname{argmax}}\, u(M)$$

**Price of fairness**: relative loss of match **efficiency** due to **fair** utility function $u_f$:

$$POF(\mathcal{M}, u_f) = \frac{u(M^*) - u(M_f^*)}{u(M^*)}$$

# FROM THEORY TO PRACTICE

**Last class, we saw that the price of fairness is low in theory**

$$POF(\mathcal{M}, u_{H \succ L}) \leq {}^{2}/_{33}$$

**Fairness criterion: extremely strict.**

**Theoretical assumptions (standard):**

- Big, dense graphs (“*n* → ∞”)
- Cycles (no chains)
- No post-match failures
- Simplified patient-donor features

**What about the price of fairness *in practice*?**

# TOWARD USABLE FAIRNESS RULES

**In healthcare, important to work within (or near to) the constraints of the fielded system**

- [Bertsimas, Farias, Trichakis 2013]
- Experience working with UNOS

**We now present two (simple, intuitive) rules:**

- **Lexicographic**: strict ordering over vertex types
- **Weighted**: implementation of "priority points"

# LEXICOGRAPHIC FAIRNESS

Find the best match that includes at least $\alpha$ fraction of highly-sensitized patients

**Matching-wide constraint:**

- Present-day branch-and-price IP solvers rely on an "easy" way to solve the pricing problem
- Lexicographic constraints → pricing problem requires an IP solve, too!

**Strong guarantee on match composition …**

- … but harder to predict effect on economic efficiency

# WEIGHTED FAIRNESS

Value matching a highly-sensitized patient at $(1+\beta)$ that of a lowly-sensitized patient, $\beta>0$

**Re-weighting is a preprocess →**

**works with all present-day exchange solvers**

**Difficult to find a "good" β?**

- Empirical exploration helps strike a balance

# THEORY VS. "PRACTICE"

*LEXICOGRAPHIC FAIRNESS*

# PRICE OF FAIRNESS: GENERATED DATA

| Size | Saidman (US) | Saidman (UNOS) | Heterogeneous |
|---|---|---|---|
| 10 | 0.24% (1.98%) | 0.00% (0.00%) | 0.98% (5.27%) |
| 25 | 0.58% (1.90%) | 0.19% (1.75%) | 0.00% (0.00%) |
| 50 | 1.18% (2.34%) | 1.96% (6.69%) | 0.00% (0.00%) |
| 100 | 1.46% (1.80%) | 1.66% (3.64%) | 0.00% (0.00%) |
| 150 | 1.20% (1.86%) | 2.04% (2.51%) | 0.00% (0.00%) |
| 200 | 1.43% (2.08%) | 1.55% (1.79%) | 0.00% (0.00%) |
| 250 | 0.80% (1.24%) | 1.86% (1.63%) | 0.00% (0.00%) |
| 500 | 0.72% (0.74%) | 1.67% (0.82%) | 0.00% (0.00%) |

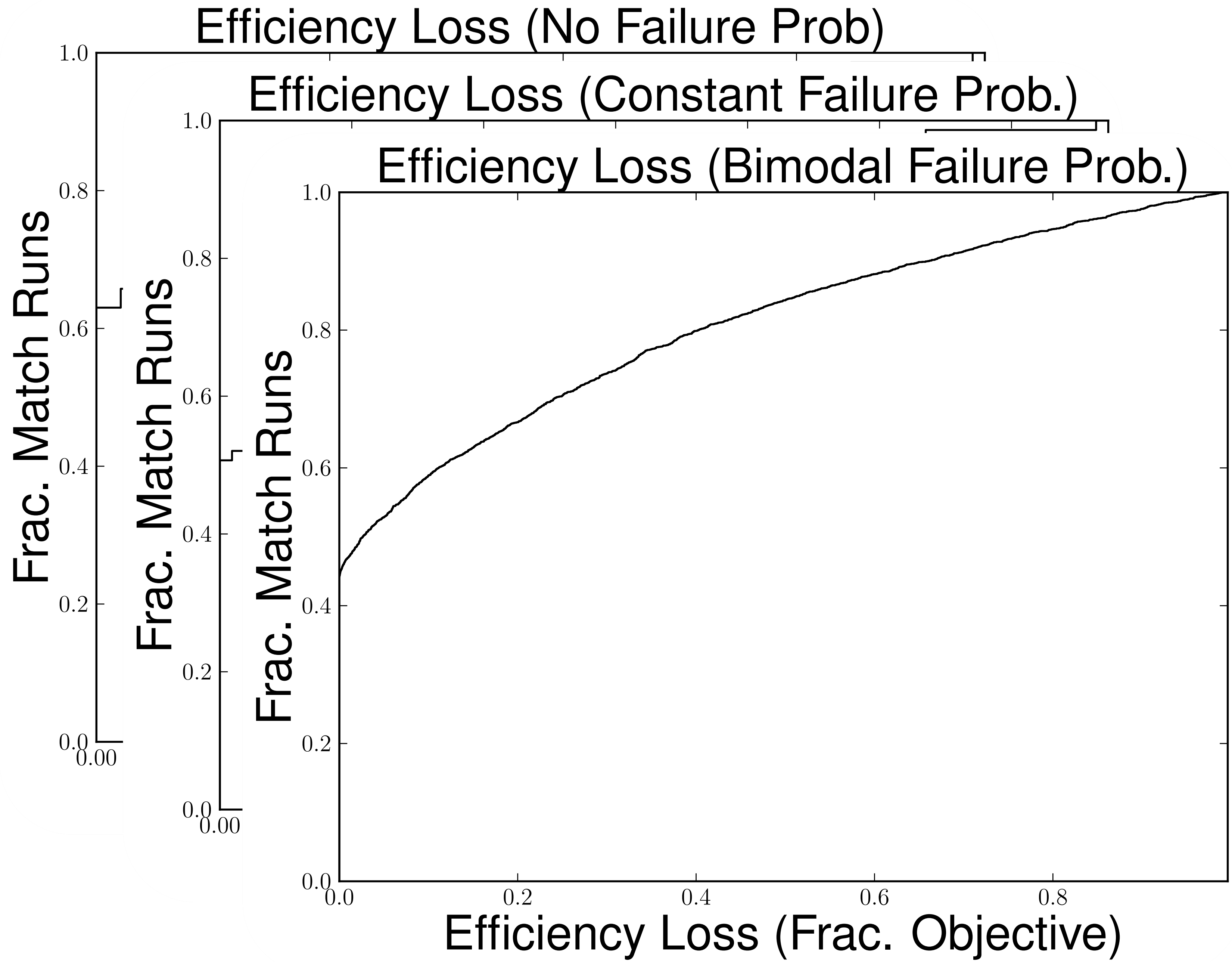
**Average (st.dev.) % loss in efficiency for three families of random graphs, under the strict lexicographic rule.**

**Good: aligns with the theory**

**Bad: standard generated models aren't realistic**

# UNOS RUNS

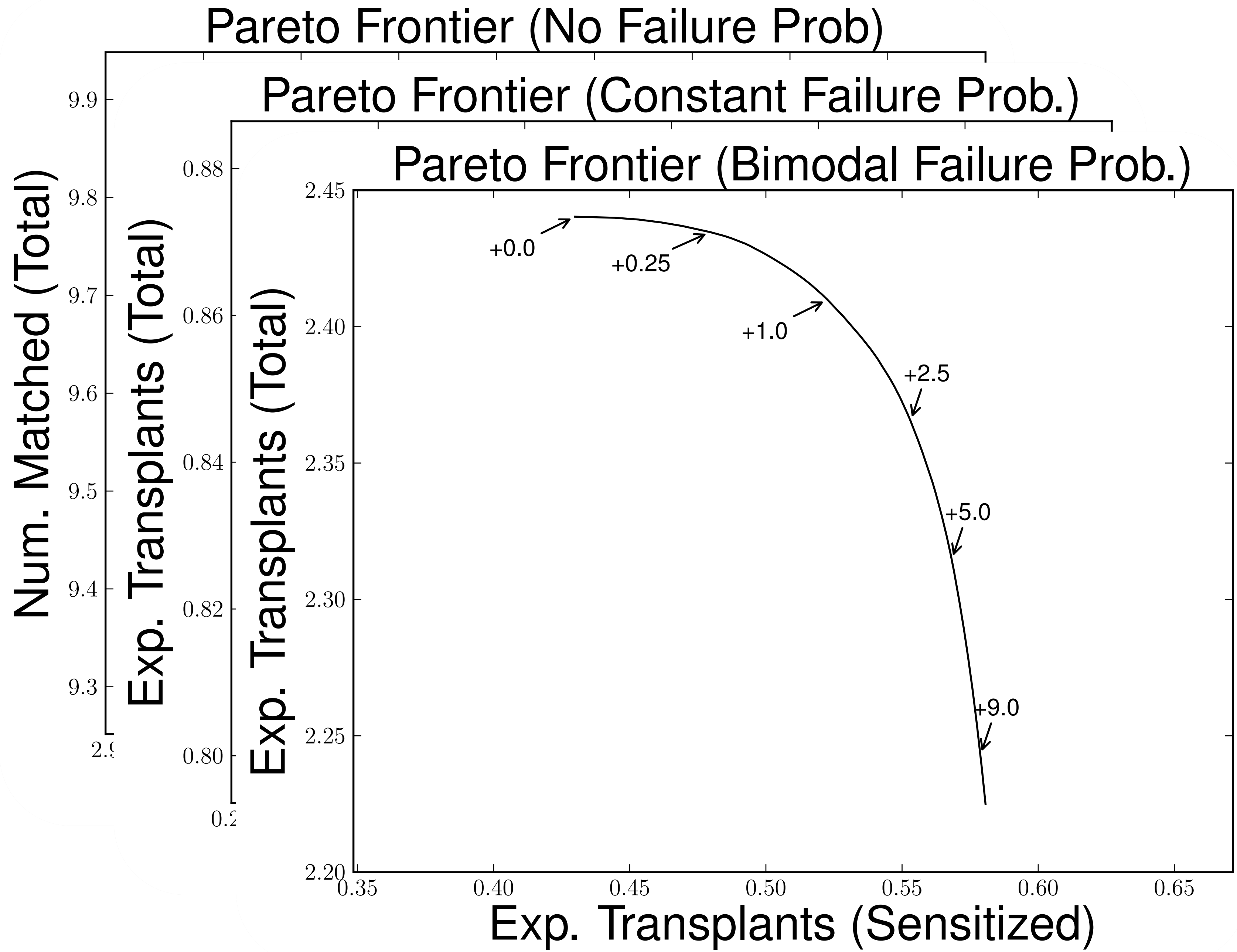
## *LEXICOGRAPHIC FAIRNESS, VARYING FAILURE RATES*



Efficiency Loss (No Failure Prob)
Efficiency Loss (Constant Failure Prob.)
Efficiency Loss (Bimodal Failure Prob.)
Frac. Match Runs
Frac. Match Runs
Frac. Match Runs
1.0
0.8
0.6
0.4
0.2
0.0
0.00
0.0
0.2
0.4
0.6
0.8
Efficiency Loss (Frac. Objective)

# UNOS RUNS

## *WEIGHTED FAIRNESS, VARYING FAILURE RATES*

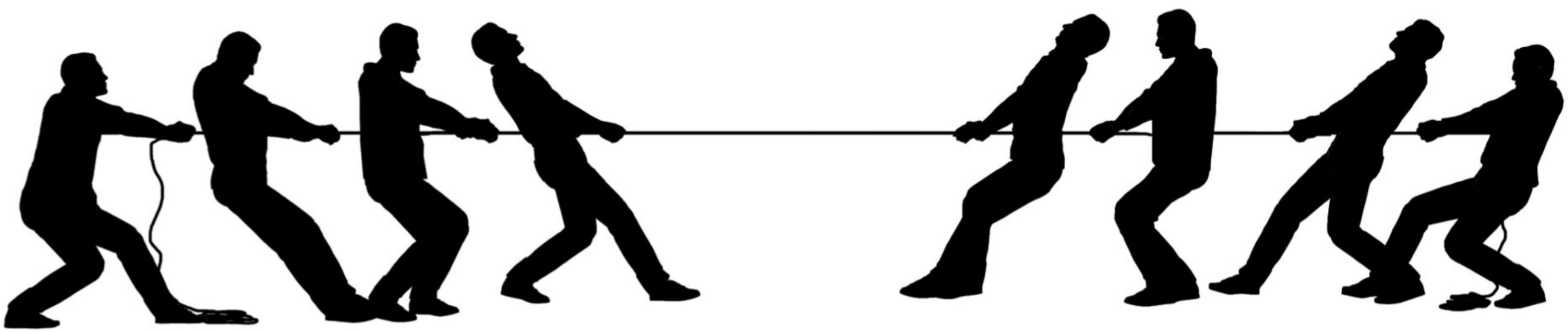

Pareto Frontier (No Failure Prob)
Num. Matched (Total)
9.9
9.8
9.7
9.6
9.5
9.4
9.3
Pareto Frontier (Constant Failure Prob.)
Exp. Transplants (Total)
0.88
0.86
0.84
0.82
0.80
Pareto Frontier (Bimodal Failure Prob.)
Exp. Transplants (Total)
2.45
2.40
2.35
2.30
2.25
2.20
+0.0
+0.25
+1.0
+2.5
+5.0
+9.0
0.35
0.40
0.45
0.50
0.55
0.60
0.65
Exp. Transplants (Sensitized)

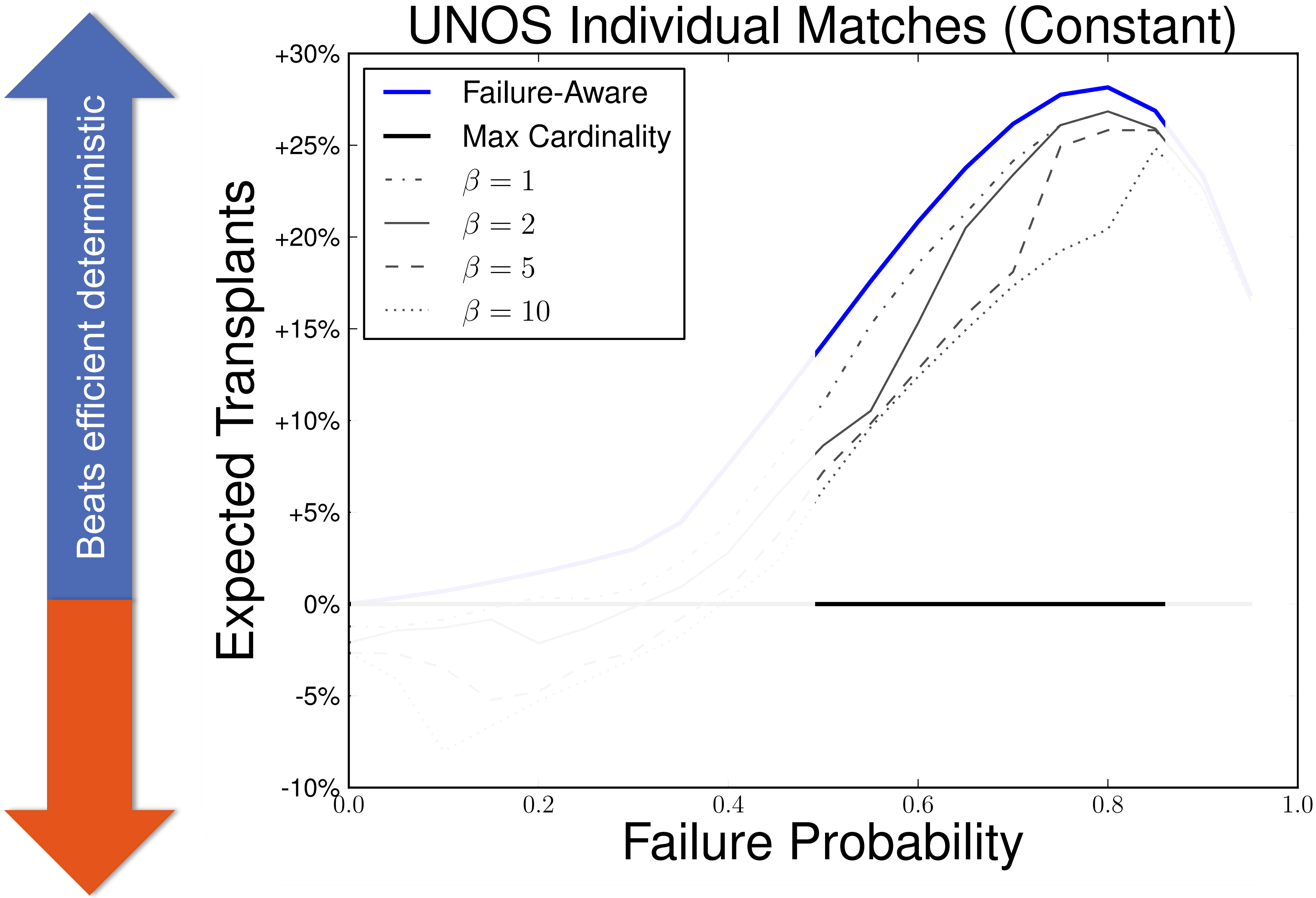
# CONTRADICTORY GOALS

**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 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)**

**Can we be more proactive in this balancing act?**

# 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 (or set packing) 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

# GENERAL THEORETICAL RESULTS

**Adaptive**: select one edge per vertex per *round*, test, repeat

THEOREM

**Stochastic matching:**
$(1-\varepsilon)$ approximation with $O_\varepsilon(1)$ queries per vertex, in $O_\varepsilon(1)$ rounds

**Stochastic k-set packing:**
$(2/k - \varepsilon)$ approximation with $O_\varepsilon(1)$ queries per vertex, in $O_\varepsilon(1)$ rounds

**Non-adaptive**: select O(1) edges per vertex, test all at once

THEOREM

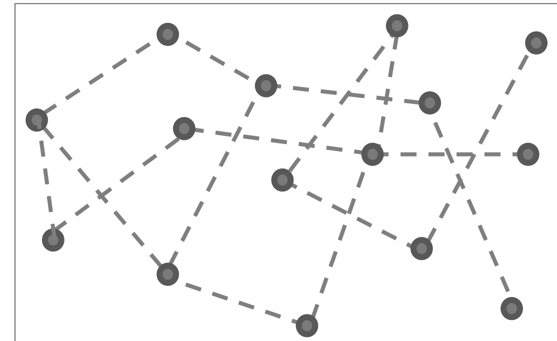
**Stochastic matching:**
$(0.5-\varepsilon)$ approximation with $O_\varepsilon(1)$ queries per vertex, in 1 round

**Stochastic k-set packing:**
$(2/k - \varepsilon)^2$ approximation with $O_\varepsilon(1)$ queries per vertex, in 1 round

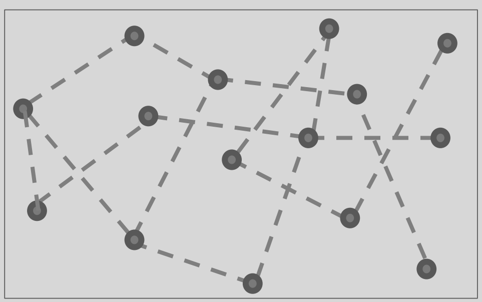
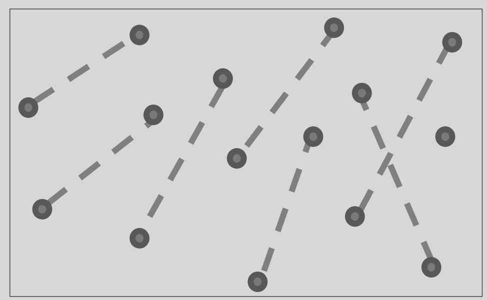
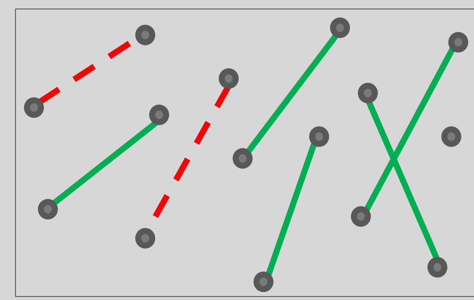
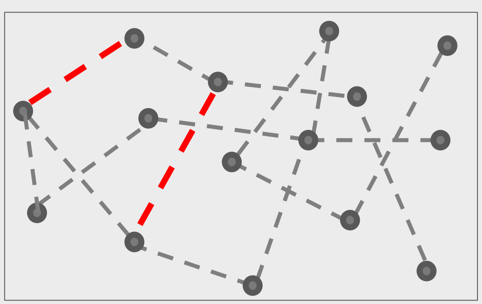
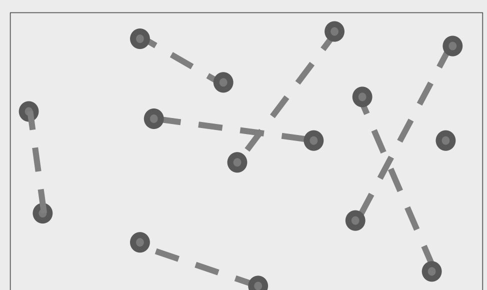
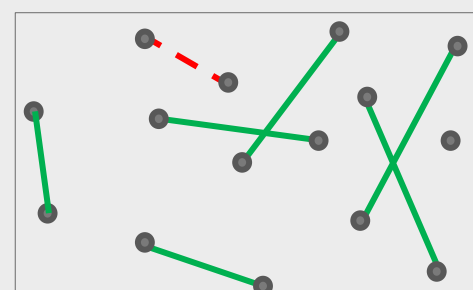
# ADAPTIVE ALGORITHM

For *R* rounds, do:

1. Pick a max-cardinality matching *M* in graph *G*, minus already-queried edges that do not exist
2. Query all edges in *M*



*Input Graph*

| *r* | Base graph | Matching picked | Result of queries |
|---|---|---|---|
| 1: |  |  |  |
| 2: |  |  |  |

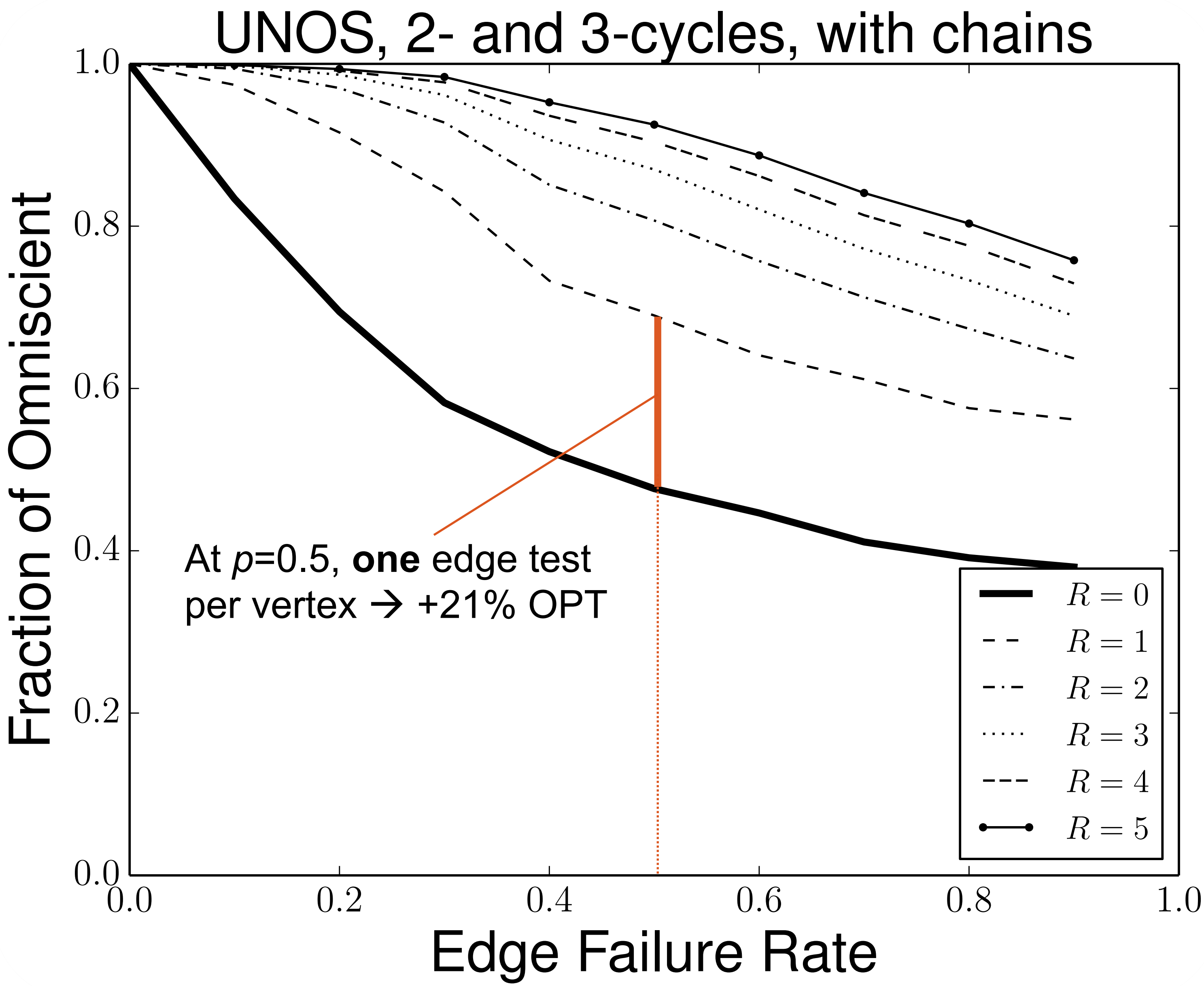
# INTUITION FOR ADAPTIVE ALGORITHM

**If at any round *r*, the best solution on edges queried so far is small relative to omniscient …**

- ... then current structrure admits *large* number of unqueried, disjoint augmenting structures
- For *k*=2, aka normal matching, simply augmenting paths

**Augmenting structures might not exist, but can query in parallel in a single round**

- Structures are constant size → exist with constant probability
- Structures are disjoint → queries are independent
- → Close a constant gap per round




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

# NEXT CLASS:
# DYNAMIC OPTIMIZATION