Stable Matching, Friendship, and Altruism

Elliot Anshelevich

Rensselaer Polytechnic Institute (RPI), Troy, New York

Joint work with: Onkar Bhardwaj (RPI), Sanmay Das (WashU), Martin Hoefer (MPI), Yonatan Naamad (Princeton)

Stable Matching

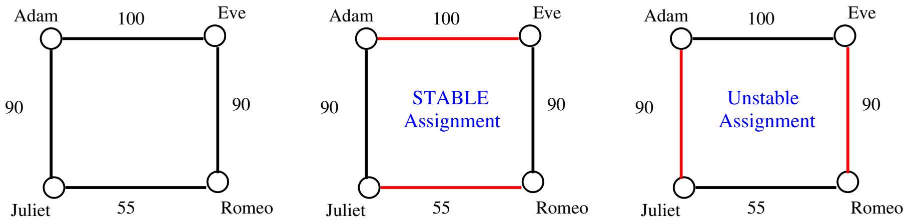
- Also known as “stable marriage"
- Classic game theory and algorithmic problem
- Applications: residents and hospitals, students and schools, kidney matching, ...

Motivation

- Stable matching with cardinal utilities
- Students told to choose project partners for class
- Stable partner assignment:
  - No two students should want to leave their present partner and be partners with each other
    - (at least one of them should be unwilling)

Motivation

- Stable matching with cardinal utilities
- Students told to choose project partners for class
- Stable partner assignment:
  - No two students should want to leave their present partner and be partners with each other
    - (at least one of them should be unwilling)



Stable Matching with Cardinal Utilities

Model

- Undirected graph, weights on the edges (denoted r_{uv})
- Nodes told to choose their partners
  - u, v partners then both get reward r_{uv}
  - No partner then 0 reward

Stability

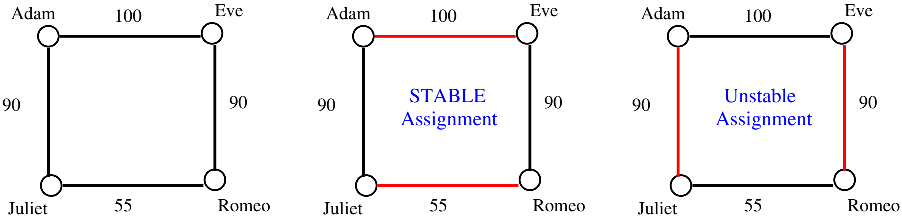
No “blocking pair"

- (x, y) a blocking pair if x prefers y over its current partner and vice versa.
- For now, higher preference = more reward

I. First Goal of this Talk

Understand some basic properties of this "nice" stable matching model

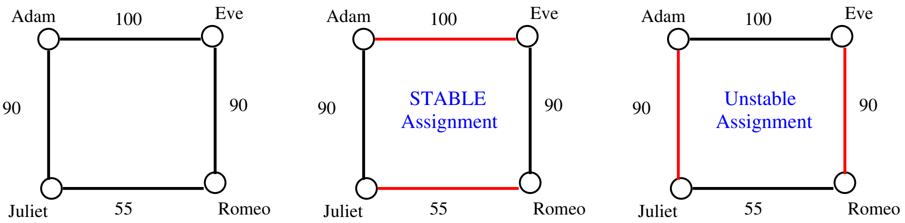
- Does a stable matching exist?
- What is the quality of stable matchings?
- Can we improve their quality?



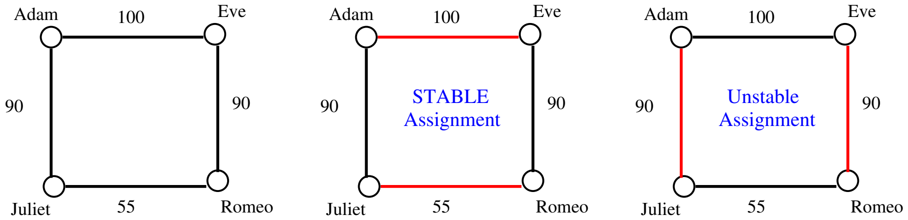
I. First Goal of this Talk

Understand some basic properties of this "nice" stable matching model

- Does a stable matching exist?
  - Yes: Greedy matchings are stable.
- What is the quality of stable matchings?
- Can we improve their quality?



Quality of Stable Matchings



Quality of a matching

- Value of matching: $v(M) = \sum_{(uv) \in M} r_{uv}$
- Price of Anarchy: PoA $= \frac{v(M^{OPT})}{v(M_{worst,stable})}$
- Price of Stability: PoS $= \frac{v(M^{OPT})}{v(M_{best,stable})}$

I. First Goal of this Talk

Understand some basic properties of this "nice" stable matching model

- Does a stable matching exist?
  - Yes: Greedy matchings are stable.
- What is the quality of stable matchings?
- Can we improve their quality?

Bounds on PoA, PoS

- $v(M) = \sum_{(uv) \in M} r_{uv}$
- Price of Anarchy $= \frac{v(M^{OPT})}{v(M_{worst,stable})}$
- Price of Stability $= \frac{v(M^{OPT})}{v(M_{best,stable})}$

I. First Goal of this Talk

Understand some basic properties of this "nice" stable matching model

- Does a stable matching exist?
  - Yes: Greedy matchings are stable.
  - In fact, stable matchings are exactly the greedy matchings.
- What is the quality of stable matchings?
- Can we improve their quality?

Bounds on PoA, PoS

- $v(M) = \sum_{(uv)\in M} r_{uv}$
- Price of Anarchy $= \frac{v(M^{OPT})}{v(M_{worst,stable})}$
- Price of Stability $= \frac{v(M^{OPT})}{v(M_{best,stable})}$

I. First Goal of this Talk

Understand some basic properties of this "nice" stable matching model

- Does a stable matching exist?
  - Yes: Greedy matchings are stable.
  - In fact, stable matchings are exactly the greedy matchings.
- What is the quality of stable matchings?
- Can we improve their quality?

Bounds on PoA, PoS

- PoA, PoS ≤ 2
- Tight Bounds

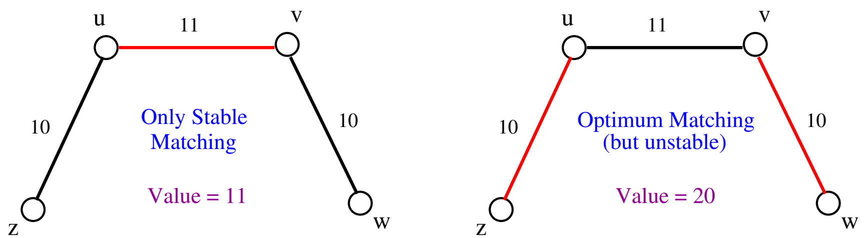
- $v(M) = \sum_{(uv)\in M} r_{uv}$
- Price of Anarchy $= \frac{v(M^{OPT})}{v(M_{worst,stable})}$
- Price of Stability $= \frac{v(M^{OPT})}{v(M_{best,stable})}$

Quality of Stable Matchings

Bounds on PoA, PoS

- PoA, PoS ≤ 2
- Tight Bounds

- $v(M) = \sum_{(uv) \in M} r_{uv}$
- Price of Anarchy $= \frac{v(M^{OPT})}{v(M_{worst,stable})}$
- Price of Stability $= \frac{v(M^{OPT})}{v(M_{best,stable})}$



II. Main topic of this talk: Friendship and Altruism

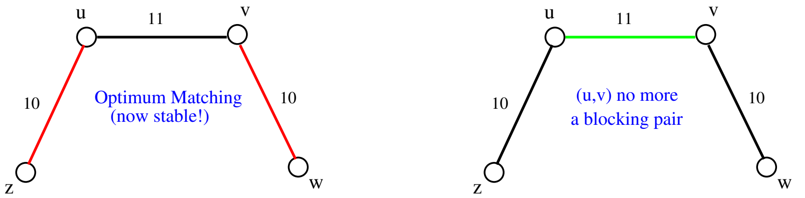
- What if nodes do not care about only their own reward?
- They care about well-being of their friends (to some extent)
- Does it improve the quality of stable matchings?

II. Main topic of this talk: Friendship and Altruism

- What if nodes do not care about only their own reward?
- They care about well-being of their friends (to some extent)
- Does it improve the quality of stable matchings?

Example

- Suppose the utility (or happiness) of nodes also counts the reward of their neighbors.



Utility Definition

More formally,

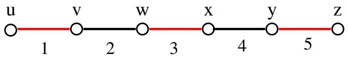
- Utility: $U(u) = R(u) + \sum_{v \neq u} \alpha_{d(u,v)} \cdot R(v)$
- A node cares α_k about well-being of nodes k-hops away
- $1 \geq \alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_{diam(G)} \geq 0$
  $\Rightarrow$ More distance means less care

Utility Definition

More formally,

- Utility: $U(u) = R(u) + \sum_{v \neq u} \alpha_{d(u,v)} \cdot R(v)$
- A node cares α_k about well-being of nodes k-hops away
- $1 \geq \alpha_1 \geq \alpha_2 \geq \cdots \geq \alpha_{diam(G)} \geq 0$
  $\Rightarrow$ More distance means less care

Example utility calculation



Suppose $\alpha_{d(u,v)} = \frac{1}{d(u,v)}$, then:

$$
\begin{aligned}
U(u) &= R(u) + R(v) + \frac{R(w)}{2} + \frac{R(x)}{3} + \frac{R(y)}{4} + \frac{R(z)}{5} \\
&= 1 + 1 + \frac{3}{2} + \frac{3}{3} + \frac{5}{4} + \frac{5}{5}
\end{aligned}
$$

Stable Matching with Friendship or Altruism

Recall, PoS $= \frac{v(M^{OPT})}{v(M_{best,stable})}$ and PoA $= \frac{v(M^{OPT})}{v(M_{worst,stable})}$

- Stable Matching still exists
- Price of Anarchy still at most 2

Stable Matching with Friendship or Altruism

Recall, PoS $= \frac{v(M^{OPT})}{v(M_{best,stable})}$ and PoA $= \frac{v(M^{OPT})}{v(M_{worst,stable})}$

- Stable Matching still exists
- Price of Anarchy still at most 2

Theorem

With Friendship, PoS $\leq \frac{2+2\alpha_1}{1+2\alpha_1+\alpha_2}$ (… a tight bound)

Bounds with Friendship

Theorem

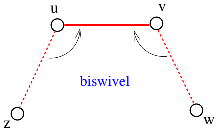
With Friendship, PoS $\leq \frac{2+2\alpha_1}{1+2\alpha_1+\alpha_2}$

Remarks:

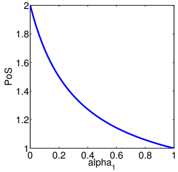
– Better than the bound of 2 without friendship

– Only α_1 and α_2 matter

– PoA stays the same



- $\alpha_1 = \alpha_2 = 1/2 \Rightarrow$ PoS ≤ 1.2
- A little friendship makes a large difference for PoS.



Proof Sketch of PoS Bound (1/3)

Algorithm

Start with matching $M =$ optimum matching

1. Select best blocking pair, say (u, v)
   – maximum r_{uv} among all blocking pairs
2. Make u, v partners (dropping their current partners)
3. Repeat

Proof Sketch of PoS Bound (1/3)

Algorithm

Start with matching $M =$ optimum matching

1. Select best blocking pair, say (u, v)
   – maximum r_{uv} among all blocking pairs
2. Make u, v partners (dropping their current partners)
3. Repeat

Without Friendship or Altruism

- Converges to stable matching in linear time
- Output: a stable matching within factor of 2 of optimal

Proof Sketch of PoS Bound (1/3)

Algorithm

Start with matching $M =$ optimum matching

1. Select best relaxed blocking pair, say (u, v)
   – maximum r_{uv} among all relaxed blocking pairs
2. Make u, v partners (dropping their current partners)
3. Repeat

Proof Sketch of PoS Bound (1/3)

Algorithm

Start with matching $M =$ optimum matching

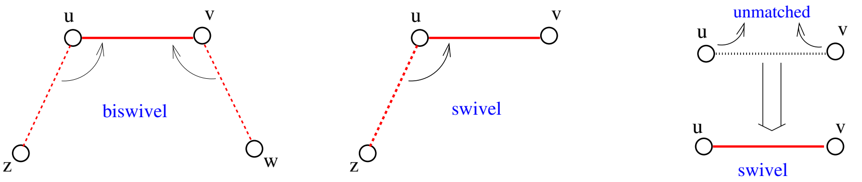
1. Select best relaxed blocking pair, say (u, v)
   – maximum r_{uv} among all relaxed blocking pairs
2. Make u, v partners (dropping their current partners)
3. Repeat

Will show

- Algorithm terminates after $O(m^2)$ iterations
- Output: a stable matching of high quality

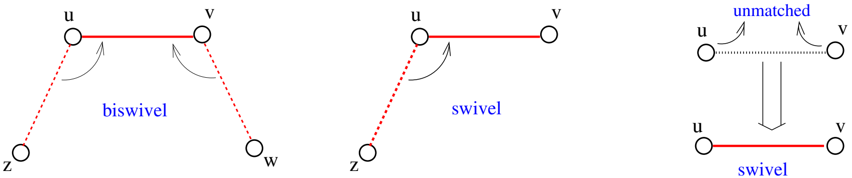
Proof Sketch of PoS Bound (2/3)

Types of blocking pairs

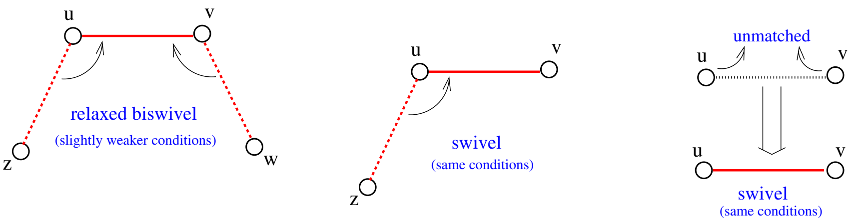


Proof Sketch of PoS Bound (2/3)

Types of blocking pairs



Relaxed blocking pairs



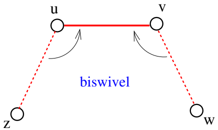
ignores (v, z) and (u, w) edges

Proof Sketch of PoS Bound (1/3)

Algorithm

Start with matching $M =$ optimum matching

1. Select best relaxed blocking pair, say (u, v) – maximum r_{uv} among all relaxed blocking pairs
2. Make u, v partners (dropping their current partners)
3. Repeat



Convergence

- Algorithm terminates after $O(m^2)$ iterations
- Output: a stable matching of high quality

Proof Sketch of PoS Bound (3/3)

Trace trajectories of edges under algorithm execution

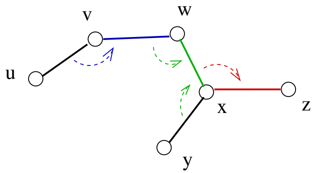


Figure: Trajectory $(uv) \rightarrow (vw) \rightarrow (wx) \rightarrow (xz)$

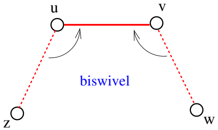
- *Quality of matching can decrease only when relaxed biswivel*
- **Cannot decrease indefinitely because the trajectory of an edge can have *at most one* relaxed-biswivel**
  – Conditions for relaxed biswivel bounds the decrease

Proof Sketch of PoS Bound

Algorithm

Start with matching $M =$ optimum matching

1. Select best relaxed blocking pair, say (u, v)
   – maximum r_{uv} among all relaxed blocking pairs
2. Make u, v partners (dropping their current partners)
3. Repeat



Convergence

- Algorithm terminates after $O(m^2)$ iterations
- Output: a stable matching within factor $\frac{2+2\alpha_1}{1+2\alpha_1+\alpha_2}$ of optimum

III. Extensions: Fractional Matching and Unequal Reward Sharing

Fractional Matching

- Each node has a budget of 1, divides it among incident edges
- Contribution Games (see A+Hoefer2012): nodes split effort among relationships they participate in
- Essentially all results mentioned above hold for fractional matching and contribution games

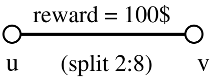
III. Extensions: Fractional Matching and Unequal Reward Sharing

Fractional Matching

- Each node has a budget of 1, divides it among incident edges
- Contribution Games (see A+Hoefer2012): nodes split effort among relationships they participate in
- Essentially all results mentioned above hold for fractional matching and contribution games

Unequal Reward Sharing

- More general scenario:
  – reward from an edge shared unequally



Unequal Reward Sharing Without Friendship

Without Friendship (all *alpha*'s zero, no concern for others rewards):

- Integral stable matching does not exist
- Fractional stable matching exists

Unequal Reward Sharing Without Friendship

Without Friendship (all *alpha*'s zero, no concern for others rewards):

- Integral stable matching does not exist
- Fractional stable matching exists

- Define $R = \max_{(uv)\in G} \frac{r_{uv}^{u}}{r_{uv}^{v}}$
  – maximum disparity in reward sharing
- Price of Anarchy is unbounded
  – PoA $\leq 1 + R$ (tight bound)
- Price of Stability is also unbounded!
  $\Rightarrow$ Stable matchings can be really bad!

Unequal Reward Sharing With Friendship

Existence of Stable Matching

(Fractional) stable matching exists
(Note: No longer a superset of stable matchings without friendship!)

Unequal Reward Sharing With Friendship

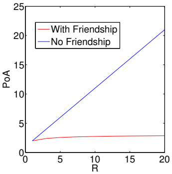
Existence of Stable Matching

(Fractional) stable matching exists
(Note: No longer a superset of stable matchings without friendship!)

Recall, PoS $= \frac{v(M^{OPT})}{v(M_{best,stable})}$ and PoA $= \frac{v(M^{OPT})}{v(M_{worst,stable})}$

Huge Reduction in Price of Anarchy

- PoA $\leq 1 + \frac{R+\alpha_1}{1+\alpha_1 R}$ (a tight bound)
- It was unbounded $(1 + R)$ without friendship
- For $\alpha = 1/2$, PoA ≤ 3 regardless of R!
  In general, at most $1 + \frac{1}{\alpha_1}$.



Conclusion

- Stable Matching (with cardinal utilities)
- Price of Anarchy, Price of Stability

Friendship and Altruism Help!

- Price of Stability improves for equal reward sharing
  – From 2 to $\frac{2+2\alpha_1}{1+2\alpha_1+\alpha_2}$
  – Good stable matching found in poly-time
- For unequal reward sharing, Price of Anarchy greatly improves
  – Now $1 + \frac{R+\alpha_1}{1+\alpha_1 R}$ from $1 + R$ – for example, from unbounded to a mere 3 when $\alpha_1 = 1/2$

Other Results

- Integral stable matchings exist for some interesting cases of reward sharing
  - Matthew Effect reward sharing (more reputation, more credit)
  - Trust reward sharing ("trustworthiness of node" also plays a role)
    – Here PoA $\leq \min\{2 + 2\alpha_1, 3\}$
- All the results can be extended to fractional matching, convex Contribution Games

Some Interesting Directions

- *k*-stable matching, fractional matching give rise to new questions
- What if Friendship and Collaboration networks are not the same?
- Different notions of altruism
- Coalitional stability
  – Strong Equilibrium may not exist with friendship
  – Fractional core always exists?
  – Other notions for hypergraph matching?

Thanks!