CSC200: Lecture 20

- Today:
  - Continue (basic) stable marriage problem
  - Some extensions of basic stable matching
  - kidney exchange; matching in a graph (next lecture)
  - We can view these problems as mechanism design without money
  - Note: Stable matching and kidney exchange discussion not in text

- After the break:
  - An additional lecture by Joanna Drummond on stable matching
  - The internet and search engines: Chapters 13,14
  - internet advertising; return to a matching market application: Ch.15
  - various network phenomena at a more aggregate (population) level (Ch.16-18), e.g., information cascades, power laws, tipping points, etc.
- Announcements
  - I may post one or two questions for Assignment 3 dealing with search engines but assignment not due until February 10.

Basic Stable Matching/Marriage Problem

- A *matching (marriage)* is a mapping μ that associates a women with each man and a man with each women
  - denote the partner of man *m* by $\mu(m) \in W$
  - denote the partner of woman *w* by $\mu(w) \in M$
  - we require that $\mu(m) = w$ iff $\mu(w) = m$
  - *Just a set of pairs (m,w) where each man, women is in exactly one pair*

- A pair *(m,w)* *blocks* matching μ if $w \succ_m \mu(m)$ and $m \succ_w \mu(w)$
  - Both *m* and *w* would prefer to be with each other than with the partners assigned to them by μ

- Matching is *stable* iff it is unblocked by any pair
  - In other words, while some man *m* might prefer a *different* partner *w*, his preferred partner *w* does *not* prefer *m* to her partner (and similarly for women)

Some Simple Examples

Man	1st	2nd	3rd
x	a	b	c
y	b	a	c
z	a	b	c

Woman	1st	2nd	3rd
a	y	x	z
b	x	y	z
c	x	y	z

Match 1
a-x
b-y
c-z

It is stable (x,y don't want to move; a, b won't move except to x,y; so c,z stuck)

Match 2
a-y
b-x
c-z

It is stable (a,b don't want to move; x, y won't move except to a,b; so c,z stuck)

Match 3
a-z
b-y
c-x

Unstable (b,x form a blocking pair)

(a,x also form a blocking pair)

Stability is an Equilibrium Concept

- A stable matching is analogous to a Nash equilibrium
  - The agents take moves (find partners) as in any standard game
  - If matching is stable, there is nobody who can *defect* an improve their "payoff" (think of ranking of your partner as your "payoff")
  - **Key difference:** "it takes two to tango". Nobody can *unilaterally* deviate, they have to have a *willing partner*. That's why we talk about blocking *pairs*: a match is unstable if a pair has a best response that breaks the match (i.e., they are a blocking pair).
  - This is a form of *coalitional stability* (since only coalitions of size two, one from each side) can change the matching
- Aside: some versions allow *individuals* to deviate (by staying "alone")
  - use *acceptability thresholds*: least preferred person one is willing to marry
  - would prefer to be "matched to yourself" than someone below threshold
    - e.g., $w: m1 \succ_w m2 \succ_w m3 \succ_w \mathbf{w} \succ_w m4 \succ_w m5$
    - if *w* matched to *m4*, she can unilaterally deviate (match to herself)

Gale-Shapley Algorithm

- How do we find stable matchings?
  - very clever, intuitive algorithm: the *Gale-Shapley* algorithm
  - also know as *deferred acceptance algorithm*
  - comes in two different varieties: male- and female-proposing
- *Female-proposing deferred acceptance (FPDA) algorithm*
  - in a sequence of rounds, certain women simultaneously propose to men; men accept proposals from their most preferred partners and pairs become “engaged”
  - women who are unengaged in any round make proposals, always to the most preferred man in their list that they haven’t yet proposed to
  - men who get a proposal(s) in any round:
    - if already engaged, stick with current partner ***if*** *s*he is preferred to all of his proposers; otherwise, break engagement and accept proposal of his most preferred proposer (old partner becomes unengaged)
    - if unengaged, accept proposal of his most preferred proposer

Illustration: Example 1

Wmn	1st	2nd	3rd	4th
a	x	y	z	w
b	y	x	w	x
c	x	z	y	w
d	y	w	x	z

A * means: already proposed to that man

Man	1st	2nd	3rd	4th
w	d	b	a	c
x	b	a	d	c
y	b	a	c	d
z	d	b	c	a

Round 1

Proposals:	New Engagements:
a: x	w: –
b: y	x: a
c: x	y: b
d: y	z: –

Illustration: Example 1

Wmn	1st	2nd	3rd	4th
a	x*	y	z	w
b	y*	x	w	x
c	x*	z	y	w
d	y*	w	x	z

A * means: already proposed to that man

Man	1st	2nd	3rd	4th
w	d	b	a	c
x	b	a	d	c
y	b	a	c	d
z	d	b	c	a

Round 2

Current:	Proposals:	New Engagements:
w: –	a: –	w: d
x: a	b: –	x: a
y: b	c: z	y: b
z: –	d: w	z: c

Done - Stable:

a:x
b:y
c:z
d:w

Illustration: Example 2

Wmn	1st	2nd	3rd	4th
a	x	y	z	w
b	y	x	w	x
c	x	**y**	**z**	w
d	y	w	x	z

A * means: already proposed to that man

Man	1st	2nd	3rd	4th
w	d	b	a	c
x	b	a	d	c
y	**c**	**b**	**a**	**d**
z	d	b	c	a

Round 1

Proposals:	New Engagements:
a: x	w: –
b: y	x: a
c: x	y: b
d: y	z: –

Illustration: Example 2

Wmn	1st	2nd	3rd	4th
a	x*	y	z	w
b	y*	x	w	x
c	x*	y	z	w
d	y*	w	x	z

A * means: already proposed to that man

Man	1st	2nd	3rd	4th
w	d	b	a	c
x	b	a	d	c
y	c	b	a	d
z	d	b	c	a

Round 2

Current:	Proposals:	New Engagements:
w: –	a: –	w: d
x: a	b: –	x: a
y: b	c: y	y: ~~b~~ c
z: –	d: w	z: –

b is "jilted"

Illustration: Example 2

Wmn	1st	2nd	3rd	4th
a	x*	y	z	w
b	y*	x	w	x
c	x*	y*	z	w
d	y*	w*	x	z

A * means: already proposed to that man

Man	1st	2nd	3rd	4th
w	d	b	a	c
x	b	a	d	c
y	c	b	a	d
z	d	b	c	a

Round 3

Current:	Proposals:	New Engagements:
w: d	a: –	w: d
x: a	b: x	x: ~~a~~ b
y: ~~b~~ c	c: –	y: ~~b~~ c
z: –	d: –	z: –

a is "jilted"

Illustration: Example 2

Wmn	1st	2nd	3rd	4th
a	x*	y	z	w
b	y*	x*	w	x
c	x*	y*	z	w
d	y*	w*	x	z

A * means: already proposed to that man

Man	1st	2nd	3rd	4th
w	d	b	a	c
x	b	a	d	c
y	c	b	a	d
z	d	b	c	a

Round 4

Current:	Proposals:	New Engagements:
w: d	a: y	w: d
x: ~~a~~ b	b: –	x: ~~a~~ b
y: ~~b~~ c	c: –	y: ~~b~~ c
z: –	d: –	z: –

a's proposal not accepted by y

(no change)

Illustration: Example 2

Wmn	1st	2nd	3rd	4th
a	x*	y*	z	w
b	y*	x*	w	x
c	x*	y*	z	w
d	y*	w*	x	z

A * means: already proposed to that man

Man	1st	2nd	3rd	4th
w	d	b	a	c
x	b	a	d	c
y	c	b	a	d
z	d	b	c	a

Round 5

Current:	Proposals:	New Engagements:
w: d	a: z	w: d
x: ~~a~~ b	b: –	x: ~~a~~ b
y: ~~b~~ c	c: –	y: ~~b~~ c
z: –	d: –	z: a

Stable:

a:z

b:x

c:y

d:w

FPDA Algorithm a bit more formally

- Maintain "already proposed (AP) list" for each woman (initially empty)
- Maintain "current engagements (CE) list" (initially empty)
  - man *m*: "current partner" is partner in CE list (none if not in list)
  - woman *w*: "engaged" if in CE list, "unengaged" if not
- At any round *t*:
  - each unengaged woman *w* proposes to the man *m*, not on her AP list, that is most preferred in her preference ranking $\succ_w$
  - for each man who receives proposals at round *t*, let *Prop(m,t)* be the set of women who proposed to *m* *plus* his current partner (if any)
    - let *w** be most preferred woman in his preference ranking $\succ_m$ among women in *Prop(m,t)*
    - add *(m,w*)* to CE list (and remove *(m,w)* if *m* currently engaged to someone *w* other to *w**)
  - stop if every woman is engaged: CE list is set of stable marriages

Why Does FPDA Work?

- **Thm:** FPDA returns a stable matching: here's a rough argument:
  - Sequence of women to which any man is engaged is *strictly improving*
    - Obvious, since man only accepts better and better offers
    - So any man will be matched to best proposer he has seen so far
  - Sequence of men proposed to by any woman get worse in each round
  - Now suppose (m,w) are a blocking pair for the μ produced by FPDA
  - Since w prefers m to $\mu(w)$, she must have proposed to m before $\mu(w)$
  - But if m prefers w to $\mu(m)$, either:
    - he would have either rejected $\mu(m)$ when proposed to by w (if w proposed *after* $\mu(m)$); or
    - he would not have accepted $\mu(m)$'s proposal (if w proposed *before* $\mu(m)$), since he would be engaged to either w or someone even more preferred than w)
  - So there can't be any blocking pairs in the marriage produced by FPDA: so it must be stable!

Properties of Deferred Acceptance

- **Thm:** FPDA requires at most $O(n^2)$ proposals *(n = #men = #women)*
  - this is easy to see
  - at each round, there must be at least one proposal, so we move down into some women's preference list at least "one man deeper" each time
  - there are *n* women, and *n* entries in each of their lists
  - example below shows that you can get very close to this
    - the (male proposing!) MPDA matching (shaded) requires 21 proposals and 17 rounds

Man	1st	2nd	3rd	4th	5th
v	a	b	c	d	e
w	b	c	d	a	e
x	c	d	a	b	e
y	d	a	b	c	e
z	a	b	c	d	e

Wmn	1st	2nd	3rd	4th	5th
a	w	x	y	z	v
b	x	y	z	v	w
c	y	z	v	w	x
d	z	v	w	x	y
e	v	w	x	y	z

Properties of Deferred Acceptance

- We don’t have a measure of social welfare
  - *we don’t add up the rank positions of the matches*
- But we can still ask if the resulting matching is good for somebody, e.g., for a specific man or woman
  - note: for any specific problem, there can be multiple stable matchings
- For example, for any woman *w*, let’s define *the optimal partner for w*, call it *opt(w)*, to be the most preferred man she could be matched with in any stable matching
  - e.g., suppose *w*: $m1 \succ_w m2 \succ_w m3 \succ_w m4 \succ_w m5$
  - consider all stable matchings: let’s imagine there are five of them
  - suppose *w* is matched to *m4* in two of them, *m5* in two, *m2* in one
  - then *opt(w) = m2*
  - *This is the best partner w could hope for in any stable matching*
- A matching is *female-optimal* if every woman *w* is matched to *opt(w)*

Good News for the Women

- **Thm:** FPDA returns the female-optimal stable matching
  - Suppose this is not true. Then some woman *w* was rejected by her optimal man *m=opt(w).*
    - *Remember women propose to men in order.*
  - Let *t* be earliest round at which this happened. In other words, *t* is the first round where some woman *w* was rejected by her optimal man *m.*
  - Then *m* must be engaged to some *w** at round *t* that he prefers to *w*.
    - *Otherwise m would have accepted w's proposal.*
  - We know *w** was not rejected by *opt(w*)* by round *t (remember t is the first round where some woman was rejected by her optimal partner).*
  - So we must have either *m=opt(w*)* or $m \succ_{w^*} opt(w^*)$
    - *Remember women propose to men in order, so w* proposes in order too.*
  - Now consider a stable matching in which *m=opt(w)* and *w* are paired
    - Call this stable matching *S*: it must exist by definition of *opt(w)*
    - And *w** is paired with some man *m** in *S*, (*m** cannot be preferred to *opt(w*)*)
  - But we've shown $m \succcurlyeq_{w^*} opt(w^*) \succcurlyeq_{w^*} m^*$ (one of the preferences is strict)
  - … and also that $w^* \succ_m w$
  - So *(m, w*)* form a blocking pair for *S*, contradicting stability of *S*.

Bad News for the Men…

- For any man *m*, let's define *the pessimal partner for m*, call it *pess(m)*, to be the least preferred woman he could be matched with in any stable matching
- A matching is *male-pessimal* if every *m* is matched to *pess(m)*

- **Thm:** FPDA returns the male-pessimal stable matching
  - use similar argument as for female-optimality

- So the female proposing algorithm gives the best possible stable matching from the perspective of any woman *w*, and the worst possible stable matching for any man *m*

… but there is a solution for the men

- Instead of women proposing to men, we can have men proposing to women: call it the *male-proposing deferred acceptance (MPDA)* algorithm
- MPDA returns the male-optimal, female-pessimal stable matching
- Interestingly, male and female interests are “in opposition”

Incentives: Should You Tell the Truth?

- If you are a matchmaking service, you'd like your clients to be truthful about their preferences
- It seems sensible for women to propose in order, and for men to accept their most preferred proposal
- Unfortunately, the Gale-Shapley is prone to *manipulation*:
  - there are situations where a person can end up with a better match by not revealing their true preferences
  - Let's see an example of manipulation by men in FPDA

Where Lying Benefits Your Matching

- Consider the following preferences
- First consider when everyone proposes and accepts proposals truthfully (always make/accept best proposal)
- Notice *m1* ends up with partner *w2* (his 2nd most preferred)

$w1: m2 \succ_{w1} m1 \succ_{w1} m3 \succ_{w1} m4$
$w2: m4 \succ_{w2} m1 \succ_{w2} m2 \succ_{w2} m3$
$w3: m1 \succ_{w3} m3 \succ_{w3} m2 \succ_{w3} m4$
$w4: m4 \succ_{w3} m3 \succ_{w3} m2 \succ_{w3} m1$

$m1: w1 \succ_{m1} w2 \succ_{m1} w3 \succ_{m1} w4$
$m2: w2 \succ_{m2} w1 \succ_{m2} w3 \succ_{m2} w4$
$m3: w3 \succ_{m3} w1 \succ_{m3} w2 \succ_{m3} w4$
$m4: w4 \succ_{m4} w3 \succ_{m4} w2 \succ_{m4} w1$

Round 1

Props:	Eng'd:
w1: m2	w1: m2
w2: m4	w2: –
w3: m1	w3: m1
w4: m4	w4: m4

Round 2

Props:	Eng'd:
w1: –	w1: m2
w2: m1	w2: m1
w3: –	w3: –
w4: –	w4: m4

Round 3

Props:	Eng'd:
w1: –	w1: m2
w2: –	**w2: m1**
w3: m3	w3: m3
w4: –	w4: m4

Where Lying Benefits Your Matching

- Consider the following preferences
- Now suppose *m1* "lies" at round 2: he rejects *w2*'s proposal, even though he prefers *w2* to his current engaged partner *w3*
- *Now m1 ends up with a better partner: w1 rather than w2!*
  - *w2 moves to m2 (who accepts, freeing up w1 for m1)*

$w1: m2 \succ_{w1} m1 \succ_{w1} m3 \succ_{w1} m4$
$w2: m4 \succ_{w2} m1 \succ_{w2} m2 \succ_{w2} m3$
$w3: m1 \succ_{w3} m3 \succ_{w3} m2 \succ_{w3} m4$
$w4: m4 \succ_{w3} m3 \succ_{w3} m2 \succ_{w3} m1$

$m1: w1 \succ_{m1} w2 \succ_{m1} w3 \succ_{m1} w4$
$m2: w2 \succ_{m2} w1 \succ_{m2} w3 \succ_{m2} w4$
$m3: w3 \succ_{m3} w1 \succ_{m3} w2 \succ_{m3} w4$
$m4: w4 \succ_{m4} w3 \succ_{m4} w2 \succ_{m4} w1$

Round 1 Props:	Round 1 Eng'd:	Round 2 Props:	Round 2 Eng'd:	Round 3 Props:	Round 3 Eng'd:	Round 4 Props:	Round 4 Eng'd:
w1: m2	w1: m2	w1: –	w1: m2	w1: –	w1: –	w1: m1	**w1: m1**
w2: m4	w2: –	w2: m1	w2: –	w2: m2	w2: m3	w2: –	w2: m3
w3: m1	w3: m1	w3: –	w3: m1	w3: –	w3: m3	w3: –	w3: m3
w4: m4	w4: m4	w4: –	w4: m4	w4: –	w4: m4	w4: –	w4: m4

Incentive Properties of Gale-Shapley

- Not easy to prove: but in FPDA, women can never benefit from “lying”: they should always propose in order of preference

- But men can benefit in certain settings (see example above) by “lying”: i.e., by rejecting a proposal that is preferred to another proposal or current partner

- Situation is opposite in MPDA: males should propose in order, and females can gain by “false” rejections

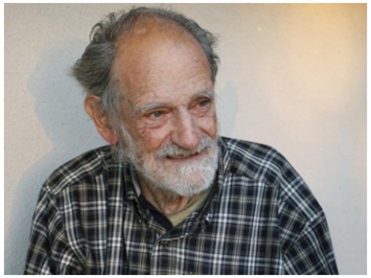
Extensions of Gale-Shapley

- Handling ties (indifference)
  - A woman might be *indifferent* between several men
  - E.g., ***w1***: $m2 \succ_{w1} m1 \succ_{w1} (m3,m5,m6) \succ_{w1} m4$
    - i.e., *m3, m5, m6* are tied, or equally good from *w1*'s perspective
  - Stability comes in different flavors now
    - without ties, any switch involving *w (m)* makes that *w (m)* better off or worse off
    - with ties, *w (m)* can be *indifferent* between her current partner and another *m*
  - So can define blocking pairs in different ways
    - B1: *(m,w)* should both be better off
    - Or B2: one of *(m,w)* should be better off and the other no worse
    - Or B3: neither of *(m,w)* should be worse off
  - Stable marriages relative to B1 (*weakly stable*) can be found by breaking ties in an arbitrary way and running GS
  - Stable marriages relative to B2 (*strongly stable*) and B3 (*super stable*) require some adjustments to the algorithm
    - Unlike weak stability: strong, super stable matching may not exist
    - E.g., suppose everybody likes everybody else equally

Extensions of Gale-Shapley

- Many-to-one matchings
  - college admissions: a college can accept multiple students, but has a limited number of spaces, and they have a ranking over students
  - students have rankings over colleges as well
  - stable matchings now assign multiple students to each college
  - other examples: NRMP, nationwide programs in Canada, US, and many other countries that assign residents to hospitals
- Model is a straightforward extension
  - each college *c* has a quota or capacity *q(c): c* can admit *q(c)* students
  - students “propose” to colleges like in normal GS
    - work down the list in order of most preferred colleges
  - colleges “hold” their most preferred students
    - if below quota, accept all (qualified) students who propose
    - if above quota, take on most preferred students, and “release” students who are least preferred (to get down to quota)
  - analysis very similar (one neat wrinkle: colleges can “lie” about their quotas to get some advantages in some circumstances)

Importance on Stable Matchings
(2012 Nobel Prize in Economics)



Lloyd Shapley used so-called cooperative game theory to study and compare different matching methods. A key issue is to ensure that a matching is stable in the sense that two agents cannot be found who would prefer each other over their current counterparts. Shapley and his colleagues derived specific methods – in particular, the so-called Gale-Shapley algorithm – that always ensure a stable matching. These methods also limit agents' motives for manipulating the matching process. Shapley was able to show how the specific design of a method may systematically benefit one or the other side of the market.

Alvin Roth recognized that Shapley's theoretical results could clarify the functioning of important markets in practice. In a series of empirical studies, Roth and his colleagues demonstrated that stability is the key to understanding the success of particular market institutions. Roth was later able to substantiate this conclusion in systematic laboratory experiments. He also helped redesign existing institutions for ***matching new doctors with hospitals, students with schools, and organ donors with patients***. These reforms are all based on the Gale-Shapley algorithm, along with modifications that take into account specific circumstances and ethical restrictions, such as the preclusion of side payments.