Lecture 4

David Zynda

August 30, 2018

Just basic review from last lecture and notations

First, let DA = Deferred Acceptance Algorithm.

Brief Review of DA:

Set of acceptable partners for i: $A(i) = \{j \in J : j \succeq_i i\}$

$X \subset J$ $\max_i X = \{j \in X : j \succeq_i j' \forall j' \in X\}$

if $X \neq \emptyset$ then $\max_i X = \emptyset$

DA (T-proposing)

Round 1

$A^1(t) = A(t)$ set for each $t \in T$ set of acceptable agents for agent t

$$\hat{P}^1 : T \to B \cup \{\emptyset\} : \hat{P}^1(t) \in \max_t A^1(t)$$

specify proposal to some maximal agent that is acceptable to her (t)

Set for each $b \in B$

$$P^1(b) = (\hat{P}^1)^{-1}(\{b\}) \cap A(b) = \{t \in A(b) : \hat{P}^1(t) = b\}$$

Induce set of proposals for b agents: the set of t's that are acceptable to b. Could be empty

$$\hat{\mu}^1 : B \to T \cup \{\emptyset\}$$
$$\hat{mu}^1 \in \max_b P^1(b)$$

Given this, the b agents will choose a maximal agents from the set of his round one proposals.

$A^{k+1}(t) =$ either **(1)** or **(2)**

(1) $A^k(t)$ if $\hat{\mu}^k(\hat{P}^k) = t$ meaning t proposed at k to $b = \hat{P}^k(t)$ and b accepted t at k.

But, he if rejected the offer, I throw him out of the running:

(2) $A^k(t) \setminus \{\hat{P}^k(t)\}$ Takeaway particular b from set if denied in step 1

if

$$\hat{\mu}^k(\hat{P}^k(t)) \neq t$$

If I was accepted on round k, then I offer the same offer to the same person on round $k+1$.

$\hat{P}^{k+1}(t) = \hat{P}^k$ if it is the case that $\hat{P}^k(t) \in A^{k+1}(t)$

$\hat{P}^{k+1}(t) \in \max_t A^{k+1}(t)$ if it is the case $\hat{P}^k(t) \notin A^{k+1}(t)$

The set of proposals that b has on round $k+1$:

$$P^{k+1}(b) = (\hat{P}^{k+1})^{-1}(\{b\}) \cap A(b)$$

Which is equivalent to:

$$\{t \in T : \hat{P}^{k+1}(t) = b : t \in A(b)$$

I hold on to exactly the same offer if it is the case that b is still a maximizer for me.

$$\hat{\mu}^{k+1} : B \to T \cup \{B\}$$

$$\hat{\mu}^{k+1}(b) = \hat{\mu}^{k}(b)$$

if it is the case that:

$$\hat{\mu}^{k+1}(b) \in \max_b P^{k+1}(b)$$

And

$$\hat{\mu}^{k+1}(b) = \max_b P^{k+1}(b)$$

if it is the case that:

$$\hat{\mu}^{k+1}(b) \notin \max_b P^{k+1}(b)$$

Matching

We defined a matching from agents of one side of a market to another side of the market such that: $\mu : (T \cup B) \to (T \cup B)$

If you go through the rounds the t agents make new proposals. As they make new proposals, it is worse for t agents. As t makes new proposals, it is also better for the b agents since they get more options which might be preferred.

Lemma 2

1. For each k and each $t \in T$, $\hat{P}^k(t) \succeq_t \hat{P}^{k+1}(t)$.
2. For each k and each $b \in B$, $\hat{mu}^{k+1} \succeq_b \hat{\mu}^k(b)$.

Proof

(1)

Recall: $A^{k+1}(t) \subset A^(t)$

$$\hat{P}^{k+1}(t) \in \max_t A^{k+1}(t)$$

$$\hat{P}^{k}(t) \in \max_t A^{k}(t)$$

$$\implies \hat{P}^k(t) \succeq_t \hat{P}^{k+1}(t)$$

(2)

Suppose: $\hat{\mu}^k(b) = t$ and so $\hat{P}^k(t) = b$

$$A^{k+1} = A^k(t) \implies \hat{P}^{k+1}(t) = \hat{P}^k(t) = b$$

$$\implies t \in \hat{P}^{k+1}(t)$$

if $\hat{mu}^{k+1}(b) = t$ then certainly $\hat{mu}^{k+1}(b) \succeq_b \hat{mu}^k(b)$ and if $\hat{mu}^{k+1}(b) \neq t$ then $\hat{mu}^{k+1}(b) \succeq_b t = \hat{\mu}^k(b)$

Lemma 3

Fix $b \neq b'$ If $\hat{mu}^k(b) = \hat{mu}^k(b')$ then $\hat{mu}^k(b) = \hat{mu}^k(b') = \emptyset$

Proof:_

Fix $b \neq b'$

$$\hat{mu}^k(b) \in P^k(b)$$

$$\hat{mu}^k(b') \in P^k(b')$$

$$P^k(b) \cap P^k(b') = \emptyset$$

If $t \in P^k(b) \cap P^k(b')$ then $\hat{P}^k(t) = b = b'$

This is a contradiction, since a function is mapping more than one thing to another (like one x to two y's). P^k is a function.

By Lemma 1 (last class) there exits a $k < \infty$ such that $\hat{mu}^K = \hat{mu}^k$ for all $k \geq K$.

$\mu^* : (T \cup B) \to (T \cup B)$

- $\mu^*(b) < \hat{\mu}^k(b)$ for each $b \in B$ such that $\hat{\mu}^k(b) \in T$
- $\mu^*(b) = b$ for each $b \in B$ such that $\hat{\mu}^k(b) = \emptyset$

By Lemma 3 for each $t \in T$, there is at most 1 b such that $\hat{\mu}^k(b) = t$.

Case 1: $\hat{\mu}^*(t) = b$ if $b = (\hat{\mu}^k)^-1(t)$ Case 2: $\mu^*(t) = t$ otherwise

So, we have constructed a matching which leads to Lemma 4:

Lemma 4

The matching $\mu^* : (T \cup B) \to (T \cup B)$ is stable.

Proof:

Need to show two things: 1. Matching is indivudally rations: $\mu^*(i) \succeq_i i$ 2. There is no blocking pair (t, b).

Step 1 for the t agent:

(1.t)

It suffices to show that for each $t \in T$ that is matched $\mu^*(b) \in B$ there is

$$\mu^*(t) \in A(t)$$

$$\mu^*(t) = b \implies \hat{P}^k(t) = b \implies b \in A^k(t) \subset A(t)$$

Step 1 for the b agent:

(1.b)

It suffices to show that for each $b \in B$ with $\mu^*(b) \in T$, $\mu^*(b) \in A(b)$. Notice by construction that $\mu^*(b) = \hat{\mu}^k(b) \in P^k(b) \subset A(b)$

(2)

Fix (t, b) and suppose contra hypothesis that (t,b) blocks μ^*. This means $b \succeq_t \mu^*(t)$ and $t \succeq_b \mu^*(b)$. There exists some $k < K$ such that $\hat{P}^k(t) = b$ and $\hat{\mu}^k(b) \neq t$.

then:

$$\mu^*(b) = \mu^*(b) \succeq_b \mu^k(b) \succeq_b t$$

This follows from Lemma 2. This implies $\mu^*(b) \succeq_b t$ which contradicts that t forms a blocking pair.

Algorithm tells us how to construct a stable match. Presuming that designer knows all inputs and preferences. Program a computer to make all those matches. However, we will talk later about when designer does not know preferences. But, now, we assume so, and have shown this is possible to construct.

No one has incentive to deviate. Or we hope. We want them to be normatively desirable. Are they good for society? There are lots of different criteria of how we understand normatively desirable matches

Are Stable Matchings Normatively desirable?

- Criteria 1: *Are stable matches Pareto efficient?*
- Stable matches tend to be pareto efficient (hw problem to show)
- Pareto efficient is a state of allocation of resources from which it is impossible to reallocate so as to make any one individual or preference criterion better off without making at least one individual or preference criterion worse off.
- Criteria 2: *selecting the best stable matching among all the stable matchings possible*

Example: (from last class)

Imagine an environment with, say, two firms T and two employees B:

$$T = \{t_1, t_2\}$$
$$B = \{b_1, b_2\}$$

Preferences of T:

$$t_1 : b_1 \succsim_{t_1} b_2 \succsim_{t_1} t_1$$
$$t_2 : b_2 \succsim_{t_2} b_1 \succsim_{t_2} t_2$$

Preferences of B:

$$b_1 : t_2 \succsim_{b_1} t_1 \succsim_{b_1} b_1$$
$$b_2 : t_1 \succsim_{b_2} t_2 \succsim_{b_2} b_2$$

Recall that there were two stable matches:

(1) All B agents get best match (B-proposing DA algorithm)

$$\mu(t_1) = b_2$$
$$\mu(t_2) = b_1$$
$$\mu(b_2) = t_1$$
$$\mu(b_1) = t_2$$

(2) All T agents get best match. (T-proposing DA algorithm)

$\mu(t_1) = b_1$ and $\mu(t_2) = b_2$ $\mu(b_1) = t_1$ and $\mu(b_2) = t_2$

Side of the Market Preferences

If I think about the I proposing DA algorithm:

$\mu_{ID} : (T \cup B) \to (T \cup B)$

Introduce an order: $\geq_I$ such that $\mu \geq_I \mu'$ if, for each $i \in I : \mu(i) \succeq_i \mu'(i)$

Write:

$\mu >_I \mu'$ if $\mu \geq_I \mu'$ and there exists some $i \in I$ such that $\mu(i) \succ_i \mu'(i)$

I will call $\mu : (T \cup B) \to (T \cup B)$ an **I-Optimal** stable match if:

1. μ is stable.
2. $\mu \geq_I \mu'$ for all other stable matches μ.

Proposition (Gale - Shapley 1962)

Suppose preferences are strict. Then for each $I \in \{T, B\}$:

$\mu_{ID} : (T \cup B) \to (T \cup B)$ is an I-optimal stable match.

The proof and example will be in the next class.