CS 6840: Algorithmic Game Theory **Spring 2017**

Lecture 37: May 1

Lecturer: Sid Banerjee *Scribe: Vidit Bhargava*

37.1 Game Setting

We will look at the game rules and settings.

- Two groups of agents, say Students (S) and Colleges (C)
- For now we assume that there is one-one correspondence between the two groups.
- Students have a strict ordering $\succ_s$ over $C \cup \{\phi\}$ i.e. a strict preference list of colleges they want to get into. The ϕ represents an agent being unmatched.
  - If $c \succ_s c'$ then s prefers c over c'
  - If $\phi \succ_s c$ then s prefers not going to college over c
- Colleges have a similar ordering $\succ_c$ over $S \cup \{\phi\}$

Note : In order to simultaneously accommodate the one-one correspondence rule and null choice, we would need to pad the ϕ so that their is one ϕ for each student and one for each college.

Matching is a function $\mu : SUC- > SUC \cup \{\phi\}$ where:

1. $\mu(s) \in C \cup \{\phi\}$ *where* $s \in S$
2. $\mu(c) \in S \cup \{\phi\}$ *where* $c \in C$
3. $\mu(s) = c \iff \mu(c) = s$ *where* $s \in S \wedge c \in C$

A matching μ is stable if it is NOT :

- Blocked by an individual i.e. there does not exist s such that $\phi \succ_s \mu(s)$
- Blocked by a college, i.e., there does not exist c such that $\phi \succ_c \mu(c)$
- Blocked by a pair i.e. there does not exist s, c such that $c \succ_s \mu(s)$ and $s \succ_c \mu(c)$

37.2 Gale Shapley Algorithm

Stable matching always exist (int the above game setting).
Algorithm : See 4820 Textbook

Theorem 37.1 *Gale Shapley Algorithm terminates in a stable matching in $O(n^2)$ time where n is the size of the 2 sets.*

Corollary 37.2 $\exists$ *a stable matching.*

37.3 Rankings

We define $R_s(c) = |\{c' : c' \succ_s c\}|$
We define $R_s(\mu) = \frac{1}{|S-S'|}\sum_{s \in S-S'} R_s(\mu(s))$ where S' is the set of unmatched students
Note : We made the assumption that being matched to ϕ yields no ranking.

For all $s \in S$, let $l(s)$ denote the lowest ranked partner of student s in any stable matching.
For all $s \in S$, let $h(s)$ denote the highest ranked partner of student s in any stable matching.
For all $c \in C$, let $l(c)$ denote the lowest ranked partner of college c in any stable matching. For all $c \in C$, let $h(c)$ denote the highest ranked partner of college c in any stable matching. CPDA (College proposes) results in the following :

- every $s \in S$ gets its $l(s)$
- every $c \in C$ gets its $h(c)$

Similarly SPDA (Student proposes) results in the following :

- every $s \in S$ gets its $h(s)$
- every $c \in C$ gets its $l(c)$

This has the following consequences :

- Safe from coalition-deviations (solution is said to be in the core). In CPDA, all colleges get the highest possible stable partners, thus they have no incentive to collate and deviate. Similarly in SPDA, students do not have any such incentive.
- The set S' and C' are the same in all stable matchings
- The mechanism is not strategy proof.

37.4 Recent Findings

37.4.1 Pittel

Consider uniformly random preference lists and $|S| = |C| = n$, then

$$R_c(\mu_{CPDA}) = O(log(n))\ with\ high\ probability$$

$$R_s(\mu_{CPDA}) = O(\frac{n}{log(n)})\ with\ high\ probability$$

37.4.2 Ashlagi et al.

What happens when $|C| \neq |S|$?
Consider uniformly random preference lists, then

$$P(\exists\ multiple\ stable\ matchings) \searrow 0\ as\ n \to \infty$$

References

[1] Itai Ashlagi, Yash Kanoria and Jacob D. Leshno, Unbalanced Random Matching Markets: the Stark Effect of Competition, Journal of Political Economy, 125(1), 69-98, 2017.

[2] Boris Pittel. On a random instance of a stable roommates problem: Likely behavior of the proposal algorithm. Combinatorics, Probability and Computing, 2:5392, 1993.