CS 286r: Assignment, Matching and Dynamics

Prof. David C. Parkes
SEAS

Class outline

- Overview of course
- Arrow's theorem
- Gibbard-Satterthwaite theorem
- Introducing money (Vickrey)
- Classic mechanisms:
  - median mechanism (social choice)
  - top-trading cycle algorithm (house allocation)
  - deferred-acceptance algorithm (matching)

What is CS 286r?

- Topics at the interface between Computer Science and Economics.
- Rotating topics class, last taught in Fall'08 by Prof. Yiling Chen on "social computing."
- Seminar style

Course Goals

- Provide an introduction to an emerging, interdisciplinary literature
- Develop a level of comfort with both economic and computational thinking
- Develop general skills related to reading papers, identifying research questions
- Provide a basis for continued research.

Fall 2009

- Assignment, Matching and Dynamics
- Algorithmic, game-theoretic and conceptual questions related to “market engineering” with applications across societal, Internet, governmental and distr. computing settings.
- Focus on mechanism design with and without money, bridging from classic economic theories to recent computational directions.

- **Assignment problems**:
  - N agents, G goods (tasks), prefs. $S \succ_i T$ on S, $T \subseteq G$.
  - Seek an assignment of goods to agents
  - Variations: with and without money, structure on preferences, initial endowment, side constraints.
  - Design criteria? Examples?

- **Assignment problems**:
  - N agents, G goods (tasks), prefs. $S \succ_i T$ on S, $T \subseteq G$.
  - Seek an assignment of goods to agents
  - Variations: with and without money, structure on preferences, initial endowment, side constraints.
  - Design criteria? Examples?

- **Matching problems**:
  - Two agent sets; e.g. S ("students") and F ("firms")
  - Two-sided preferences, $f_1 \succ_s f_2$, $s_1 \succ_f s_2$
  - Seek a bipartite matching
  - Variations: indifferences, externalities, multiple units.
  - Design criteria? Examples?

Motivating problems

- Harvard PDP-1 computer (1968)
- Sponsored Search (2002+)
- Wireless spectrum allocation (1994+)
- Undergraduate housing (1999+)
- Course allocation (1996+)

Motivating problems

- Harvard PDP-1 computer (1968)
- Sponsored Search (2002+)
- Wireless spectrum allocation (1994+)
- Undergraduate housing (1999+)
- Course allocation (1996+)

- Babysitting co-ops (1970s+)
- High school matching (2003+ in New York City)
- Medical students to residencies (1998+ NMRP)
- File-sharing networks (1999+)
- TV Ad Auctions (2009+); Ad Exchanges (2006+)
- Kidney Exchanges (2004+)
- Crowdsourcing (2001+; e.g. Innocentive,TopCoder)

Dynamics

- Agents arrive and depart
- Set of available goods change
- Preference shocks
- Examples
  - new jobs arriving to schedule on PDP-1
  - dynamic assignment of ad inventory to ads
  - new donor-recipient pairs arriving
  - housing: seniors leave, freshmen arrive
  - changing inventory of last-minute theater tickets
  - learning “taste” for different files

Some technical themes

- Approximate strategyproofness: qualitative and quantitative “degrees” of non-manipulability
- Characterizations and algorithms for dynamic assignment and matching without money
- Approximations via mechanisms without money (AMD agenda)
- Preference representations; role of simplicity?
- Role of transitive trust and “scrip” systems for distributed work platforms
- Using methods of online stochastic optimization within dynamic mechanisms

An active research area

- Nexus of game theory, computer science (AI and theory) and operations research.
- Many conferences:
  - International Conference on Autonomous agents and Multi-Agent systems (in Chicago, May 2010)
  - ACM Conference on Electronic Commerce (at Harvard, June 2010)
  - Int. Workshop on Computational Social Choice (COMSOC, Dusseldorf, Sept 2010)
  - Workshop on Internet and Network Economics (WINE, Rome, Dec 2009)
  - Related papers in STOC/FOCS, AAAI/IJCAI
  - NBER Market Design; Univ. Bonn “Visiting program in Mechanism Design” etc.

Course Requirements

- Algorithms (CS 124), AI (CS 181, 182) and optimization (AM 121) helpful.
- Microeconomic theory (EC 1011a, EC 1056, EC 2056) helpful.
- Ph.D. students in Computer science and Economics/Business-Economics welcomed!
- A level of mathematical sophistication.

- Talk to me if you have concerns about your background. Look at the papers.

Course Structure

- Introductory lectures: this one, game theory, MD with money, MD without money
- Student led presentation and discussion:
  - ≈ two papers per class
  - meet with me before class to discuss the papers
- Submit comments (from class page)
- Two problem sets
- Final paper

Grading policy

- Participation and comments: 25-30%
- Problem sets: 20-25%
- Presentation and leading discussion: 15%
- Final paper 35%

Final Paper

- Develop a deep understanding of a specific topic related to the class
- May be computational, theoretical, empirical or experimental. Work in a pair with approval.
  - Best papers work on an open research problem.
- Can be an expositional paper: two related technical papers, including an exposition of (at least) two technical results and critical discussion.
- Submit proposal, give a short presentation. Final paper by the end of reading week.

Office hours

- David Parkes, parkes@eecs.harvard.edu
- 2.30-3.30pm Wednesdays, MD 229 (TODAY!)
- Later: Move, likely to Thurs/Fri to meet with students in advance of presenting papers

- Shaili Jain, shailij@eecs.harvard.edu
- 2-4pm Tuesdays, second floor MD lobby

Schedule

- http://www.eecs.harvard.edu/cs286r/schedule.html

For Wednesday 9/9

- Submit comments on reading from reading taken from Chapters 3, 5 and 6 of "Multiagent Systems" by K. Leyton-Brown and Y. Shoham. Reading is posted on the class schedule.
- What is unclear? What would you like to hear about in class? What did you enjoy?
- Two paragraphs maximum.

Class outline

- Overview of course
- Arrow's theorem
- Gibbard-Satterthwaite theorem
- Introducing money (Vickrey)
- Classic mechanisms:
  - median mechanism (social choice)
  - top-trading cycle algorithm (house allocation)
  - deferred-acceptance algorithm (matching)