Networks II: Market Design—Lecture 2
Matching Markets with Non-transferable Utilities

ARPITA GHOSH

Dept. Of Information Science,
Cornell University

'Stuff'

- New to course:
  - Please catch up on Lecture 1, especially course information
  - Missed material: Slides on CMS (recall **not** full transcript!)
- Recall also: In general, responsible for content of lecture (logistics and course material), whether present in class or not

Outline

- Introduction to matching markets without money
- Introduction to preferences
- Binary preference structures: Ideal allocations; perfect matchings
- Introduction to Matching Theorem

Matching

- Matching markets: Allocation of indivisible resources
- Two sets of entities: Entity from one side of market 'can be matched' only with an entity from other side
  - Value created only when entities from opposite sides are paired
  - Entities: Agents, objects
  - Retail goods, markets for services, knowledge exchange, . . .
- Matching markets with *'one-sided preferences'*:
  - Agents on one side, objects on the other
  - Allocation/exchange of objects among agents
- Two-sided preferences: Agents on both sides

Allocations in matching markets

- Central issue: Which entities to pair together
  - How to choose matching?
- Networks I: Maximize *total value* of match
  - *Suppose* each edge has 'value' v_{ij}
  - v_{ij}: Agent i's (maximum) willingness to pay for object j
  - Pair agents and objects to maximize welfare: $\sum v_{ij}$
  - Market-clearing *prices* achieve welfare-maximizing allocation

Matching markets without money

- Networks I: Matching markets with money
- Money—payments—central to "making all agents happy"
  - Cannot always give every agent item they like best
  - But: Can set prices so every agent likes her assignment best
- Money: *Transfer utility* between agents
- What happens in matching markets where payments are infeasible?
  - 'Matching markets with non-transferable utilities'

Markets without money: New conceptual questions

- Settings with non-transferable utilities
  - Agent-object markets: Dorm rooms, course allocation ...
  - Agent-agent markets: Students-schools, men-women, ...
- 'What happens' in matching markets without money?
  - Recall central issue: How to allocate/pair entities?
  - Agents with preferences: Cannot always make everybody happy
  - No money: Cannot make up for unhappiness with money!
  - What are 'good' allocations?

But before we go there...

- Are there really 'important' markets without money?
  - And can anything really be done about them?

- Yes! 2012 Nobel Prize in Economics for 'market design'
  - Al Roth and Lloyd Shapley: "For the theory of stable allocations and the practice of market design"
  - Market design: Moving beyond traditional economics (analyzing institutions) to *designing* for better outcomes
  - All cited applications: Design of matching markets without money

A quick aside: ‘Stable allocations and market design’

- ‘Theory of stable allocations’:
  - Marriage model, Gale-Shapley algorithm, ...
- ‘Practice of market design’:
  - Redesign of NRMP (hospital-intern residency) match ($\sim$ 1984)
  - Redesigning school choice assignments for fairness (Boston, NYC, $\sim$ 2003)
  - Designing *kidney exchanges* to maximize number of transplants, ... ($\sim$ 2003 onwards)

Outline

- Introduction to matching markets
  - Matching markets: One-sided and two-sided preferences
  - Markets with non-transferable utilities
- One-sided markets
  - Binary preference structures: Ideal allocations and perfect matchings
  - Existence of perfect matchings: Proving the Matching Theorem

- Our question: 'What happens' in matching markets without money?
  - How to allocate/pair entities?
- Note: Need 'language' for preferences—could be a **design element**!
- To begin: Matching markets with one-sided preferences
- Simplest preferences: Binary $(0-1)$ preferences
  - Each agent finds each object *acceptable* or *not*
  - Room allocation, Doodle polls for timeslots, ...

Doodle

DASHBOARD | ARPITA GHOSH

Final project presentation

Edit your poll | 9 | 0 | less than a minute ago

Where: Rhodes 490

Table view | Administration

Most popular option: Wed 230-250pm | Close poll

9 participants	Wed 930-950am	Wed 950-1010am	Wed 1010-1030am	Wed 1030-1050am	Wed 1050-1110am	Wed 130-150pm	Wed 150-210pm	Wed 210-230pm	Wed 230-250pm	Wed 250-310pm
T.L.						✓	✓			
M.R.	✓	✓							✓	✓
R. R.	✓					✓	✓			
P.J.						✓	✓	✓	✓	✓
Z.Y.									✓	✓
J.O.		✓	✓	✓	✓				✓	✓
H.B.						✓			✓	✓
N.K.				✓	✓					
D.C.				✓	✓	✓	✓	✓	✓	
Arpita Ghosh	Yes (Yes) No	Yes (Yes) No	Yes (Yes) No	Yes (Yes) No	Yes (Yes) No	Yes (Yes) No	Yes (Yes) No	Yes (Yes) No	Yes (Yes) No	Yes (Yes) No
Yes	2	2	1	3	3	5	4	2	6	5
Ifneedbe	0	0	0	0	0	0	0	0	0	0
No	7	7	8	6	6	4	5	7	3	4

No option works for me | Save

Markets with one-sided binary preferences

- Canonical setting for binary preferences
  - Room allocation: n students, n rooms
  - A room is either acceptable to a student or not
- Choosing an allocation: Ideal of 'perfect happiness'
  - Can we assign students to rooms to 'make everyone happy'?
- Modeling choices: Binary preferences, no group preferences, no conditional preferences, centralized allocation, . . .

Markets with one-sided binary preferences

- *Abstract model:* Preferences as *bipartite graph* $\mathcal{M}$
  - Bipartite graph: Nodes can be partitioned into N_L, N_R so no edges amongst nodes in N_L, N_R
  - LHS nodes: rooms; RHS nodes: students
  - Edge: Room acceptable to student
- Matching: Subset of edges in $\mathcal{M}$ such that no two edges share an endpoint node
- Size of matching: Number of edges in it

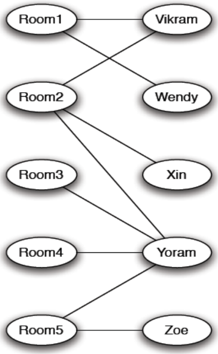
- *Perfect matching:* Set of edges that contains each node in $\mathcal{M}$, *exactly* once
- There is an assignment of rooms to students to 'make everyone happy' *iff* $\mathcal{M}$ contains a perfect matching
- How to decide whether or not $\mathcal{M}$ has a perfect matching?

Outline

- Introduction to matching markets
  - One-sided and two-sided markets
  - Markets with non-transferable utilities
  - Market design
- One-sided markets
  - Binary preference structures: Ideal allocations and perfect matchings
  - Existence of perfect matchings: The Matching Theorem

Constricted sets

- $\mathcal{M}$: Bipartite graph with n nodes on each side



- *Constricted* set S in $\mathcal{M}$: $|S| > |N(S)|$
  - For simplicity, in this course: S always on 'agent side' of graph

Constricted sets and perfect matchings: The Matching Theorem

How to decide whether or not $\mathcal{M}$ has a perfect matching?

Theorem

A bipartite graph (with equal numbers of nodes on the left and right) has no perfect matching iff *it contains a* constricted set.

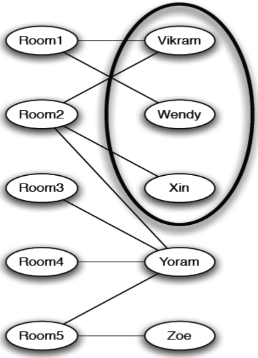
- Proven independently by Konig (1931); Hall (1935)
- 'Matching theorem'; 'Hall's theorem'

Why study the Matching Theorem?

- Why is the matching theorem useful?
  - Provide *evidence* of non-existence of perfect matching
  - Proof contains important ideas (including for 'improving' matchings in more complex settings)
- What will we learn from the proof?
  - *Enlarging a matching*: Concept of augmenting paths
  - How to *find* a perfect matching
  - Maximum matchings
- 'Improving' matchings in other models: Key ideas used in School Choice!

Proving the Matching Theorem

- A bipartite graph has no perfect matching **iff** it contains a constricted set $|S|$ with $|S| > |N(S)|$
- One direction is easy: *If* constricted set, no perfect matching

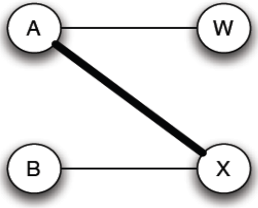


Proving the Matching Theorem

- ‘Only if’: No perfect matching $\Rightarrow$ Constricted set
  - No other reason for a perfect matching to not exist(!)
- Key issue: Identifying a constricted set, knowing *only* that there is no perfect matching
- Outline of proof structure:
  - No perfect matching: Maximum matching does not include all nodes
  - Start with a maximum matching
  - Try to enlarge maximum matching: Will fail
  - *Failed attempt at enlarging returns a constricted set* (unless original matching was perfect)!

Enlarging matchings

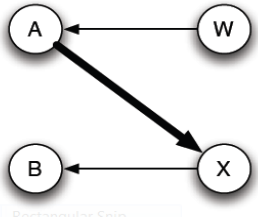
- How do we enlarge a *given* matching in a bipartite graph?



- *Alternating path*: A path that alternates between non-matching and matching edges

Enlarging matchings

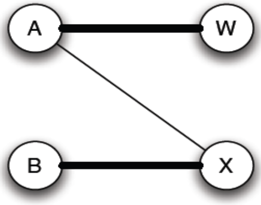
- *Augmenting paths*: Alternating paths with **unmatched** endpoints



- Claim: A matching in a bipartite graph can be enlarged if there is an augmenting path

Enlarging matchings

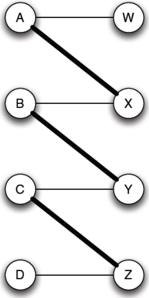
- *Augmenting paths*: Alternating paths with **unmatched** endpoints



- Claim: A matching in a bipartite graph can be enlarged if there is an augmenting path

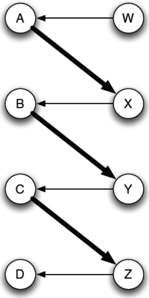
Finding augmenting paths

- Augmenting paths can be much longer than our previous example:



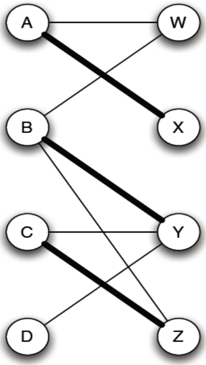
Finding augmenting paths

- Augmenting paths can be much longer than our previous example:



Finding augmenting paths

- And they aren't always easy to find!
- Is this a maximum matching?



- If not, can you find an augmenting path?

Proving Hall's theorem: Where are we now?

- Recall key issue: Identifying a constricted set, knowing *only* that there is no perfect matching
  - Start with a *maximum matching*
  - Try to enlarge it
  - Augmenting paths: A matching in a bipartite graph can be enlarged if there is an augmenting path
  - Show this produces a constricted set unless the matching was perfect
- Alternating Breadth-first Search: A way to look for augmenting paths so that a failure to find one produces a constricted set— unless the matching was perfect to start with