Chap.12 Matching Markets

Presentation

Jiao Luo

TUM

November 16, 2018

Start with a Question

Start with a Question

- Do you have experienced a match?

Start with a Question

- Do you have experienced a match?
- YES!

Start with a Question

- Do you have experienced a match?
- YES!
- Why am I so sure?

Start with a Question

- Do you have experienced a match?
- YES!
- Why am I so sure?

- AHA! This Seminar!
- For me *MatchingMarkets* $\succ$ *AuctionDesign* $\succ$ *Networks* (preference list)

Another Example - Matching

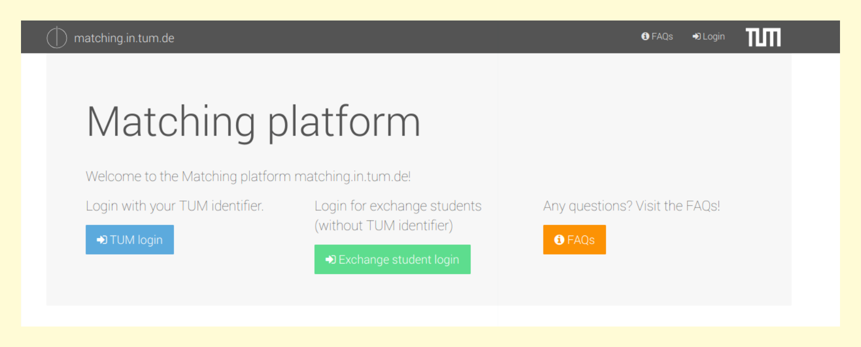


Figure: TUM Matching platform[1]

[1] https://matching.in.tum.de/

Another Example - Matching

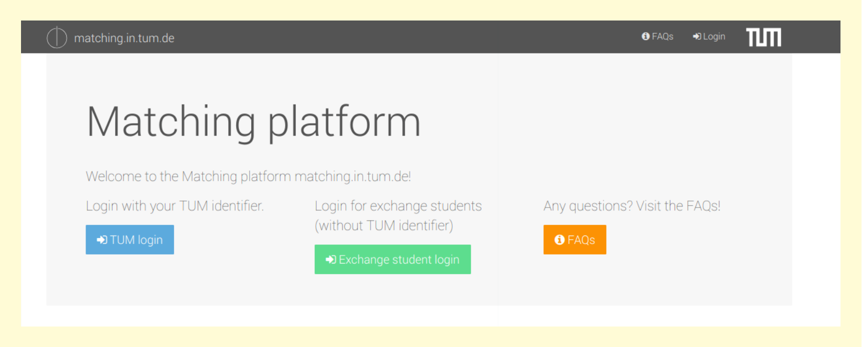


Figure: TUM Matching platform[1]

- Spoiler: It a tool from our DSS-Chair.

[1] https://matching.in.tum.de/

Set and Strict Preference Order

Agent	$x \in X$, the set of all agnets
Matching	$m \in M$, a set of machings
Strict Preference Order	$m \succ_x m'$, agent x strict prefers m to m'

Set and Strict Preference Order

Agent $x \in X$, the set of all agnets

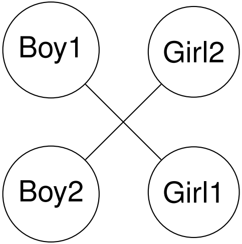
Matching $m \in M$, a set of machings

Strict Preference Order $m \succ_x m'$, agent x strict prefers m to m'

$$MatchingMarkets \succ_{Jigao} AuctionDesign$$

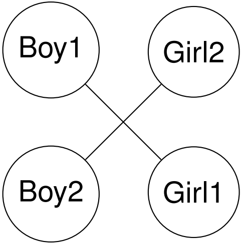
Three Types of Matching Problems

Two-Sided
Matching

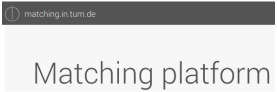


Three Types of Matching Problems

Two-Sided Matching

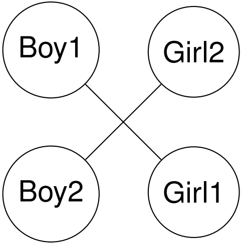


Assignment Problems

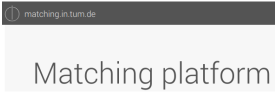


Three Types of Matching Problems

Two-Sided Matching



Assignment Problems



Kidney-Paired Donation

Matching in Graph

Unweighted Graphs in Matching

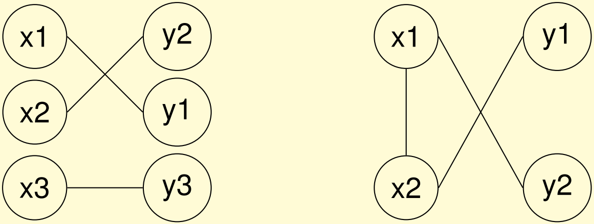
- a vertex v is matched in M: $M(v) \in V \setminus v : \{v, M(v)\} \in M$.
- Otherwise, v is unmatched in M: $\{v, \emptyset\}$

Bipartite Graph

- for every edge (u, v): u v in opposite subsets.

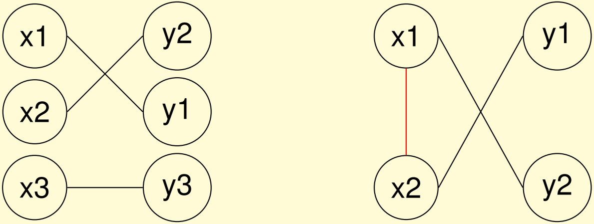
Matching in Graph - Quiz1

Following are two graphs representing the matching between 2 sides.
Which is a bipartite graph?



Matching in Graph - Quiz1

Following are two graphs representing the matching between 2 sides.
Which is a bipartite graph?



The left one is a bipartite graph.
The right one is not a bipartite graph.

Stable Marriage Problem

- a set of men X and a set of women Y.
- M is set of marriages.
- $|X| = |Y| = N = |M|$.
- Only strict preference order.
- If $(x, y) \in M$, then $x = M(y)$ and $y = M(x)$.
- Or unmatched

Assumption: No Payments

Assumption: Only Strict Preference Order

Assumption: Only Strict Preference Order

Assumption: Only Strict Preference Order

The strict preference order here: $y_{red} \succ_x y_{blue}$.
More specific $y_{red} \succ_x M(x)$.

Assumption: Only Strict Preference Order

The strict preference order here: $y_{red} \succ_x y_{blue}$.

More specific $y_{red} \succ_x M(x)$.

Maybe this moment y_{blue} prefer not to be matched with x any more?

Prefer to be Unmatched

- $\emptyset \succ_{y_{red}} x$
- x is *unacceptable* for y_{red}

Blocking Pair and Stable Marriage

Blocking Pair[2] if $(x, y) \notin M$ but:

- y prefers x to $M(y)$ and
- x prefers y to $M(x)$
- If $\emptyset \succ_y M(y)$ for y, then $(y, \emptyset)$ form a blocking pair. Similar for x.

Stable Marriage[3] no blocking pair for M

Unstable Marriage at least one blocking pair for M

[2] Dan Gusfield, Robert W.Irving. The Stable Marriage Problem: Structure and Algorithms. Chapter 1: Elementary Concept and Result(pp. 6). The MIT Press, 1989

[3] See footnote 2

Blocking Pair and Stable Marriage - Quiz2

Yes-No Questions: Are the following statements true?

- Man x and woman y form a blocking pair (x, y) for matching M if $M(x) \neq y$ and $M(x) \neq y$.

Blocking Pair and Stable Marriage - Quiz2

Yes-No Questions: Are the following statements true?

- Man x and woman y form a blocking pair (x, y) for matching M if $M(x) \neq y$ and $M(x) \neq y$.
  No It should be corrected with $y \succ_x M(x)$ and $x \succ_y M(y)$
- A stable matching is one in which no pair of agents prefer each other over their respective matches.

Blocking Pair and Stable Marriage - Quiz2

Yes-No Questions: Are the following statements true?

- Man x and woman y form a blocking pair (x, y) for matching M if $M(x) \neq y$ and $M(x) \neq y$.
  No It should be corrected with $y \succ_x M(x)$ and $x \succ_y M(y)$
- A stable matching is one in which no pair of agents prefer each other over their respective matches. Yes

Blocking Pair and Stable Marriage - Quiz2

Last one: Is this a unstable marriage regarding the context that we have talked about?

Blocking Pair and Stable Marriage - Quiz2

Last one: Is this a unstable marriage regarding the context that we have talked about?

I would say yes. We do have $y_{red} \succ_x M(x)$.

Blocking Pair if $(x, y) \notin M$ but:

- ~~y prefers x to $M(y)$~~ and
- x prefers y to $M(x)$
- If $\emptyset \succ_y M(y)$ for a woman y, then $(y, \emptyset)$ form a blocking pair.

But I assume the y_{blue} is $M(x)$ and at this moment she prefers not to be matched with x. Then a blocking pair $(y, \emptyset)$

Gale–Shapley Deferred Acceptance Algorithm

Theorem (Gale–Shapley 1962)

The Gale–Shapley algorithm guarantees to find a stable matching for ***any*** *problem instance.*

COLLEGE ADMISSIONS AND THE STABILITY OF MARRIAGE

D. GALE* AND L. S. SHAPLEY, Brown University and the RAND Corporation

1. Introduction. The problem with which we shall be concerned relates to the following typical situation: A college is considering a set of n applicants of which it can admit a quota of only q. Having evaluated their qualifications, the admissions office must decide which ones to admit. The procedure of offering admission only to the q best-qualified applicants will not generally be satisfactory, for it cannot be assumed that all who are offered admission will accept. Accordingly, in order for a college to receive q acceptances, it will generally have to offer to admit more than q applicants. The problem of determining how many and which ones to admit requires some rather involved guesswork. It may not be known (a) whether a given applicant has also applied elsewhere; if this is known it may not be known (b) how he ranks the colleges to which he has applied; even if this is known it will not be known (c) which of the other colleges will offer to admit him. A result of all this uncertainty is that colleges can expect only that the entering class will come reasonably close in numbers to the desired quota, and be reasonably close to the attainable optimum in quality.

Gale–Shapley Deferred Acceptance Algorithm[4]

Algorithm 1: GS DA Algorithm (men-proposing version)

Input : a list of preference order
Output: a stable marriage M

```
Initialize M to empty matching.
while x is unmatched and hasn't proposed to every woman do
    x proposes to the most-preferred woman y to whom he has
      not yet propose;
    if y is unmatched then
        add (x, y) to M;
    else if y prefers x to current partner x' then
        Replace (x', y) with (x, y) in M;
    else
        y rejects x;
```

[4] D. Gale and L. S. Shapley. College admissions and the stability of marriage. The American Mathematical Monthly (pp.69:9–15). 1962

Gale–Shapley D.A. Algorithm - Quiz3

Try to apply **Gale–Shapley Deferred Acceptance Algorithm** regarding the following preference list. There are three men and three women, with strict preference orders.

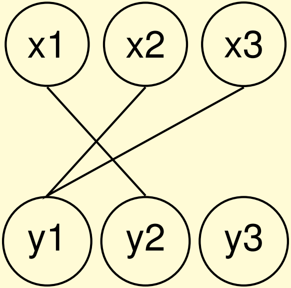
$$
\begin{array}{ll}
y_2 \succ_{x_1} y_1 \succ_{x_1} y_3 & \quad x_1 \succ_{y_1} x_3 \succ_{y_1} x_2 \\
y_1 \succ_{x_2} y_3 \succ_{x_2} y_2 & \quad x_3 \succ_{y_2} x_1 \succ_{y_2} x_2 \\
y_1 \succ_{x_3} y_2 \succ_{x_3} y_3 & \quad x_1 \succ_{y_3} x_3 \succ_{y_3} x_2
\end{array}
$$

Quiz3 - Step1

$y_2 \succ_{x_1} y_1 \succ_{x_1} y_3 \qquad x_1 \succ_{y_1} x_3 \succ_{y_1} x_2$

$y_1 \succ_{x_2} y_3 \succ_{x_2} y_2 \qquad x_3 \succ_{y_2} x_1 \succ_{y_2} x_2$

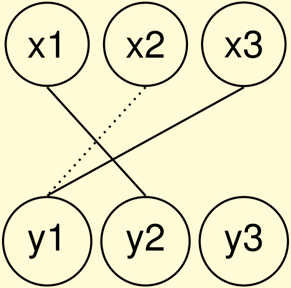
$y_1 \succ_{x_3} y_2 \succ_{x_3} y_3 \qquad x_1 \succ_{y_3} x_3 \succ_{y_3} x_2$



$y_1 : x_3, x_2$

Quiz3 - Step1

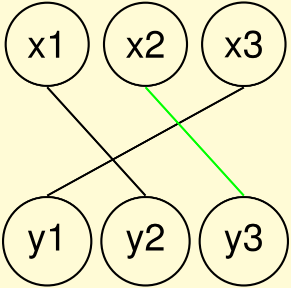
$$
\begin{array}{ll}
y_2 \succ_{x_1} y_1 \succ_{x_1} y_3 & \qquad x_1 \succ_{y_1} x_3 \succ_{y_1} x_2 \\
y_1 \succ_{x_2} y_3 \succ_{x_2} y_2 & \qquad x_3 \succ_{y_2} x_1 \succ_{y_2} x_2 \\
y_1 \succ_{x_3} y_2 \succ_{x_3} y_3 & \qquad x_1 \succ_{y_3} x_3 \succ_{y_3} x_2
\end{array}
$$



$y_1 : x_3 \; \cancel{x_2}$

Quiz3 - Step2

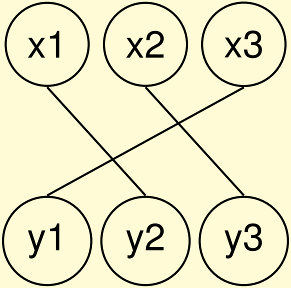
$$
\begin{array}{ll}
y_2 \succ_{x_1} y_1 \succ_{x_1} y_3 & x_1 \succ_{y_1} x_3 \succ_{y_1} x_2 \\
y_1 \succ_{x_2} y_3 \succ_{x_2} y_2 & x_3 \succ_{y_2} x_1 \succ_{y_2} x_2 \\
y_1 \succ_{x_3} y_2 \succ_{x_3} y_3 & x_1 \succ_{y_3} x_3 \succ_{y_3} x_2
\end{array}
$$



$x_2 : \cancel{y_1}\ y_3\ y_2$

Quiz3 - Step2

$$
\begin{array}{lll}
y_2 \succ_{x_1} y_1 \succ_{x_1} y_3 & \qquad & x_1 \succ_{y_1} x_3 \succ_{y_1} x_2 \\
y_1 \succ_{x_2} y_3 \succ_{x_2} y_2 & \qquad & x_3 \succ_{y_2} x_1 \succ_{y_2} x_2 \\
y_1 \succ_{x_3} y_2 \succ_{x_3} y_3 & \qquad & x_1 \succ_{y_3} x_3 \succ_{y_3} x_2
\end{array}
$$



$x_2 : \cancel{y_1}\ \cancel{y_3}\ y_2$

Stable now!

Proof of Correctness: Termination

Observation 1 Man proposes to woman in decreasing order of preference.

Observation 2 Once a woman is matched, she never becomes unmatched.

Observation 3 No agent proposes to another agent more than once,

Theorem

The algorithm terminates in at most N^2 iterations

Proof of Correctness: Termination

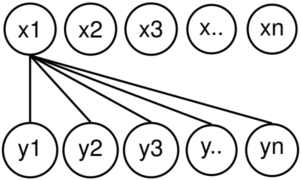
Theorem

The algorithm terminates in at most N^2 iterations

Proof.

- Each iteration, x proposes to y to whom he has not proposed before
- Proposal can be rejected.
- $|X| = |Y| = N$.
- $N * N$ possible proposals.

□



Proof of Correctness: Matching

Theorem

Everybody gets married

Proof.

- Once a woman is matched, she never becomes unmatched.
- Man is always rejected then be unmatched.
- Assume there exists a man that is not machted, then also a woman unmatched.
- Proved using a contradiction.

□

Proof of Correctness: Sability

Theorem

The algorithm produces a stable matching.

Proof.

- Assume there exists a blocking pair (x, y). So $x \succ_y M(y)$ and $y \succ_x M(x)$
- x must have proposed to y and y rejected x.
- y only reject x, when $M(y) \succ_y x$
- Proved using a contradiction.

$\square$

2012 Nobel Prize in Economics

Lloyd Shapley. Stable matching theory and Gale–Shapley algorithm.

> COLLEGE ADMISSIONS AND THE STABILITY OF MARRIAGE
>
> D. GALE* AND L. S. SHAPLEY, Brown University and the RAND Corporation
>
> **1. Introduction.** The problem with which we shall be concerned relates to the following typical situation: A college is considering a set of n applicants of which it can admit a quota of only q. Having evaluated their qualifications, the admissions office must decide which ones to admit. The procedure of offering admission only to the q best-qualified applicants will not generally be satisfactory, for it cannot be assumed that all who are offered admission will accept.

Alvin Roth. Applied Gale–Shapley to matching med-school students with hospitals, students with schools, and organ donors with patients.

Figure: Lloyd Shapley(left) and Alvin Roth(right)

Application: Matching Residents to Hospitals

- National Resident Matching Program (NRMP)[5].
- Original use just after WWII.
- Match med-school students to hospitals.
- Men $\approx$ hospitals, Women $\approx$ med school residents. But not same!
- Also called **The MATCH**.

[5] https://en.wikipedia.org/wiki/National_Resident_Matching_Program

Summary

Stable Marriage Problem

- $|X| = |Y| = N = |M|$
- $(x, y) \in M \equiv x = M(y) \wedge y = M(x)$.

Blocking Pair

- $x \succ_y M(y)$
- $y \succ_x M(x)$
- $\emptyset \succ_y M(y) \Rightarrow (y, \emptyset)$

Stable Marriage

No blocking pair for M.

GS DA Algorithm – $O(n^2)$

<u>Initialize</u> M to empty matching.
while *x is unmatched and hasn't proposed to every woman* **do**
    x proposes to the most-preferred and not proposed woman y;
    if *y is unmatched* **then**
        add (x, y) to M;
    else if *y prefers x to current partner x'* **then**
        Replace (x', y) with (x, y) in M;
    else
        y rejects x;

Summary

Stable Marriage Problem

- $|X| = |Y| = N = |M|$
- $(x, y) \in M \equiv x = M(y) \wedge y = M(x)$.

Blocking Pair

- $x \succ_y M(y)$
- $y \succ_x M(x)$
- $\emptyset \succ_y M(y) \Rightarrow (y, \emptyset)$

Stable Marriage

No blocking pair for M.

GS DA Algorithm – $O(n^2)$

<u>Initialize</u> M to empty matching.
while *x is unmatched and hasn't proposed to every woman* **do**
  x proposes to the most-preferred and not proposed woman y;
  if *y is unmatched* **then**
    add (x, y) to M;
  else if *y prefers x to current partner x'* **then**
    Replace (x', y) with (x, y) in M;
  else
    y rejects x;

Any Questions?