

Game Theory 2023

Ulle Endriss

Institute for Logic, Language and Computation

University of Amsterdam

Plan for Today

Matching markets are scenarios where agents have preferences over what other agent they get paired up with. Important applications:

- Matching junior doctors to hospitals
- Matching school children to schools
- Kidney exchange (different model from what we'll discuss)

Today is going to be an introduction to this topic, largely focusing on the basic scenario of *one-to-one* matching:

- *the stable matching problem* and the connection to *hedonic games*
- finding a *stable* solution with the *deferred-acceptance algorithm*
- various *extensions* of the model and other *properties* of solutions

A good general reference, somewhat emphasising algorithmic issues, is the book chapter by Klaus et al. (2016).

B. Klaus, D.F. Manlove, and F. Rossi. Matching under Preferences. In: *Handbook of Computational Social Choice*, Cambridge University Press, 2016.

Matching Problems

A *matching problem* is a tuple $\langle A, B, \succ^A, \succ^B \rangle$, where

- $A = \{1, \dots, n\}$ is a finite set of agents on the *left side* and $B = \{1, \dots, n\}$ is one on the *right side* of the market;
- $\succ^A = (\succ_1^A, \dots, \succ_n^A)$ is a profile of strict *preference orders* on B , one for each agent $i \in A$; and
- $\succ^B = (\succ_1^B, \dots, \succ_n^B)$ is a profile of strict *preference orders* on A , one for each agent $i \in B$.

Looking for one-to-one pairing of members of A with members of B .

Exercise: *What makes for a good solution (matching)?*

Example: Matching Job Seekers and Companies

100 people looking for a job and 100 companies offering one job each. Each of the job seekers '*on the left side of the market*' ranks all of the companies '*on the right side of the market*', and vice versa.

We want to find a way to pair them up that is stable:

No job seeker and company (representative) should want to ignore the proposed matching and instead prefer to arrange their own private employment contract.

Exercise: Can you always find such a *stable matching*?

Remark: In the original (Nobel-winning) paper and (still) much of the literature they instead speak of men and women getting married.

Aside: Embedding into Hedonic Games

We can think of $\langle A, B, \succ^A, \succ^B \rangle$ as a *hedonic game* $\langle N, \succeq \rangle$ with $N = \{1, \dots, 2n\}$, where the first n players represent the agents in A and the remaining n players those in B .

Each player ranks all coalitions consisting of herself and one player from the other side according to the matching problem, above all singleton coalitions, above all other coalitions.

Recall: Core membership and Nash/individual/contractual stability.

Exercise: Which of them corresponds to 'stability' in matching?

The Gale-Shapley Algorithm

Theorem 1 (Gale and Shapley, 1962) *There exists a stable matching for any combination of preferences of left and right agents.*

Proof: Consider the *deferred-acceptance algorithm* below.

- In each round, every left agent who is not yet matched proposes to her favourite right agent she has not proposed to before.
- In each round, every right agent picks her favourite amongst the proposals received *and* her current match (if any).
- Stop when everyone is matched to someone.

We observe: First, this always *terminates* with a complete matching. Second, that matching must be *stable*: for if not, that unhappy left agent would have proposed to that unhappy right agent ... ✓

D. Gale and L.S. Shapley. College Admissions and the Stability of Marriage. *American Mathematical Monthly*, 69:9–15, 1962.

Left-Optimal Matchings

A stable matching is *left-optimal* (*right-optimal*) if every agent on the *left* (*right*) likes it at least as much as any other stable matching.

Theorem 2 (Gale and Shapley, 1962) *The matching returned by the deferred-acceptance algorithm is left-optimal.*

Proof: We show that no left agent is ever rejected by an *achievable* partner (achievable = assigned to under some stable matching).

Proof by induction over rounds. Suppose so far nobody has been rejected by an achievable partner, but now b rejects a for $a' \succ_b a$.

We need to show that b is *not* achievable for a .

By induction hypothesis, $b \succ_{a'} b'$ for (other) achievable partners b' of a' (as otherwise a' would not want to propose to b at this point in time).

For the sake of contradiction, suppose there exists a stable matching with (a, b) and (a', b') for some b' . But this is blocked by (a', b) . ✓

Remark: One can also show that the outcome is always *right-pessimal*.

Fairness

Left-optimal matchings (returned by deferred-acceptance algorithm) arguably are not fair. But what is *fair*?

- One option is to implement the stable matching that *minimises* the *regret* of the agent worst off (regret = number of agents on the opposite side you prefer to your assigned partner).

Gusfield (1987) gives an algorithm for min-regret stable matchings.

- Similarly, we can implement the stable matching that maximises *average satisfaction* (i.e., that minimises average regret).

Irving et al. (1987) give an algorithm for this problem.

D. Gusfield. Three Fast Algorithms for Four Problems in Stable Marriage. *SIAM Journal of Computing*, 16(1):111–128, 1987.

R.W. Irving, P. Leather, and D. Gusfield. An Efficient Algorithm for the “Optimal” Stable Marriage. *Journal of the ACM*, 34(3):532–543, 1987.

Stable Matching under Incomplete Preferences

In an important generalisation of the basic matching problem, the agents are allowed to specify which agents of the other side they consider *acceptable*, and they only report a strict ranking for those.

- Now the assumption is that an agent would rather remain *alone* than get a partner they consider unacceptable.
- Now a matching is *stable* if no two agents want to get matched to each other rather than to their assigned partners *and* if nobody wants to leave her assigned partner and be alone instead.
- The *deferred-acceptance algorithm* can easily be extended to this setting: just require that proposers don't propose to unacceptable partners and proposees don't accept unacceptable offers.

This is called the matching problem with *incomplete preferences*.

Impossibility of Strategyproof Stable Matching

A matching mechanism is *strategyproof* if it never gives an agent (on either side of the market) an incentive to misrepresent her preferences.

Theorem 3 (Roth, 1982) *There exists **no** matching mechanism that is **stable** as well as **strategyproof** for both sides of the market.*

The proof on the next slide uses only two agents on each side, but it relies on a manipulation involving agents misrepresenting which partners they find *acceptable*. Alternative proofs, using three agents on each side, involve only changes in preference (not acceptability).

A.E. Roth. The Economics of Matching: Stability and Incentives. *Mathematics of Operations Research*, 7:617–628, 1982.

U. Endriss. Analysis of One-to-One Matching Mechanisms via SAT Solving: Impossibilities for Universal Axioms. Proc. AAAI-2020.

Proof

Suppose there are two agents on each side, with these preferences:

$$\begin{array}{lcl} a_1 : b_1 \succ b_2 & | & b_1 : a_2 \succ a_1 \\ a_2 : b_2 \succ b_1 & | & b_2 : a_1 \succ a_2 \end{array}$$

Two stable matchings: $\{(a_1, b_1), (a_2, b_2)\}$ and $\{(a_1, b_2), (a_2, b_1)\}$.

So any stable mechanism will have to pick one of them.

- Suppose the mechanism were to pick $\{(a_1, b_1), (a_2, b_2)\}$.
Then b_2 can pretend that she finds a_2 unacceptable, thereby making $\{(a_1, b_2), (a_2, b_1)\}$ the only stable matching.
- Suppose the mechanism were to pick $\{(a_1, b_2), (a_2, b_1)\}$.
Then a_1 can pretend that she finds b_2 unacceptable, thereby making $\{(a_1, b_1), (a_2, b_2)\}$ the only stable matching.

Hence, for any possible stable matching mechanism there is a situation where someone has an incentive to manipulate. ✓

Preferences with Ties

We can further generalise the matching problem by allowing for *ties*, i.e., by allowing each agent to have a weak preference order over the (acceptable) agents on the other side.

We can still *compute a stable matching* in polynomial time:

- arbitrarily break the ties (i.e, refine weak into strict orders)
- apply the standard deferred-acceptance algorithm

Now different stable matchings can *differ in size*:

$$\begin{array}{ll} a_1 : b_1 \mid b_2 & b_1 : a_1 \sim a_2 \\ a_2 : b_1 \succ b_2 & b_2 : a_2 \mid a_1 \end{array}$$

Both $\{(a_2, b_1)\}$ and $\{(a_1, b_1), (a_2, b_2)\}$ are stable.

Finding a *maximal* stable matching is NP-hard (Manlove et al., 2002).

D.F. Manlove, R.W. Irving, K. Iwama, S. Miyazaki, and Y. Morita. Hard Variants of Stable Marriage. *Theoretical Computer Science*, 276(1–2):261–279, 2002.

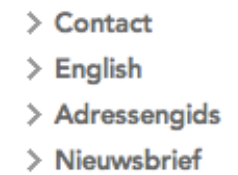
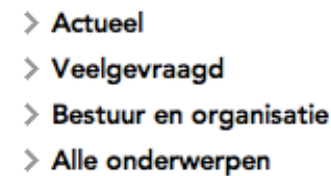
Variants

Variants and generalisations are applicable to many scenarios:

- *Residents-Hospitals Problem*
 - Matching of junior doctors (residents) to hospitals.
 - Many-to-one variant of matching problem with incomplete preferences, with each hospital having a certain capacity.
- *School Choice*
 - Matching of school children to schools.
 - Similar, but schools have *priorities* rather than preferences (distance to home, sibling already at school, etc.).

Main difference is interpretational: schools are not economic agents.

When hospitals/schools have weak preferences/priorities, we need to find a way to *break ties* when capacity limits are reached.



[Home](#) > [Onderwijs & jeugd](#) > [Voortgezet onderwijs](#) > [Centrale matching](#)

[Zoeken...](#)

Centrale matching

9 december 2014

In het kort

Informatie over matching, de nieuwe methode van toewijzen van leerlingen aan VO-scholen.

Centrale Matching

Dit jaar gaat het toewijzen van leerlingen aan VO-scholen anders dan in de jaren daarvoor. Amsterdam gaat gebruik maken van het systeem 'centrale matching'.

Op deze pagina

- > [Centrale Matching](#)
- > [Hoe werkt matching?](#)
- > [Matching: tijdspad](#)
- > [Matching: Deferred Acceptance](#)

Hoe werkt matching?

1. Je krijgt voor elke school in het voortgezet onderwijs in Amsterdam een lotingnummer.
2. Je wordt tijdelijk geplaatst op de school die het hoogst op je voorkeurslijst staat.
3. Elke school kijkt of er genoeg plek is voor alle leerlingen die tijdelijk bij de school zijn geplaatst. Als dit het geval is, dan houdt de school al deze leerlingen vast. Als dat niet het geval is, dan houdt de school de leerlingen vast die voorrang hebben. De lotingnummers voor de school worden gebruikt om te bepalen welke leerlingen niet op de school geplaatst kunnen worden.

Case Study: Amsterdam School Choice

System used prior to 2015 (“adaptive Boston mechanism”):

- Use lottery to rank all children. Use ranking to refine every school’s priority list into a strict order.
- Ask children to announce their top choices. Award top choices subject to capacity constraints and following refined priority lists. Remove matched children and places from system. Repeat.

System introduced for 2015 (DA with local tie-breaking):

- Refine priorities as above, but use separate lottery for each school.
- Use many-to-one variant of deferred-acceptance algorithm.

System introduced for 2016 (DA with global tie-breaking):

- Same, but use just a single lottery for all schools.

Original system not stable or (child-)strategyproof. New systems are. Local tie-breaking fairer, but less efficient (more swaps desired).

Summary: Matching Markets

We have seen several *variants* of the basic matching problem:

- basic matching problem, extension to incomplete preferences, extension to preferences with ties
- we have hinted at possible extensions to many-to-one variants

We have discussed various desirable *properties* of matchings:

- stability: no pair has an incentive to break the matching
- efficiency: no mutually beneficial swaps possible after assignment
- strategyproofness for one side of the market: no incentive to lie
- strategyproofness for both sides: incompatible with stability
- fairness: possibly expressed in terms of 'regret'
- maximality (in terms of cardinality): computationally intractable

We have seen how the deferred-acceptance algorithm of Gale and Shapley can be used to compute stable matchings efficiently.

Course Review

This has been an introduction to game theory, covering these topics:

- strategic games in normal form and Bayesian games
- strategic games in extensive form and imperfect-information games
- auctions and mechanism design (“inverse game theory”)
- coalitional games: TU / NTU / hedonic games and matching

In a strategic game a *solution* is a profile of strategies. In a coalitional game it is a coalition structure and a choice of who gets what.

A *solution concept* suggests what solutions will emerge in a game.

We have focused on *stability*, less so on efficiency and fairness.

We have hardly spoken about *applications* of game theory in other disciplines, but they are there (that’s why the field is so successful) and you should try to discover those most relevant to you . . .