BEEM147: Topics in Microeconomic Theory II
Matching and Market Design

Alejandro Robinson-Cortés
University of Exeter

Lecture Notes
Spring Term 2021

Contents

Disclaimer. The bulk of these lecture notes is based on material I first learned on several classes at Caltech, mainly: Federico Echenique's SS201c, Leeat Yariv's SS211c, and Luciano Pomatto's Ec117. They also draw from the classic textbook on two-sided matching by Roth and Sotomayor (1990), and the more recent one by Haeringer (2017) on market design. The lecture slides of Muriel Niederle and Nicole Immorlica, both publicly available online, have also been an invaluable resource. I have tried to cite original sources and provide additional references whenever possible. The notes are work in progress. All errors are my own. Please let me know if you find any mistakes or missing citations: a.robinson-cortes@exeter.ac.uk.

The warning symbol on the left indicates that a subsection could use some work. At times, I have opted to discuss a reference or result without much detail, rather than omitting it.

1 House allocation

Consider the following allocation problem: how to assign a set of distinct objects to agents with heterogeneous preferences? For concreteness, in what follows we shall refer to the objects as *houses*. However, the framework will be general in that it will apply to any set of objects you may think of besides houses. The main assumption is that the objects are indivisible.

1.1 Preliminaries: preferences and utility

A **binary relation** R over a set X is defined as a subset of $X \times X$. If $(x, x') \in R$, we say that x and x' are related by R or R-related, and write xRx'. Similarly, if $(x, x') \notin R$, we write not xRx'. Binary relations can be used for a wide range of purposes. For example, the order "less than or equal to" $\leq$ defined over the real numbers $\mathbb{R}$ is a binary relation. For our purposes, binary relations will serve us to describe agents' preferences over houses.

Let I be a finite set of **agents** and H a finite set of **houses**. An agent's preferences for houses are summarized by their preference relation, which is a binary relation over H. Denote the preference relation of agent $i \in I$ by $\succcurlyeq_i$, where $h \succcurlyeq_i h'$ stands for agent i preferring house h at least as much as they prefer house h'.

Definition 1.1. *A binary relation R over a set X is (i) **complete** if for every $x, x' \in X$, xRx', $x'Rx$, or both; (ii) **transitive** if for every $x, x', x'' \in X$, xRx' and $x'Rx''$ imply xRx''; (iii) **antisymmetric** if for every $x, x' \in X$, xRx' and $x'Rx$ imply $x = x'$.*

Definition 1.2. *A **preference relation** is a complete and transitive binary relation. The set of preference relations over a set H is denoted by $\mathcal{R}(H)$. A preference relation is said to be **strict** if it is antisymmetric. The set of strict preference relations over H, also known as **linear orders**, is denoted by $\mathcal{P}(H)$.*

Requiring agents to have complete preferences amounts to assuming they are able to compare any pair of houses in H, say h and h', and have one of three opinions: (i) *I like house h more than house h'*, i.e., $h \succcurlyeq h'$ and not $h' \succcurlyeq h$; (ii) *I like h less than h'*, i.e., $h' \succcurlyeq h$ and not $h \succcurlyeq h'$, or (iii) *I am indifferent between them*, i.e., $h \succcurlyeq h'$ and $h' \succcurlyeq h$. Given $\succcurlyeq \in \mathcal{R}(H)$, define the auxiliary binary relations $\succ$ and $\sim$ as follows. For $h, h' \in H$,

- $h \succ h' \iff h \succcurlyeq h'$ and not $h' \succcurlyeq h$;
- $h \sim h' \iff h \succcurlyeq h'$ and $h' \succcurlyeq h$.

Oftentimes, $\succ$ is referred to as the **strict part** of $\succcurlyeq$ (which in turn is referred to as its **weak counterpart**), and $\sim$ is called the **indifference relation**.

The transitivity requirement in Definition 1.2 imposes consistency within an agent's preferences. It precludes agents from preferring house h_1 to h_2, and h_2 to h_3, while at the same time preferring house h_3 to h_1. Also, note that strict preferences rule out indifference. That is, assuming that an agent has a strict preference relation means that they are able to rank all houses, from their favorite to their least favorite one, without being indifferent between any of them. Since $h \succcurlyeq h'$ and $h \neq h'$ if and only if $h \succ h'$ for $\succcurlyeq \in \mathcal{P}(H)$, at times we may denote strict preferences simply by $\succ$.

Exercise 1.3. Prove that, for $\succcurlyeq \in \mathcal{P}(H)$, $h \succcurlyeq h'$ and $h \neq h'$ if and only if $h \succ h'$.

Exercise 1.4. Using the definitions above, for both $\succcurlyeq \in \mathcal{R}(H)$ and $\succcurlyeq \in \mathcal{P}(H)$, evaluate the following[1]: (1) $\succ$ is complete; (2) $\succ$ is transitive; (3) $\succ$ is antisymmetric; (4) $\sim$ is complete; (5) $\sim$ is transitive; (6) $\sim$ is antisymmetric. (Note that you must evaluate a total of 12 statements.)

Besides its intuitive appeal, one of the main advantages of assuming that agents have complete and transitive preferences is that we can represent them numerically.

Proposition 1.5. *A binary relation $\succcurlyeq$ over H is a preference relation if and only if it admits a **utility representation**, i.e., if there exists a function $U : H \to \mathbb{R}$ such that:*

$$h \succcurlyeq h' \quad \iff \quad U(h) \geq U(h') \quad \forall\, h, h' \in H.$$

Exercise 1.6. Prove Proposition 1.5.

Note that the interpretation of a utility function is quite limited. In particular, even though it is tempting to interpret utility differences in terms of "preference intensity," this is not correct. That is, the magnitude of the difference $U(h) - U(h')$ does not contain any information regarding the preference over h and h'. Only its sign matters: $U(h) - U(h') > 0 \Leftrightarrow h \succ h'$; $U(h) - U(h') < 0 \Leftrightarrow h' \succ h$, and

[1] To "evaluate" a statement means to state whether it is true, in which case a proof must be provided, or false, in which case a counterexample must be provided.

$U(h) = U(h') \Leftrightarrow h \sim h'$. This is because, by definition, preference relations are *ordinal*; they do not contain any *cardinal* information about an agent's preferences.

Exercise 1.7. Let $\succcurlyeq$ be a preference relation over H, and assume that U is a utility function that represents $\succcurlyeq$. Show that $V : H \to R$ also represents $\succcurlyeq$ if there exists an increasing function $f : \mathbb{R} \to \mathbb{R}$ such that $V = f \circ U$.

Exercise 1.8. In the real world, it is natural to think that the more someone likes a house, the more they would be willing to pay for it. In this sense, the willingness to pay for an object is a cardinal measure of its desirability. Let $WTP(h)$ denote the willingness to pay for house $h \in H$. On the one hand, WTP keeps the ordinal "flavor" of a utility function: an agent "'likes" h at least as house h' if $WTP(h) \geq WTP(h')$. On the other hand, we could also measure "preference intensity" by comparing the differences in WTP: say that an agent "likes much more" house h to h' than what they "like" house h' compared to h'' if $WTP(h) - WTP(h') > WTP(h') - WTP(h'')$. How would you alter our current framework, stated in terms of (ordinal) preferences relations, to incorporate a cardinal desirability measure such as the WTP? In particular, what would you add? What are the advantages and disadvantages of modeling the "intensity" of preferences?

1.2 House allocation problems

A **house-allocation problem** is defined by a tuple $(I, H, (\succcurlyeq_i)_{i \in I})$, where I is a nonempty finite set of agents, H a nonempty finite set of houses, and $(\succcurlyeq_i)_{i \in I}$ a preference profile. Assume that each agent has strict preferences, $\succcurlyeq_i \in \mathcal{P}(H)$ for every $i \in I$. An allocation of houses is described by a matching.

Definition 1.9. *A **matching** is a function $\mu : I \cup H \to I \cup H \cup \{\emptyset\}$ such that, for every $i \in I$ and $h \in H$, (i) $\mu(i) \in H \cup \{\emptyset\}$; (ii) $\mu(h) \in I \cup \{\emptyset\}$; (iii) $\mu(i) = h$ if and only if $\mu(h) = i$. Denote the set of all matchings between agents and houses by $\mathcal{M}(I, H)$.*

A matching specifies which agent is assigned to which house. If $\mu(i) = h$ then house h is assigned to agent i, which is equivalent to $\mu(h) = i$. Note that we allow for agents to remain unmatched by "matching" them with the empty set $\mu(i) = \emptyset$. Similarly, we allow for houses to remain unassigned, which we specify by $\mu(h) = \emptyset$.

Definition 1.10. *Given a preference profile $(\succcurlyeq_i)_{i\in I}$, a matching μ is **Pareto efficient** if there is no other matching $\nu \in \mathcal{M}(I,H)$ such that $\nu(i) \succcurlyeq_i \mu(i)$ for all $i \in I$ and $\nu(j) \succ_j \mu(j)$ for at least one $j \in I$.*

Pareto efficiency is perhaps the weakest notion of "optimality" of a matching. If a matching is *not* Pareto efficient, it means that we can find another matching in which everyone is at least as well off and at least someone is strictly better off. In this sense, it seems hard to justify that a matching is desirable if it is not Pareto efficient.

Example 1.11. *Let $I = \{1,2,3,4\}$ and $H = \{a,b,c,d\}$. The preferences are given by:*

$$\succ_1 : b,c,d,a; \quad \succ_2 : a,b,c,d; \quad \succ_3 : a,c,d,b; \quad \succ_4 : a,d,b,c.$$

That is to say, agent 1 prefers house b over all the houses, followed by house c, house d, etc. Consider the matching μ given by:

$$\mu(1) = d, \quad \mu(2) = a, \quad \mu(3) = c, \quad \mu(4) = b.$$

*Is matching μ Pareto efficient? No, since agents 1 and 4 can trade their houses and be (strictly) better off. In this sense, we say that the resulting matching μ' **Pareto dominates** matching μ, where μ' is given by $\mu'(1) = b$, $\mu'(2) = a$, $\mu'(3) = c$, and $\mu'(4) = d$. Question: is matching μ' Pareto efficient?*

1.3 Review of mechanism design

An **environment** is a pair (I,X), where I is a finite set of agents with $|I| = n$, and X a finite set of possible outcomes, e.g., $X = \mathcal{M}(I,H)$. Each agent $i \in I$ ranks outcomes in X according to the linear order $\succcurlyeq_i \in \mathcal{P}(X)$. A **social choice function** is a function $f : \mathcal{P}(X)^n \to X$, mapping preference profiles $(\succcurlyeq_i)_{i\in I}$ to outcomes. A **mechanism** is a tuple $(M_1,\ldots,M_n,g)$, in which each M_i is a nonempty **set of messages** for agent $i \in I$, and $g : \times_{i\in I} M_i \to X$ maps profiles of messages to outcomes. Given a mechanism $(M_1,\ldots,M_n,g)$, a **strategy** for an agent i is a function $s_i : \mathcal{P}(X) \to M_i$, mapping preferences to messages.

A **strategy profile** is a collection of strategies, one for each player, which we denote as

$$s = (s_1,\ldots,s_n) = (s_i, s_{-i}).$$

A strategy s_i is a **dominant strategy** for agent i if, for all $\succcurlyeq_i \in \mathcal{P}(X)$,

$$g(s_i(\succcurlyeq_i), m_{-i}) \succcurlyeq_i g(m_i', m_{-i})$$

for all $m_i' \in M_i$ and all $m_{-i} \in M_{-i}$. That is, a strategy is dominant if an agent always finds it optimal to choose the message prescribed by the strategy, whatever their preferences and the messages chosen by others. A strategy profile $s = (s_1, \ldots, s_n)$ is a **dominant strategy equilibrium** if s_i is a dominant strategy for every $i \in I$.

A social choice function f is **dominant strategy implementable** if there exists a mechanism $(M_1, \ldots, M_n, g)$ with a dominant strategy equilibrium $s = (s_1, \ldots, s_n)$ such that, for every preference profile $(\succcurlyeq_i)_{i \in I} \in \mathcal{P}(X)^n$, the mechanism results in the same outcome as the social choice function, i.e.,

$$g[(s_i(\succcurlyeq_i))_{i \in I}] = f[(\succcurlyeq_i)_{i \in I}].$$

Given a social choice function f, the **direct revelation mechanism** associated with f is defined by setting each $M_i = \mathcal{P}(X)$ and $g = f$. That is, in a direct revelation mechanisms agents report preferences and the mechanism chooses whatever outcome is dictated by the social choice function.

Definition 1.12. *A social choice function f is **strategy-proof** if the strategy profile in which everyone reports their true preference, i.e., $s_i(\succcurlyeq_i) = \succcurlyeq_i$ for every $i \in I$, is a dominant strategy equilibrium in the direct revelation mechanism associated with f. In other words, f is strategy-proof if, for all $\succcurlyeq = (\succcurlyeq_i, \succcurlyeq_{-i}) \in \mathcal{P}(X)$,*

$$f(\succcurlyeq_i, \succcurlyeq_{-i}) \succcurlyeq_i f(\succcurlyeq_i', \succcurlyeq_{-i}),$$

for all $\succcurlyeq_i' \in \mathcal{P}(X)$ and all $i \in I$.

Theorem 1.13 (Revelation Principle)**.** *A social choice function is dominant strategy implementable if and only if it is strategy-proof.*

The importance of the revelation principle is that it allows us to focus on direct revelation mechanisms when studying the implementability of social choice functions. Furthermore, the property of strategy-proofness is very intuitive when thought of in terms of incentives. Simply put, social choice functions (or mechanisms for that matter) are strategy-proof if individuals find it optimal to report their

true preferences. Strategy-proof mechanisms cannot be "gamed", or, more precisely, in strategy-proof mechanisms agents have no incentives to misreport the truth.

When studying house allocation problems, we shall study **direct revelation matching mechanisms** of the form $\phi : \mathcal{P}(H)^n \to \mathcal{M}(H, I)$, where $n = |I|$ is the number of agents. That is, matching mechanisms map preference profiles to matchings between agents and houses. We shall say that a mechanism is Pareto efficient if it always generates a Pareto efficient matching. Similarly, that it is strategy-proof if all agents always find it profitable to report their true preferences. Denote a profile of preferences $(\succcurlyeq_i)_{i\in I} \in \mathcal{P}(H)^n$ simply by $(\succcurlyeq_i)$.

Definition 1.14. *A **matching mechanism** $\phi : \mathcal{P}(H)^n \to \mathcal{M}(H, I)$ is **Pareto efficient** if, for every preference profile $(\succcurlyeq_i) \in \mathcal{P}(H)^n$, the matching $\phi[(\succcurlyeq_i)]$ is Pareto efficient. It is **strategy-proof** if, for every preference profile $(\succcurlyeq_i) \in \mathcal{P}(H)^n$ and every agent $i \in I$,*

$$\phi[(\succcurlyeq_i, \succcurlyeq_{-i})](i) \succcurlyeq_i \phi[(\succcurlyeq', \succcurlyeq_{-i})](i) \qquad \text{for every } \succcurlyeq' \in \mathcal{P}(H).$$

Notice that while both Pareto efficiency and strategy-proofness are properties of a mechanism, the former does not rely on any assumption on agents' behavior. That is, to determine whether a mechanism is Pareto efficiency or not, we do not need to assume that agents are optimizing. By contrast, the notion of strategy-proofness relies on the notion of optimizing agents. To illustrate, consider the following example.

Example 1.15. *Consider the follow mechanism. Order the agents according to some arbitrary order: i_1, i_2, etc. Given a preference profile $(\succcurlyeq_i)$, assign each agent to the house they rank as their second most preferred house. If two agents rank the same house as their second choice, assign it to the agent who goes first in the order. Assign unassigned agents to their third choice and break ties in the same way. Continue in the same way (with the fourth choice, and so on) until all agents or all houses have been assigned. Question: is this mechanism Pareto efficient?*

The mechanism is clearly not Pareto efficient. To see this formally, consider a simple counterexample: two agents, i_1 and i_2, two houses, h and h', and preferences given by $h \succ_{i_1} h'$ and $h' \succ_{i_2} h$. Under this profile of preferences, the mechanism assign the matching $\mu(i_1) = h'$ and $\mu(i_2) = h$, which is clearly not Pareto efficient since both agents would be better off by trading houses.

A keen observer might object by noting that agent i_1 would clearly not report their true preferences to the mechanism in this setting. Given that the other agent is reporting $h' \succ_{i_2} h$, i_1 could be assigned to their top choice by reporting the alternative preference relation: $h' \succ'_{i_1} h$. In such case, the mechanism would assign a matching that is Pareto efficient under the true preference profile, the one given by $\mu'(i_1) = h$ and $\mu'(i_2) = h'$. However, while this argument shows that the mechanism is not strategy-proof (since i_1 would rather lie and misreport their preferences), it is not the correct way to check whether the mechanism is Pareto efficient or not.

The definition of Pareto efficiency for a matching (see Definition 1.10) takes a preference profile as given, and Definition 1.14 states that a mechanism is Pareto efficient if its output is Pareto efficient at every preference profile. This means that to determine whether a mechanism is Pareto efficient or not, we take the preference profile as given, and do not consider whether it is the "true" one. Pareto efficiency is a notion that does not depend on agents behavior. Contrast this with the definition of strategy-proofness, which explicitly requires comparing the outcomes of a mechanism across distinct preference profiles, one of which is assumed to be the "true" one.

Exercise 1.16. Consider the same mechanism as in Example 1.15 with the difference that agents are assigned their top choice (ties are broken in the same way, according to the predetermined order, and assigning agents to their next choices). Evaluate whether this mechanism is (i) Pareto efficient and (ii) strategy-proof.

1.4 Serial dictatorship

The question we tackle now is whether Pareto efficient matchings exist in house-allocation problems, and, if so, how to find them. To do so, we introduce our first allocation mechanism, which takes the form of an algorithm. Define a **priority order** of agents as a one-to-one and onto function $\pi : \{1, \dots, n\} \to I$, where $\pi(k)$ denotes the agent in the k-th spot.

Algorithm 1.17 (Serial Dictatorship)**.** *Given a preference profile $(\succcurlyeq_i)_{i \in I} \in \mathcal{P}(H)^n$ and a priority order π, proceed in steps as follows. Initially, all houses are available.*

- *In the k-th step, assign agent $\pi(k)$ to their top choice from the set of available houses. Remove the newly assigned house from the set of available houses. Proceed until every agent has been assigned to a house or there are no more available houses.*

In other words, the Serial Dictatorship (SD) algorithm assigns the agent with the highest priority to their favorite house, the agent with the second-highest priority to their favorite house among the remaining ones, and so on, until there is no house left or all agents have been assigned to a house. Clearly, the algorithm is not "fair" in that it favors agents with higher priority. A common practice to circumvent this issue is to assign priorities randomly. We shall study this "random" version later on in more detail. Nonetheless, the most attractive feature of SD is that it always generates a matching that is Pareto efficient. And not only that, it actually characterizes the set of Pareto efficient matchings.

Proposition 1.18. *A matching μ is Pareto efficient if and only if there exists a priority order π such that μ is the matching generated by the Serial Dictatorship algorithm under π.*

Proof. ($\Leftarrow$) First we show that the outcome of SD is Pareto efficient. Proceed by contradiction. Let $\mu \in \mathcal{M}(I, H)$ be the outcome of SD, and assume there exists $\nu \in \mathcal{M}(I, H)$ such that $\nu(i) \succcurlyeq_i \mu(i)$ for all $i \in I$ and $\nu(j) \succ_j \mu(j)$ for some $j \in I$. To simplify notation, wlog, assume $\pi(i) = i$ so that agent k chooses in the k-th step. Since agent 1 is getting their top choice under μ, $\nu(1) = \mu(1)$. Now consider agent 2. For a moment, suppose that $\nu(2) \succ_2 \mu(2)$. This could only happen if house $\nu(2)$ was not available in the second round for agent 2, i.e., if $\nu(2) = \mu(1)$. But this would be a contradiction since $\mu(1) = \nu(1)$; hence, $\nu(2) = \mu(2)$. The proof then follows by induction showing that $\nu(k) = \mu(k)$ for every $k \geq 3$ in the same fashion. And, hence, we reach a contradiction, implying that μ must be Pareto efficient.

($\Rightarrow$) Let μ be a Pareto efficient matching. To show that μ is the outcome of a SD for some priority order π, first, we claim that under μ some agent must be getting their top choice. Suppose not. Then let each agent point to their top choice and let each house point to its owner under μ. This must lead to a cycle since the number of agents is finite (why?). Move every agent in the cycle to the house they are pointing to. This new allocation Pareto dominates μ, which is a contradiction. Hence, order the $m \geq 1$ agents that are getting their top choice in μ as $\pi(1), \pi(2), \ldots, \pi(m)$. Repeat the same argument with the remaining $n - m$ agents and the houses that are not owned by any of the first m agents. Continue in the same fashion until every agent has been assigned a priority order. Note that, by construction, μ is the resulting matching of the SD under π. Q.E.D.

Example 1.19. *Consider the same house-allocation problem as in Example 1.11. Let us compute the outcome of SD with the priority order $\pi(i) = i$ for every $i = 1, \dots, 4$.*

1. *In the first step, agent 1 is assigned to house b.*
2. *In the second step, agent 2 to house a.*
3. *In the third step, agent 3 to house c (since house a is no longer available).*
4. *In the final step, agent 4 is assigned to house d (since house a is no longer available).*

Note that the resulting matching is μ' as described in Example 1.11. Therefore, by Proposition 1.18, we conclude that μ' is Pareto efficient.

Exercise 1.20. How many Pareto efficient matchings are there in the house allocation problem in Example 1.11? How many priority orders are there?

Exercise 1.21. Let $|I| = |H| = n$. A preference profile $(\succcurlyeq_i)_{i \in I}$ is "exact" if the number of Pareto efficient matchings under $(\succcurlyeq_i)_{i \in I}$ is the same as the number of priority orders on I. How many "exact" preference profiles are there in $\mathcal{P}(H)^n$?

Exercise 1.22. Let $|I| = |H| = n$. How many preference profiles with a unique Pareto efficient matching are there in $\mathcal{P}(H)^n$?

According to Proposition 1.18, SD always generates matchings that are Pareto efficient. However, the proposition assumes that we know the preferences of the agents; otherwise, how would we be able to run the algorithm. In real life, an outside observer generally does not know the preferences of the participating agents. Therefore, now we tackle the question of whether an agent would voluntarily report their true preferences when participating in a SD. In other words, is Serial Dictatorship strategy-proof?

Proposition 1.23. *The Serial Dictatorship mechanism is strategy-proof.*

Proof. Fix $j \in I$. Denote a profile of preferences $(\succcurlyeq_i)_{i \in I}$ simply by $(\succcurlyeq_i)$. Furthermore, let $\phi[(\succcurlyeq_i)] \in \mathcal{M}(I, H)$ be the matching generated by SD. Given a preference profile $(\succcurlyeq_i)$, let $C_j[(\succcurlyeq_i)]$ be the set of houses available to j in their turn of the SD-algorithm. That is, $C_j[(\succcurlyeq_i)]$ is the set from which j chooses their top choice. Crucially, note that the set $C_j[(\succcurlyeq_i)]$ depends on the preference of the agents who have

a higher priority than j (the ones choosing before), but it does not depend on j's own preferences. Hence, we may write $C_j[(\succcurlyeq_{-j})]$. The preferences reported by j will only affect the house to which they are assigned to from $C_j[(\succcurlyeq_{-j})]$, which implies that the mechanism is strategy-proof since j cannot do better than to report their true preference. Q.E.D.

Notes

The material in this section is well-known and standard in the literature. For a standard treatment of preference relations and their use in microeconomics, see Mas-Colell, Whinston, and Green (1995). For a more thorough decision-theoretic treatment, see Kreps (1988). For a standard treatment on mechanism design, including the revelation principle, see Diamantaras et al. (2009). For a standard treatment of the Serial Dictatorship algorithm and house allocation problems, see Chapter 11 of Haeringer (2017).

Additional exercises

Exercise 1.24. Give 20 examples of real-life allocation problems that can be modeled as house-allocation problems.

Exercise 1.25. Give an example of a mechanism for the house-allocation problem that (i) is not Pareto efficient or strategy-proof; (ii) is Pareto efficient but not strategy-proof; (iii) is not Pareto efficient but is strategy-proof. *Note*: for each of cases (i)–(iii) you need to provide the corresponding proof or counterexample.

Exercise 1.26. Modify the definition of a house allocation problem by allowing each binary relation $\succcurlyeq_i$ to be a general preference relation, not necessarily a linear order.

(a) In which of the 20 examples you provided in Exercise 1.24 would it be reasonable to expect agents to be indifferent among objects? Why?

(b) How would you modify the Serial Dictatorship mechanism? (Since preferences might have ties, the mechanism, as described in Algorithm 1.17, is not well defined).

(c) Notice that the definition of Pareto Efficiency can be applied, without modification, to preference relations. Can you modify the Serial Dictatorship mechanism in such a way that the resulting mechanism is Pareto efficient?

Exercise 1.27. So far, we have assumed that agents only care about the house they are assigned to; they have no preferences for the houses of other agents. In real life this may not be the case: someone could like the house they are assigned to inasmuch it is better (or worse) than the houses assigned to their peers.

(a) In which of the 20 examples you provided in Exercise 1.24 would it be sensible to assume that agents may have preferences for the houses occupied by other agents? Why?

(b) Show via counterexample that, in this setting, the Serial Dictatorship mechanism may fail to generate a Pareto efficient matching. Is it strategy-proof?

Exercise 1.28. Can you find a direct matching mechanism ϕ that is Pareto efficient, strategy-proof, and different to the Serial Dictatorship mechanism? Formally, let ϕ_{SD} be the Serial Dictatorship mechanism. There must exist a profile of preferences $(\succcurlyeq_a)$ such that the two matchings $\phi[(\succcurlyeq_a)]$ and $\phi_{SD}[(\succcurlyeq_a)]$ are different.

2 Housing market

A **housing market** is the same as a house-allocation problem with the difference that each agent is assumed to own a house initially. For simplicity, assume $|I| = |H| = n$. Label the houses in H as $h_1, h_2, \ldots, h_n$ so that h_i is the house owned by agent $i \in I$ at the outset.[2] This model allows us to formalize the basic idea of an exchange economy. That is, the question is how to mediate a voluntary exchange of houses among agents in which everyone is made better off. The next two examples show that, even though it always generates Pareto efficient outcomes, Serial Dictatorship fails to deliver sensible outcomes in the presence of property rights

Example 2.1. *Consider the same setup as in Example 1.11 with the difference that each agent owns a house initially. Recall, $I = \{1, 2, 3, 4\}$ and $H = \{a, b, c, d\}$, with preferences given by:*

$$\succ_1 : b, c, d, \underline{a}; \quad \succ_2 : a, \underline{b}, c, d; \quad \succ_3 : a, \underline{c}, d, b; \quad \succ_4 : a, \underline{d}, b, c,$$

where we have underlined the house initially owned by each agent. That is, agent 1 owns house a, agent 2 owns house b, and so on. Note that the initial allocation is in itself a matching. In this case, it is not Pareto efficient since agents 1 and 2 would be strictly better off by exchanging their houses. That is, agent 1 would rather have b instead of a, and agent 2 house a instead of b. From Example 1.19, we know that the resulting matching is Pareto efficient (since it is the outcome of SD with $\pi(i) = i$ for $i = 1, \ldots, 4$.).

However, running SD might not always generate sensible outcomes. Consider the priority order $\pi = \{3, 1, 2, 4\}$. It can easily be verified that running SS with π results in the following matching:

$$\mu(1) = b, \quad \mu(2) = c, \quad \mu(3) = a, \quad \mu(4) = d.$$

By Proposition 1.18, we know that μ is Pareto efficient. Nonetheless, note that agent 2 has no incentives to participate in this mechanism since they are left worse off. That is, while SD assigns house c to agent 2, they would rather stay with house b.

[2] At times, we shall abuse notation and label agents i_k, for $k = 1, 2, \ldots, n$, in which case we denote the house owned by agent i_k simply by h_k.

Example 2.2. *Let* $I = \{1, 2, 3\}$ *and* $H = \{a, b, c\}$ *with preferences given by:*

$$\succ_1 \colon b, c, \underline{a}; \quad \succ_2 \colon a, \underline{b}, c; \quad \succ_3 \colon a, b, \underline{c}.$$

Consider the matching

$$\mu(1) = c, \quad \mu(2) = b, \quad \mu(3) = a.$$

It can easily be verified that μ *is Pareto efficient and assigns each agent a house they prefer at least as much as the one they initially own. Indeed, note that* μ *is the outcome of SD with* $\pi = \{3, 2, 1\}$*. However, note that agents 1 and 2 would be better off by not participating in the mechanism and trading their initial houses amongst themselves. That is, agent 1 would rather have house* b *(in exchange for* a*) than getting* c *in* μ*, and agent 2 would rather have house* a *(in exchange for* b*) than keeping house* b *in* μ*.*

2.1 Individual rationality and the core

Individual rationality captures the idea of property rights. A matching is individually rational if it assigns to each agent a house that they find at least as good as the one they already own.

Definition 2.3. *Given a preference profile* $(\succcurlyeq_i)_{i \in I}$*, a matching* $\mu \in \mathcal{M}(I, H)$ *is* ***individually rational*** *if* $\mu(i) \succcurlyeq_i h_i$ *for every* $i \in I$*. A mechanism is individually rational if it always produces matchings that are individually rational.*

Note that in Example 2.1 matching μ is not individually rational even though it is Pareto efficient. Such example illustrates why Pareto efficiency does not assure that individuals will end up with a house at least as good as the one they own when they enter an exchange. The definition of Pareto efficiency does not take into account that houses are initially owned by agents (while individual rationality does). Nevertheless, Example 2.2 shows that individual rationality does not guarantee that *groups* of individuals will want to participate in an exchange.

Definition 2.4. *A matching* $\mu \in \mathcal{M}(I, H)$ *is* ***blocked*** *by a group of agents* $A \subseteq I$*, also called a* ***coalition****, if there exists another matching* $\nu \in \mathcal{M}(I, H)$ *such that* (i) *for all* $a \in A$*,* $\nu(a)$ *is initially owned by someone in* A*, and* (ii) $\nu(a) \succcurlyeq_a \mu(a)$ *for all* $a \in A$*, and* $\nu(a) \succ_a \mu(a)$ *for some* $a \in A$*.*

The notion of *blocking* captures the idea that a group of agents (a.k.a. a coalition) may reject an allocation because they find it mutually beneficial to trade on the sidelines amongst themselves. Hence, the property of not being blocked is desirable inasmuch as we wish agents to trade voluntarily.

Definition 2.5. *The **core** of a housing market is the set of all matchings that are not blocked by any coalition.*

Proposition 2.6. *Every matching in the core is individually rational and Pareto efficient.*

Proof. Let μ be a matching in the core. Note that μ must be individually rational, since otherwise there would exist an agent $i \in I$ such that the coalition $A = \{i\}$ blocks μ. Similarly, if it were not Pareto efficient, then the coalition $A = I$ would block μ. Q.E.D.

Proposition 2.6 above shows that the property of being in the core implies individual rationality and Pareto efficiency. Furthermore, as Example 2.2 above shows, it is stronger in the sense that the converse is not true. Intuitively, the core is an appealing notion when thinking of *decentralized* exchanges. That is, if agents were to exchange houses on their own, it is natural to think that the resulting matching would lie in the core. Otherwise, the agents in a blocking coalition would further exchange their houses to improve their final allocation. This line of reasoning brings up the question of whether the core is nonempty. That is, if left to their own devices to exchange houses, would agents be able to "converge" to an allocation in which no further exchange is possible?

2.2 Top-trading cycle

In this section we introduce the Top-Trading Cycle (TTC) algorithm, first proposed by David Gale. As we shall see, TTC will be the key to showing that the core of a housing market is always nonempty. Furthermore, it will give us a simple way to compute matchings that are in the core.

Algorithm 2.7 (Top-Trading Cycle)**.** *Given a preference profile $(\succcurlyeq_i)_{i\in I} \in \mathcal{P}(H)^n$, proceed in steps as follows. Initially, all agents are unmatched and all houses are available.*

- *Consider a directed graph in which unmatched agents and available houses are vertices (a.k.a. nodes). Each agent points to their favorite available house, and each house points to their owner.*
- *Find all cycles in the graph, i.e., chains of houses and agents of the form* $(h^m, i^m)_{m=1}^M$ *where each* h^m *points to* i^m*, who points to* h^{m+1}*, and* i^M *points to* h^1*. Assign each agent within a cycle to the house they are pointing to and remove them from the market.*
- *Repeat the above procedure with the remaining agents and their houses until there are no more houses in the market.*

Before moving on, we need to verify that the TTC is a well-defined algorithm, i.e., that it always generates a well-defined matching in a finite number of steps. First, note that in every step an agent leaves the market if and only if the house they originally owned also leaves the market. This is because agents are always in the same cycle as the house they own. Hence, the set of available houses is the same as the houses owned by unmatched agents and vice versa. Second, we need to show that the mechanism does not get "stuck." That is, we need to prove that (i) we can always find at least one cycle, and (ii) no two cycles will ever intersect (so that we can readily assign agents to the houses they are pointing to within a cycle).

Proposition 2.8. *Every directed graph formed by agents pointing to a unique house and houses pointing to a unique agent has at least one cycle, and no two cycles intersect.*

Exercise 2.9. Prove Proposition 2.8.

Corollary 2.10. *The Top-Trading Cycle algorithm generates a matching in a finite number of steps.*

Exercise 2.11. Explain why Corollary 2.10 follows from Proposition 2.8.

Now that we know TTC always generates a well-defined matching, we show that it always generates matchings that are Pareto efficient and individually rational.

Proposition 2.12. *The Top-Trading Cycle algorithm generates a matching that is individually rational and Pareto efficient.*

Proof. First, we show Pareto efficiency. The proof is very similar to the one for Serial Dictatorship (cf. Proposition 1.18). By contradiction, assume there exists a matching

$\nu \in \mathcal{M}(I, H)$ that Pareto dominates μ, the outcome of TTC. All agents leaving the market in round 1 of the TTC mechanism obtain their top choice (among all houses); hence, their assignment in ν must be the same as in μ. Since all agents leaving in round 1 get their top choices in both μ and ν, agents leaving in round 2 cannot get in ν, houses that, under μ, were assigned to agents leaving in round 1. Hence, they must also be getting the same houses under ν and μ. The same argument applies inductively for every round, yielding a contradiction.

To show that the resulting matching is individually rational it suffices to note that houses never leave the market prior to their owners. Since agents always leave the market with their most preferred house among the available ones, which includes their own house, the house they are assigned to will be at least as preferred as the one they own. Q.E.D.

Next, we show that the TTC algorithm not only produces Pareto efficient and individually rational matchings, but also that belong in the core. Furthermore, we show that the core contains a single matching, which is precisely the one generated by TTC. Therefore, TTC gives us both a way to prove that the core of a housing market is always nonempty, and an algorithm to find the unique matching in the core.

Theorem 2.13 (Shapley and Scarf 1974; Roth and Postlewaite 1977)**.** *The core of a housing market is nonempty and contains a unique matching, the one generated by the Top-Trading Cycle algorithm.*

Proof. Let μ be the matching generated by TTC. Let I_k denote the set of agents who leave the market in the k-th round of TTC. Note that $I_1, I_2, \ldots, I_K$ form a partition of the set of agents.

First, we show that the matching generated by TTC is in the core. By contradiction, assume there exists a coalition $A \subseteq I$ that blocks μ with another matching $\nu \in \mathcal{M}(I, H)$. Consider the subset of agents in the coalition that are strictly better of under ν than under μ, i.e., let $j \in B$ if and only if $j \in A$ and $\nu(j) \succ_j \mu(j)$. Let k^* be the first round in which an agent in B leaves the market, i.e., $k^* = \min\{k : B \cap I_k \neq \emptyset\}$. Let $j \in B \cap I_{k^*}$. Then, under ν, agent j is getting a house, $\nu(j)$, that left the market before round k^*. Denote this round by k^{**}. Then, there is a member of the coalition, $a_1 \in A$, who initially owned house $\nu(j)$ and left the market in round

k^{**}. Furthermore, a_1 is not in B, i.e., $\nu(a_1) = \mu(a_1)$. In round k^{**}, a_1 belongs to a cycle of the form:

$$a_1 \to h_{a_2} \to a_2 \to \cdots \to a_m \to h_{a_1}.$$

The key of this part of the proof is to show that agents $a_2, \ldots, a_m$ are also part of the coalition A, and are not in B (the same as a_1), which implies that everyone leaving the market in round k^{**} is getting the same house under μ and ν. In particular, a_m is getting h_{a_1}, which is the house originally owned by a_1, i.e., $\nu(j)$, a contradiction. To show this, note that $\nu(a_1) = \mu(a_1)$ implies that $h_{a_2} = \nu(a_1)$, so a_2 is also a member of the coalition (since a_1 is getting their house in ν). Following the same reasoning, $h_{a_3} = \nu(a_2)$, implying that a_3 is also a member of the coalition, and so on and so forth.

To conclude the proof of the Theorem, we need to show that there is no other matching in the core besides μ. By contradiction, assume there exists ν in the core such that $\mu \neq \nu$. Let i be the first agent who leaves the market in TTC with $\mu(i) \neq \nu(i)$. Wlog, assume $i \in I_k$. Hence, every agent in $I_1, \ldots, I_{k-1}$ gets the same house under μ and ν. This implies that, under ν, every agent in I_k is getting a house of an agent who leaves on k or afterwards. Since, under μ, agent i is getting their favorite among all these houses and $\mu(i) \neq \nu(i)$, ν makes agent i worse off. However, this would imply that agents in I_k can form a coalition and block ν with μ, which is a contradiction. Q.E.D.

Next, we show that TTC is also a strategy-proof mechanism. That is, agents have no incentives to misreport their preferences when their house is assigned via TTC. Therefore, mediating exchanges via TTC guarantees that: (i) the resulting allocation will be Pareto efficient, and (ii) agents will have incentives to participate in the mechanism and report their true preferences.

Theorem 2.14 (Roth 1982)**.** *The Top-Trading Cycle mechanism is strategy-proof.*

To formally prove Theorem 2.14, we require the following lemma. Intuitively, the lemma shows that agents cannot affect the cycles leaving the market before them by misreporting their preferences.

Lemma 2.15. *Fix an agent i and a profile of preferences $\succcurlyeq_{-i}$ for the other agents. Consider two preference relations for i, $\succcurlyeq_i$ and $\succcurlyeq_i'$. Let k and k' be the rounds in the TTC at which*

agent i leaves the market when reporting $\succcurlyeq_i$ and $\succcurlyeq_i'$, respectively. At round $\min\{k, k'\}$, the houses and agents remaining in the market are the same under the two preferences.

Proof. The key to show this lemma is to note that whether i reports $\succcurlyeq_i$ or $\succcurlyeq_i'$ does not affect any of the cycles formed before i leaves the market. Indeed, wlog, assume $k' \geq k > 1$, so that there is at least one round before i leaves the market when reporting $\succcurlyeq_i$. The cycle formed in round 1 depends on the preferences reported by other agents, which are fixed in $\succcurlyeq_{-i}$. Even if i points to a house in the cycle, the only way for i to form part of the cycle is for someone in the cycle to point to the house owned by i, which does not occur (otherwise i would be in the cycle and would leave in round 1). Therefore, whatever preferences i reports, $\succcurlyeq_i$ or $\succcurlyeq_i'$, at the start of round k, since the set of cycles leaving the market in prior rounds is the same, the set of remaining houses and agents is also the same. Q.E.D.

Proof of Theorem 2.14. Consider an agent i with true preferences $\succcurlyeq_i$, a fixed profile $\succcurlyeq_{-i}$ for other agents, and alternative preferences $\succcurlyeq_i'$ for i. Let k and k' be the rounds in TTC at which i leaves the market when reporting $\succcurlyeq_i$ and $\succcurlyeq_i'$, respectively. Consider two cases.

First, assume $k \geq k'$, i.e., the case in which i would leave the market at the same time or before by misreporting $\succcurlyeq_i'$. At the beginning of round k', by Lemma 2.15, the set of agents and houses in the market is the same under both $\succcurlyeq_i$ and $\succcurlyeq_i'$. Note that under $\succcurlyeq_i'$, agent i leaves the market with some house h', which is part of a cycle:

$$h' = h_{i_1} \to i_1 \to h_{i_2} \to i_2 \to h_{i_3} \to \cdots \to h_i \to i,$$

in which $i_1, i_2, \ldots$ are all pointing to their favorite houses, under $\succcurlyeq_{-i}$. The key is to note that the chain $(h', i_1, h_{i_2}, i_2, h_{i_3}, \ldots, h_i)$ will remain in the market until i chooses to close off the cycle, either by pointing to h' or by pointing somewhere else that eventually reaches h' (or any other house in the cycle). Hence, by reporting the truth, i will point to their top choice in every subsequent round, and might get something better, or eventually pick h' if it is the best remaining house. In other words, i has no incentives to "close" the cycle before and leave the market with h'.

Second, assume $k' > k$, i.e., the case in which i would leave the market afterwards by misreporting $\succcurlyeq_i'$. Note that by reporting the truth, i leaves the market at time k with the best house among all the remaining ones at the start of round k.

Since the houses remaining at the start of round k' is a subset of the ones at round k, i has no incentive to misreport their preferences to leave afterwards. Q.E.D.

The next Theorem goes further and shows that, actually, TTC is the unique mechanism that satisfies the above properties. That is, there exists no other mechanism that is also strategy-proof, Pareto efficient and individually rational.

Theorem 2.16 (Ma 1994)**.** *A mechanism is strategy-proof, Pareto efficient and individually rational if and only if it is the Top-Trading Cycle mechanism.*

Proof. We will show that TTC is the unique mechanism satisfying the three properties: strategy-proofness, Pareto efficiency, and individual rationality. Let τ denote the TTC mechanism. Let ϕ be another mechanism. By contradiction, assume that ϕ is distinct to τ and also satisfies all three properties. Fix a profile of preferences $(\succcurlyeq_i) \in \mathcal{P}(H)^n$. Let I_1 be the set of agents who leave the market in the first round of TTC. We show that, for all $i \in I_1$, both mechanisms assign the same allocation, i.e.,

$$\phi[(\succcurlyeq_i)](i) = \tau[(\succcurlyeq_i)](i).$$

Towards a contradiction, assume this is not the case. Since every $i \in I_1$ is getting their top choice in TTC, then there must exist some i in I_1 that is strictly better under TTC than under ϕ, i.e., $\tau[(\succcurlyeq_i)](i) \succ_i \phi[(\succcurlyeq_i)](i)$. By individual rationality of ϕ, we have $\tau[(\succcurlyeq_i)](i) \succ_i \phi[(\succcurlyeq_i)](i) \succcurlyeq_i h_i$. Hence, under TTC, agent i is trading with other agents. Consider the following cycle:

$$i = i_1 \to h_2 \to i_2 \to h_3 \to \cdots \to i_m \to h_i \to i.$$

For each agent in the cycle, the house they are pointing to is their top choice under. Since agent i is strictly better off, the agents in this cycle reach an allocation that Pareto dominates the original allocation. However, since $\tau[(\succcurlyeq_i)](i) \succ_i \phi[(\succcurlyeq_i)](i)$, this is not the case under ϕ. That is, the allocation within the agents in the cycle under ϕ admits a reshuffling (or trade) among such agents in which everyone is better off (and i is strictly better off). The rest of the proof lies in noting that this cannot be the case while ϕ satisfies the three properties.

Consider the following alternative preferences for each agent in the cycle, $\succcurlyeq'_{i_k}$. Assume each agent in the cycle reports under $\succcurlyeq'_{i_k}$ the house they are getting un-

der TTC as their top choice, i.e., the top choice of i_k is h_{k+1}, but they (mis)report their own house h_k as their second top choice. Hence, when reporting $\succcurlyeq'_{i_k}$, every agent in the cycle gets the same allocation under TTC as when they report their true preferences.

Under ϕ, when i_k reports $\succcurlyeq'_{i_k}$, they must get either their top choice h_{k+1} or their original house h_k (recall that for every agent in the cycle the house they obtain under ϕ is between their top choice and their own house).

Since ϕ is strategy-proof, agent $i = i_1$ must be getting their own house under ϕ when reporting $\succcurlyeq'_i$; otherwise, they would be getting h_2 (their top choice) and would have incentives to misreport (recall that they get a house strictly worse than their top choice when reporting their true preferences to ϕ). That is, $\phi(\succcurlyeq'_i, \succcurlyeq_{-i}) = h_1$. This, in turn, implies that, at the profile

$$(\succcurlyeq'_i, \succcurlyeq_{i_2}, \ldots, \succcurlyeq_{i_{m-1}}, \succcurlyeq'_{i_m}),$$

agent m must also be assigned to their own house h_m. This is because at $\succcurlyeq'_{i_m}$ agent i_m can only be assigned their own house or their top choice, h_i, which is being assigned to i under $\succcurlyeq'_i$. Consequently, at the profile

$$(\succcurlyeq'_i, \succcurlyeq_{i_2}, \ldots, \succcurlyeq'_{i_{m-1}}, \succcurlyeq'_{i_m}),$$

agent $m-1$ must also be assigned to their own house. By induction, at the profile

$$(\succcurlyeq'_i, \succcurlyeq'_{i_2}, \ldots, \succcurlyeq'_{i_{m-1}}, \succcurlyeq'_{i_m}),$$

everyone in the cycle is assigned to their own house. But then this allocation would not be Pareto efficient, since the agents could trade to the TTC allocation and be better off. Hence, we conclude that ϕ and τ must assign the same allocation to all agents in I_1. The proof follows inductively over the sets of agents who leave the market in TTC in subsequent rounds. Q.E.D.

2.3 House allocation with existing tenants

Consider a house-allocation problem $(I, H, (\succcurlyeq_i)_{i\in I})$ in which a subset of agents I_E are **existing tenants**, they already own a house, while the rest of the agents $I_N =$

$I \setminus I_E$ are **new applicants**, they do not own a house. Let $H_O = \{h_i : i \in I_E\}$ be the set of **occupied houses**, where h_i denotes the house occupied by $i \in I_E$. Similarly, let $H_V = H \setminus H_O$ denote the set of **vacant houses**. As before, assume that $\succcurlyeq_i$ is a linear order for every $i \in I$. A **house-allocation problem with existing tenants** is given by the tuple $(I_E, I_N, H_O, H_V, (\succcurlyeq_i)_{i \in I})$. House-allocation problems with existing tenants are a generalization of house-allocation problems and housing markets. Note that a house-allocation problem with existing tenants boils down to a house-allocation problem if $I_E = H_O = \emptyset$. Similarly, a house-allocation problem with existing tenants is a housing market if $I_N = H_V = \emptyset$.

House-allocation problems with existing tenants are commonly faced by university administrators who need to assign students to dormitory rooms. While there are new students who do not have a room, students from upper-years already have rooms. In what follows, we analyze several mechanisms used in practice to assign students to dormitories at some U.S. universities.

2.4 Assigning students to dormitories

The first algorithm allows agents who own a house to keep it and opt out of the mechanism. Then, it runs Serial Dictatorship with the remaining agents and houses. When the priority order is randomly assigned, this mechanism is commonly known as a "housing lottery." Housing lotteries have been used to assign undergraduate housing at Carneige Mellon, Duke, Michigan, Northwestern, and Penn.

Algorithm 2.17 (Serial Dictatorship with Squatting Rights)**.** *Let π be a priority order over agents (which may be random or favor some students over others, e.g., senior students choose before juniors). Every existing tenant decides whether they want to participate in the mechanism or keep their current house. Those who do not wish to participate are assigned to their current houses and leave the market. All the other houses become available, and serial dictatorship is applied to the remaining houses and agents.*

Exercise 2.18. Evaluate the following statements: (1) Serial Dictatorship with Squatting Rights is Pareto efficient, (2) it is strategy-proof.

The following mechanism runs a Serial Dictatorship for all agents, but gives existing tenants priority over their own house. Namely, the houses of existing tenants

are not available for agents with higher priority, and become available for those with lower priorities only if existing tenants decide to pick another house in their turn.

Algorithm 2.19 (Serial Dictatorship with Waiting List)**.** *Let π be a priority order over agents. Initially, only vacant houses are available. Proceed in steps as follows.*

- *For each agent, consider the set of **currently acceptable houses**. For existing tenants, this is the set of available houses that are at least as preferred as their own house, plus their own house. For new applicants, this is the set of available houses.*
- *Pick the agent with the highest priority among those who have at least one currently acceptable house. Assign this agent to their top choice among their currently acceptable houses. Remove the agent and their newly assigned house from the market.*
- *If the agent was an existing tenant who was assigned to a house different to the one they used to own, the house they used to own becomes available.*

Exercise 2.20. Evaluate the following statements: (1) Serial Dictatorship with Waiting List is Pareto efficient, (2) it is strategy-proof.

The following mechanism aims to give priority to existing tenants over their own house inasmuch as it affects agents with higher priorities. This mechanism is used in one of the residencies at MIT.

Algorithm 2.21 (MIT NH4)**.** *Let π be a priority order over agents. The first agent is **tentatively assigned** to their top choice among all houses, the next agent is tentatively assigned to their top choice among the remaining houses, and so on, until all houses have been assigned or a **squatting conflict** occurs. A squatting conflict occurs if it is the turn of an existing tenant, say $i \in I_E$, and they find all of the available houses worse than h_i, the house they previously owned. This means that another agent, say $i' \in I$, called the **conflicting agent**, was tentatively assigned to h_i previously. At this point, the existing tenant i is assigned to their own house h_i, and they are removed from the market. All the tentative assignments up to the conflicting agent i' are erased. Upon this point, the squatting conflict is resolved, and the algorithm follows in the same fashion, starting with the conflicting agent i' again. Every squatting conflict is resolved in the same way.*

Exercise 2.22. Evaluate the following statements: (1) MIT NH4 is Pareto efficient, (2) it is strategy-proof.

2.5 TTC with existing tenants

Another possibility is to extend the TTC algorithm to a setting with existing tenants. The resulting mechanism is called YRMH-IGYT, which stands for "You request my house—I get your turn." This algorithm is due to Abdulkadiroğlu and Sönmez (1999).

Algorithm 2.23 (YRMH-IGYT)**.** *Let π be a priority order over agents. Initially, all agents are unassigned, and the set of* ***available*** *houses is the set of vacant houses. Proceed in steps as follows.*

- *Consider a directed graph in which every unassigned agent points to their favorite house on the market (regardless of whether it is available or not), every available house on the market points to the unassigned agent with the highest priority, and every occupied house on the market points to its owner.*
- *According to Proposition 2.8, there will be at least one cycle and no two cycles will intersect. Assign every agent in a cycle to the house they are pointing to and remove them from the market.*
- *Whenever an existing tenant is assigned to a house while having the highest priority, the house which they used to occupy becomes available for the next round.*

Exercise 2.24. Assume an existing tenant is part of a cycle at some point in the YRMH-IGYT algorithm. Show that the house owned by the existing tenant will *not* be in any cycle (and thus will become available in the next round) if and only if the existing tenant has the highest priority.

Exercise 2.25. Why does the YRMH-IGYT algorithm is called "You request my house—I get your turn"?

Theorem 2.26 (Abdulkadiroğlu and Sönmez 1999)**.** *The YRMH-IGYT algorithm is Pareto efficient, individually rational, and strategy-proof.*

Exercise 2.27. Prove Theorem 2.26.

Notes

Shapley and Scarf (1974) first proved that the core of a housing market is nonempty, but their proof did not rely on the Top-Trading Cycle algorithm. David Gale suggested a simpler proof using the TTC. Roth and Postlewaite (1977) proved that the

core of a housing market contains a unique matching. Strategy-proofness of the TTC is due to Roth (1982), and its characterization is due to Ma (1994). The discussion of house-allocation problems with existing tenants draws heavily from Abdulkadiroğlu and Sönmez (1999), who proposed and studied the properties of the YRMH-IGYT mechanism.

3 Kidney exchange

Getting a kidney transplant is the preferred treatment for people suffering from acute kidney failure. Transplanted kidneys come from both deceased and living donors. However, there is a worldwide shortage of kidneys. In 2016, over 120,000 people were waiting for a lifesaving organ transplant in the U.S. Of these, more than 100,00 were waiting for kidneys. The median patient waits over 3.5 years to receive a kidney. Of the 17,107 kidney transplants that took place in the U.S. in 2014, 11,570 (67.6%) came from deceased donors and 5,537 (32.4%) came from living donors. In every country in the world, with the exception of Iran, it is illegal to buy and sell human kidneys.

Typically, living donors are relatives or closely-related people who are willing to donate one of their kidneys to a loved one. However, despite the good intentions, wishing to donate a kidney is sometimes not enough. For a kidney donation to be successful, the blood and tissue types of the donor and the recipient need to be compatible. One of the most successful applications of market design to date has been to increase the supply of kidneys from living donors by performing **kidney exchanges**. In a two-way kidney exchange, your donor gives their kidney to patient k and the donor of patient k gives you theirs. In other words, patients exchange kidney donors. In a chain of exchanges, there are K pairs of donors and patients. Donor $k \in \{1, \ldots, K-1\}$ gives their kidney to patient $k+1$, and donor K gives their kidney to patient 1. The first kidney exchange in the world was made in 1991 in South Korea. In Europe, the first kidney exchange was made in Switzerland in 1999. In the U.S., it was in 2000 in Rhode Island.

In this section, we will apply some of the tools from previous sections to the problem of how to design kidney exchanges. We will also learn additional tools to design optimal pairwise exchanges.

3.1 Blood and tissue type compatibility

Humans may have one of four different ABO blood-types: O, A, B, or AB. As far as blood-types are concerned, everyone can donate or receive a kidney from someone with the same blood-type, but not necessarily so across blood-types. Figure 1 illustrates blood-type compatibilities. People with blood-type O may donate a kidney

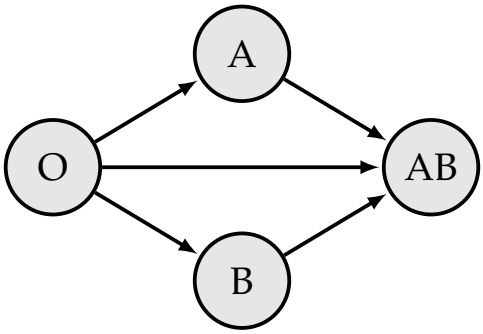

Figure 1: Blood-type compatibility

to anyone, but cannot receive a kidney from someone with a different blood-type. People with blood-type A or B may donate a kidney to AB's, but may receive a kidney only from O-types. And people with blood-type AB cannot donate a kidney to someone with a distinct blood-type, but may receive a kidney from anyone. Around 41.2% of the worldwide population has blood-type O, 29.4% has A, 23.12% has B, and 6.2% has AB. Interestingly, the distribution of ABO blood-types varies across countries and ethnicities.[3]

If one person wishes to donate a kidney to another person, in addition to their blood-types being compatible, there needs to be "tissue type compatibility." The tissue type, formally known as HLA type, is a combination of six proteins. As the mismatch in the HLA types of a donor and patient increases, the likelihood of a successful transplant decreases. Moreover, donors and patients must pass a "cross-match" test, through which it is determined if a patient has antibodies against the HLA in the donor kidney. The presence of antibodies effectively rules out transplantation.

Blood-type and tissue-type compatibilities have been used in the design of mechanisms to allocate donated kidneys. For instance, when a cadaveric kidney (donated by a deceased patient) becomes available for transplantation, the priority of each patient in the waiting list is typically determined by factors including the blood-type, HLA type-compatibility, time spent on the waiting list, etc. Similarly, the effects of distinct design choices may depend on the structure of the ABO blood-type compatibilities. For example, a proposed allocation rule known as an *indirect exchange*

[3]For more details, see: https://en.wikipedia.org/wiki/Blood_type_distribution_by_country. Blood-types are usually specified along their RhD antigen, which can be negative or positive. While patients and donors must be compatible in both their ABO type and RhD antigen for blood transfusions, only the ABO blood-type matters for kidney transplants.

program aims to increase the amount of kidneys by exchanging kidneys from living donors for priorities in the waiting list. That is, a living donor who is incompatible with their intended recipient donates their kidney, and, in exchange, their intended recipient receives a higher priority for the next compatible kidney. However, it has been observed that such indirect exchange programs can harm type O patients who have no living donors and are currently in the waiting list.

3.2 Kidneys as houses, tenants as donors

In its simplest form, kidney exchange may be seen as a direct application of a house allocation problem with existing tenants by viewing kidneys as "houses." Incompatible patient-donor pairs are existing tenants, each of whom "owns a house." Patients who need a kidney and have no donor are "new applicants." They do not "own a house" initially. Finally, cadaveric kidneys and kidneys donated from altruistic donors are "empty houses." Patients have a preference ranking over kidneys which may depend on the blood and tissue-type compatibilities, location of the kidney, and any other factor that may affect the probability of a successful transplant, such as donor age, kidney size, medical history, etc. Importantly, patients may have heterogeneous preferences over otherwise compatible kidneys.

The one difference between kidney exchange and house allocation with existing tenants is that the set of cadaveric kidneys is ex-ante unknown. That is, in an indirect exchange, a patient is given a place in the waiting list, which in essence is a lottery over kidneys. Roth, Sönmez, and Ünver (2004) analyze a modification to the Top-Trading Cycle algorithm in which they allow for patients to have preferences over the set of *currently* available kidneys and a place in the waiting list, denoted by w. The difference with including option w in the model is that several patients can be assigned to a place in the waiting list. Therefore, instead of only having cycles, we may find w-chains, which are chains in which the last patient is assigned to the waiting list. In this scenario, a patient may be in multiple w-chains, whereas in a traditional housing market each agent is in a unique cycle. Therefore, the TTC algorithm must be adjusted appropriately. Roth, Sönmez, and Ünver (2004) study the implications of using the TTC with different "chain selection rules."

In practice, a difficulty of using the TTC mechanism to allocate donated kidneys is that transplants must be done *simultaneously*. The reason is because one cannot

force someone to donate a kidney. To avoid having a donor "backing out" after their intended recipient has received a kidney, transplants are done at the same time. Performing simultaneous organ transplants is not an easy logistical task. It is usually easier and cheaper to carry out exchanges in "small" exchanges or cycles. In the next section, we analyze the problem of kidney exchange when we restrict to pairwise exchanges.

3.3 Pairwise kidney exchange

A **pairwise kidney exchange problem** is a tuple (I, R), where I is a set of n patient-donor pairs, and R is a compatibility matrix. In particular, $R = (r_{ij})_{i \neq j}$ where $r_{ij} = 1$ if the pairs i and j are **compatible**, i.e., the patient in pair i can receive the kidney of the donor in pair j and vice versa. A **matching** is a function $\mu : I \to I$ such that $\mu(i) = j$ if and only if $\mu(j) = i$ and $r_{ij} = 1$. Denote the set of all matchings by $\mathcal{M}(I, R)$. That is, μ specifies a pairwise exchange among compatible patient-donor pairs in I, where pairs i and j exchange kidneys if $\mu(i) = j$, and pair i does not participate in the exchange if $\mu(i) = i$. Therefore, for every matching $\mu \in \mathcal{M}(I, R)$, the set of patients who receive a kidney under μ is given by:

$$M_\mu = \{i \in I : \mu(i) \neq i\}.$$

Definition 3.1. *A matching $\mu \in \mathcal{M}(I, R)$ is **efficient** if there does not exist another matching $\nu \in \mathcal{M}(I, R)$ such that*

$$M_\mu \subseteq M_\nu \quad \textit{and} \quad M_\mu \neq M_\nu.$$

A matching μ is efficient if and only if it is *maximal*, i.e., there is no other matching ν in which all the patients getting a kidney in μ also get a kidney under ν, plus some others.

Exercise 3.2. Note that a pairwise kidney exchange problem does not include preferences, which, in turn, does not allow to have a definition of Pareto efficiency. How would you alter the definition of a pairwise kidney exchange problem, so that the problem admits a notion of Pareto efficiency? What would it be? How would it be related to the notion of an "efficient matching" in Definition 3.1?

We can think of a pairwise kidney exchange problem (I, R) as an undirected graph with n vertices, one for each patient-donor pair, in which two vertices are connected if they are compatible. That is, R is the matrix of edges. In the next section we will introduce a useful tool from graph theory.

3.4 Matroids

A **matroid** is a pair $(X, \mathcal{I})$ where X is a finite set, called the **ground set**, and $\mathcal{I}$ a collection of subsets of X, called the **independent sets**, that satisfy the following properties:

(i) the subsets of independent sets are also independent, i.e., if $J \in \mathcal{I}$ and $J' \subseteq J$ then $J' \in \mathcal{I}$;

(ii) if one independent set J is larger than another one J', i.e., $|J| > |J'|$, then there exists $x \in J \setminus J'$ such that $J' \cup \{x\}$ is an independent set.

Matroids appear in a variety of contexts. If we take X as the set containing the columns of a matrix, and let $\mathcal{I}$ be the collection of all linearly independent columns, then $(X, \mathcal{I})$ forms a matroid. As another example, let X be some finite set and n some integer smaller than $|X|$. If we let $\mathcal{I} = \{S \subseteq X : |S| \leq n\}$, then the pair $(X, \mathcal{I})$ is a matroid.

Proposition 3.3. *A group of patient-donors pairs $J \subseteq I$ is said to be **matchable** if there exists a matching in which every pair in J receives a kidney. Let $\mathcal{I}$ be the collection of all groups of patient-donor pairs that are matchable, i.e.,*

$$\mathcal{I} = \{J \subseteq I : \exists\, \mu \in \mathcal{M}(I, R) \text{ s.t. } J \subseteq M_\mu\}\,.$$

Then, $(I, \mathcal{I})$ is a matroid.

Proof. The first part of the proof is immediate since $J' \subseteq J \subseteq M_\mu$ for some matching μ implies $J' \in \mathcal{I}$. Let $J, J' \in \mathcal{I}$ with $|J| > |J'|$. Let μ and μ' be two matchings such that

$$J \subseteq M_\mu \quad \text{and} \quad J' \subseteq M_{\mu'}.$$

We want to show that there exists some $i \in J \setminus J'$ and some matching ν such that $J' \cup \{i\} \subseteq M_\nu$.

If there exists some $i \in J \setminus J'$ such that $\mu'(i) \neq i$, i.e., such that μ' already matches i, then $J' \cup \{i\} \subseteq M_{\mu'}$, and we can simply take $\nu = \mu'$. Hence, assume every $i \in J \setminus J'$ is not matched under μ'. Fix $i^1 \in J \setminus J'$. Pair i^1 is matched by μ to some pair i^2. Then,

(a) if $i^2 \notin M_{\mu'}$, define ν by adding a match between i^1 and i^2 to μ'. Then, we have that $J' \cup \{i^1\} \subseteq M_\nu$.

(b) If $i^2 \in M_{\mu'}$, then i^2 is matched to some $i^3 \neq i^1$ under μ'. If $i^3 \notin M_\mu$, then we have formed a finite sequence, also known as a *path*, of agents i^1, i^2, i^3 such that

$$i^1 \in J \setminus J', \mu(i^1) = i^2, \mu'(i^2) = i^3, i^3 \notin M_\mu.$$

If $i^3 \in M_\mu$, then the path would have one more pair; namely, i^4, the match of i^3 under μ, which is different to i^1 and i^2. Subsequently, if i^4 is matched under μ', the path would go on. This means that, in general, we can define all such *alternating paths*, which are sequences of pairs with the first pair in $J \setminus J'$, and each subsequent pair being matched under both μ and μ', except the last one, which is matched only under one of the two matchings. That is, an alternating path is a sequence of pairs $i^1, i^2, \ldots, i^k$ such that

$$i^1 \in J \setminus J', \mu(i^1) = i^2, \mu'(i^2) = i^3, \mu(i^3) = i^4, \ldots, i^k \notin M_\mu \cap M_{\mu'},$$

where last pair might belong to either J or J', but not both. We argue that there must exist at least one path P such that the last pair i^k is matched under μ. Notice that if this were not the case, this would imply that there are more elements in $J' \setminus J$ than in $J \setminus J'$ (since all the pairs within a path are unique and each pair appears in exactly one path), which is a contradiction since $|J| > |J'|$. Take any path in which the last agent is matched under μ, and define matching ν as follows. For every i along the path, let $\nu(i) = \mu(i)$; otherwise, let $\nu(i) = \mu'(i)$. Hence, note that matching ν is well-defined and $J' \cup \{i^k\} \in M_\nu$.

Q.E.D.

The next result shows why it is useful to think of the matroid underlying a pairwise kidney exchange problem.

Proposition 3.4. *If $\mu, \nu \in \mathcal{M}(I, R)$ are efficient, then $|M_\mu| = |M_\nu|$.*

Proof. Suppose μ and ν are efficient, but $M_\mu > M_\nu$. By condition (ii) of a matroid, there exists matching ν' and some $i \in M_\mu \setminus M_\nu$ such that $M_\nu \cup \{i\} \subseteq M_{\nu'}$, which contradicts ν being efficient. Q.E.D.

The above result shows that looking for efficient matchings is equivalent to maximizing the number of matches. However, note that this result is no longer true if we allow exchanges among more than two patient-donor pairs.

Exercise 3.5. Provide an example of a multi-way kidney exchange in which there are two efficient matchings with distinct number of matches.

3.5 Priority Mechanisms

In this section, we study priority mechanisms in the context of pairwise kidney exchange. Fix an ordering $\pi : I \to I$ of patient-donor pairs. To simplify the notation, assume that $I = \{1, \ldots, n\}$ are already ordered according to π.

Definition 3.6. *Given a pairwise kidney exchange problem (I, R), define the set of priority mechanisms as follows. Let $\mathcal{E}^0 = \mathcal{M}(I, R)$, and define for every $k \leq n$,*

$$\mathcal{E}^k = \begin{cases} \{\mu \in \mathcal{E}^{k-1} : \mu(k) \neq k\} & \text{if } \exists\, \mu \in \mathcal{E}^{k-1} \text{ s.th. } \mu(k) \neq k \\ \mathcal{E}^{k-1} & \text{otherwise} \end{cases}$$

*The set of **priority matchings** is given by $\mathcal{E}^n$.*

In step 1, we consider $\mathcal{E}^1$, the set of all matchings μ under which patient 1 receives a kidney. In step 2, we consider all the matchings $\mu \in \mathcal{E}^1$ in which patient 2 also receives a kidney, and so on. A priority matching matches as many patients as possible starting with the patient with the highest priority and following the priority ordering, never "skipping" or "sacrificing" a higher priority patient because of a lower priority patient.

Proposition 3.7. *Every priority matching is efficient.*

Exercise 3.8. Prove Proposition 3.7.

The result is not quite obvious. Intuitively, one might expect that by restricting attention in step 1 to matchings where patient 1 receives a kidney, we might end up with an inefficient allocation. However, this is not the case. Even though two different priority matchings might differ in the set of patients that are matched, they will always match the same *number* of patients. As Roth, Sönmez, and Ünver (2005) note in their original paper, *"there is no trade-off between priority allocation and the number of transplants that can be arranged."*

In practice, the basic data for the problem (I, R) is determined by the laboratory results of tissue type compatibility tests. However, in a kidney exchange it is impossible to prevent a patient from declining a medically compatible kidney. Given a priority mechanism, a patient might report a subset $A_i \subseteq K_i$ of kidneys as acceptable to them, where $K_i = \{j \in I : r_{ij} = 1\}$.

Proposition 3.9. *In a priority matching mechanism, it is strategy-proof for the patients to report $A_i = K_i$.*

Proof. Let ϕ be a priority mechanism. It is clear that a patient who is matched cannot gain by lying. Consider a patient i who is unmatched when they report the truth. Let $\mathcal{E}^k$ be the k-th step set obtained when agent i reports the truth, and consider a deviation to A_i and the resulting set $\mathcal{E}_d^k$. Then, $\mathcal{E}_d^k \subseteq \mathcal{E}^k$ for every $k < i$ since, at every step, by excluding some compatible kidneys, the agent can only shrink the chances of a match. Let μ be the matching selected by ϕ under truth-telling. By the definition of priority matching, if $\mu(i) = i$ then $\mu'(i) = i$ for all $\mu' \in \mathcal{E}^{i-1}$. Since $\mathcal{E}_d^{i-1} \subseteq \mathcal{E}^{i-1}$, this implies that i must also be unmatched when lying.

Q.E.D.

Notes

Roth, Sönmez, and Ünver (2004) first applied and extended the results from house allocation with existing tenants to study kidney exchange problems. The discussion on pairwise kidney exchange and priority mechanisms is based on their follow-up paper Roth, Sönmez, and Ünver (2005).

4 Random allocations

Deterministic mechanisms, such as the Serial Dictatorship, often depend on exogenous priority orders. Depending on our design goals, this can be a design *feature* or *liability*. Priority orders make sense when there are external reasons for which we might wish to prioritize certain market participants over others. However, in the absence of a valid justification, priority orders may be a liability since they are asymmetric, and thus unfair, by nature. A natural way to overcome this is to randomise allocations. Indeed, randomisation is common in multiple real-life allocation mechanisms. It is used to assign students to public schools when the number of seats is scarce, to ration offices and parking spaces, and to select citizens to serve as jury members. In this section, we study random allocations.

Formally, a **random allocation problem** has the same ingredients as a house allocation problem: a set of n agents I, a set of n objects X, and a preference profile $(\succcurlyeq_i)_{i\in I} \in \mathcal{P}(X)^n$. Instead of focusing on deterministic matchings as we have done in previous sections, we focus on random allocations.

Definition 4.1. *A **random allocation** is an n-by-n bistochastic matrix $P = (P_{ix})_{i,x}$, where P_{ix} is the probability that object x is assigned to agent i. P **bistochastic matrix** means that the entries of every row and column of P add up to 1, that is, for every $i \in I$ and $x \in X$,*

$$\sum_{x'\in X} P_{ix'} = 1 \quad \textit{and} \quad \sum_{i'\in I} P_{i'x} = 1.$$

Denote the set of all random allocations by $\mathcal{A}(I, X)$.

4.1 Birkhoff-von Neumann theorem

A random allocation can be thought of as specifying a probability distribution over the set of objects for each agent. In principle, if we take these distributions to be independent, nothing in Definition 4.1 seems to preclude an agent to be allocated more than a single object. However, note that the bistochastic requirement limits the degrees of freedom we have over these distributions. A natural way of guaranteeing that each agent is assigned exactly one object would be to define a random allocation as a probability distribution over the set of matchings $\mathcal{M}(I, X)$. As it turns out,

the Birkhoff-von Neumann Theorem establishes some sort of equivalence between random allocations and lotteries over deterministic matchings.

As an example, consider the following random allocation with two agents and two objects:

$$P = \begin{bmatrix} P_{1x} & P_{1y} \\ P_{2x} & P_{2y} \end{bmatrix} = \begin{bmatrix} 1/3 & 2/3 \\ 2/3 & 1/3 \end{bmatrix}.$$

It is tempting to implement P by assigning the first object x among agents with probabilities $(P_{1x}, P_{2x}) = (1/3, 2/3)$, and then the second object y independently with probabilities $(P_{1y}, P_{2y}) = (2/3, 1/3)$. However, this would be incorrect since there would be a positive probability of assigning both objects to the same agent. Namely, note that after assigning the first object, the second object is automatically assigned. Indeed, this idea is captured by the fact that the second row (or column) in P is actually redundant given the bistochastic requirement.

Another possibility is to consider the two deterministic allocations, i.e., matchings, μ_1 and μ_2 given by:

$$\mu_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \text{and} \quad \mu_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

In this case, we could think of a random allocation as a lottery over the set of matchings, i.e., $\theta \in \Delta\mathcal{M}(I, X)$ with $\theta(\mu_1) = 1/3$ and $\theta(\mu_2) = 2/3$.[4] Clearly, in this example both P and θ result in the same final outcome. The Birkhoff-von Neumann Theorem establishes that this decomposition of a random allocation into deterministic matchings is always possible.

Theorem 4.2 (Birkhoff-von Neumann)**.** *Every random allocation P can be decomposed into a lottery θ over the set of deterministic matchings, such that, for every $i \in I$ and $x \in X$,*

$$P_{ix} = \sum_{\mu \in \mathcal{M}(I,X)} \theta(\mu) \cdot 1\{\mu(i) = x\}.$$

Notably, the Birkhoff-von Neumann Theorem does not establish uniqueness. That is, there might be multiple lotteries over the set of matchings that result in the same random allocation.

[4] Here, we denote the set of all probability measures over a set X by ΔX.

4.2 Ordinal preferences and stochastic dominance

So far, we have assumed that each agent is endowed with a preference relation. As we discussed in Section 1, preference relations are *ordinal* by nature. The advantage of this assumption is that the mechanisms we have considered so far have not relied on agents' preference *intensities*.[5] However, when it comes to ranking random allocations, preference relations are not enough. The reason is because, as the following example shows, they do not include information about preferences for *risk*.

Example 4.3. *Suppose the set of objects consists of monetary sums, $X = \{\$0, \$10, \$18\}$. Even if we take for granted that every agents prefers more money to less, this does not allow us to rank random allocations. For instance, consider agent i and the random allocations P and Q such that*

$$P_{i,\$10} = 1 \quad \textit{and} \quad Q_{i,\$0} = Q_{i,\$18} = 1/2.$$

Under P, agent i gets \$10 for sure, while under Q they get nothing with probability $1/2$ and \$18 otherwise. Does agent i prefer P or Q? Given the information on $\succcurlyeq_i$, we cannot tell. An agent may find Q too risky and prefer P, or may not mind the risk and prefer Q.

Now, consider random allocation R such that $R_{i,\$18} = R_{i,\$10} = 1/2$. In this case, R assigns both \$18 and \$10 with equal probability, while P assigns all probability to \$10. Whatever the outcome in R, agent i can only be better off than the corresponding outcome in P. Therefore, it seems reasonable to assume that i prefers R to P. Similarly, note that it is reasonable to assume that R is better than Q.

Example 4.3 above shows that, only in certain cases, preference relations are enough to obtain a ranking over random allocations. The next definition formalizes this notion.

Definition 4.4. *Given a preference profile $(\succcurlyeq_i)_{i\in I}$, and two random allocations P and Q, we say that P (**first-order**) **stochastically dominates** Q if for every $i \in I$ and every*

[5] Measuring preference intensity is not straightforward since it is not clear what is the right "scale." In real-life, agents usually express preference intensity via willingness to pay. However, this notion heavily relies on agents' initial wealth. In the absence of a standard scale or utility measure, it is not obvious how to elucidate individual preferences. Furthermore, it is not clear that agents will agree on the right scale to express preference intensity. For example, if we use willingness to pay, agents with relatively low wealth may not find it acceptable, and may misreport their utility if the right incentives are not in place.

$x \in X$,

$$\sum_{y \in X : y \succcurlyeq_i x} P_{iy} \geq \sum_{y \in X : y \succcurlyeq_i x} Q_{iy}.$$

Intuitively, random allocation P stochastically dominates Q if, for every agent i and every object x, the probability of obtaining an object better than x is higher under P than under Q. Importantly, note that stochastic dominance is an incomplete order, not all random allocations P and Q are comparable. Building on the definition of stochastic dominance, we define our efficiency notion for random allocations.

Definition 4.5. *A random allocation is* ***ordinally efficient*** *if there is no other random allocation that stochastically dominates it.*

Exercise 4.6. What is the intuition behind the definition of ordinal efficiency? Namely, how does it relate to the notion of Pareto efficiency in deterministic allocations? Can you think of a different way of defining "efficiency" in the Pareto sense for random allocations? Would agents participate in mechanisms that do not produce ordinally efficient allocations? Why?

4.3 Cardinal preferences

Another approach to the one described above is to take a stance on agents' preferences for risk. That is, why not assume that agents preferences are represented by a utility function and take expectations? As we shall see, taking this route without making further assumptions is equivalent to working with the notion of stochastic dominance.

Suppose that each agent comes endowed with a utility function $u_i : X \to \mathbb{R}$ that is consistent with their preference relation $\succcurlyeq_i$. Say that agent i prefers random allocation P to Q if and only if its expected utility is higher under P than under Q, i.e.,

$$\sum_{x \in X} u_i(x) P_{ix} \geq \sum_{x \in X} u_i(x) Q_{ix}.$$

Note that this expected-utility ranking is complete over the space of random allocations. Denote by $\mathcal{U}(\succcurlyeq)$ the set of all utility functions that represent the preference

relation $\succcurlyeq$. That is,

$$\mathcal{U}(\succcurlyeq) = \{u : X \to \mathbb{R} : \forall\, x, y \in X, x \succcurlyeq y \Leftrightarrow u(x) \geq u(y)\}\,.$$

The next result shows that, if random allocation P stochastically dominates Q, then P provides higher utility than Q for every agent under *every* utility function that is compatible with their ordinal preferences. Therefore, ordering random allocations according to stochastic dominance amounts to being agnostic about the cardinal aspect of preferences. In this sense, stochastic dominance may be seen as an "assumption-free" ordinal criterion.

Proposition 4.7. *Let $(\succcurlyeq_i)_{i\in I}$ be a profile of agents' preferences. The random allocation P first-order stochastically dominates Q if and only if for every $i \in I$ and every utility function $u_i \in \mathcal{U}(\succcurlyeq_i)$,*

$$\sum_{x\in X} u_i(x)P_{ix} \geq \sum_{x\in X} u_i(x)Q_{ix}.$$

Exercise 4.8. Prove Proposition 4.7.

4.4 Random serial dictatorship

The random serial dictatorship is perhaps the simplest way of allocating objects. Its definition is familiar. First, each agent submits a linear order $\succcurlyeq_i$ over the set of objects. Then, we randomly choose an order over the agents. There are as many as $n!$ orders, and each one is chosen with the same probability. We then assign the first agent to their top choice, the second agent to their top choice among the remaining ones, and so on.

Given the preference profile reported by the agents $(\succcurlyeq_i)$, the random serial dictatorship produces a random allocation $P[(\succcurlyeq_i)]$, where

$$P[(\succcurlyeq_i)]_{xi} = \mathbb{P}\left\{\text{object } x \text{ is assigned to agent } i\right\}.$$

This probability is not easy to compute, as it depends in a complicated way on the profile of preferences $(\succcurlyeq_i)$. For example, fix an agent i and suppose that under their reported preference $\succcurlyeq_i$ object x^* is ranked as their top choice. What is the probability

that i gets their top choice, $P[(\succcurlyeq_i)]_{ix^*}$? We know that

$$P[(\succcurlyeq_i)]_{ix^*} \geq \frac{1}{n}$$

since agent i will have the top priority with probability $1/n$, and hence will be assigned x^*. However, this probability could be higher depending on the preferences of the other agents.

Exercise 4.9. Say that a random mechanism $\phi : \mathcal{P}(X)^n \to \mathcal{A}(X)$ is **ex-post Pareto efficient** if, for all $(\succcurlyeq_i) \in \mathcal{P}(X)^n$, every realisation of the random allocation $\phi[(\succcurlyeq_i)]$ is a Pareto efficient matching. Evaluate: the Random Serial Dictatorship mechanism is ex-post Pareto efficient.

Exercise 4.10. Evaluate: the Random Serial Dictatorship mechanism is strategy-proof.

Random serial dictatorship is a mechanism that is well known, and has many desirable properties, among which are its simplicity and fairness. Perhaps surprisingly, the mechanism is *not* guaranteed to produce a random allocation that is ordinally efficient. Consider the following example.

Example 4.11. *Let $I = \{1, 2, 3, 4\}$ and $X = \{x, y\}$. Assume $x \succ_i y$ for $i = 1, 2$, and $y \succ_i x$ for $i = 3, 4$.*[6] *Let P be the random allocation corresponding to the random serial dictatorship. The probabilities that agent 1 obtains objects x and y are, respectively,*

$$P_{1x} = \frac{5}{12} \quad \textit{and} \quad P_{1y} = \frac{1}{12}.$$

To obtain the above expressions, proceed as follows. There are $4! = 24$ possible orders. Under half of them, agent 1 is in the third or fourth priority and gets no object. They obtain their top-choice x if they get the first priority (6 cases) or if they are in the second priority (6 cases) while agent 2 is not in the top priority (2 cases). Hence, they obtain their top choice in $6 + 6 - 2 = 10$ cases, and the probability of obtaining it is thus $10/24 = 5/12$. Note that agent 1 obtains their second choice y if and only if agent 2 gets the top priority and they get the second one (2 cases), which yields a probability of $2/24 = 1/12$. Finally, note that the probabilities for agent 2 are the same as for agent 1, and those for agents 2 and 3 are symmetrical.

[6] Having two objects instead of four simplifies the calculations, but it is not essential. The same phenomenon can happen with n agents and n objects.

The resulting random allocation P is not ordinally efficient. Consider the following mechanism. We flip two fair coins. The firs coin determines whether agent 1 or 2 gets object x, and the second coin whether agent 3 or 4 gets y. Denote the resulting random allocation by Q. Therefore,

$$Q_{1x} = Q_{2x} = Q_{3y} = Q_{4y} = 1/2, \quad \textit{and} \quad Q_{1y} = Q_{2y} = Q_{3x} = Q_{4x} = 0.$$

Note that Q first-order stochastically dominates P. The probability of getting an object is one half for each agent under both mechanisms. However, all the probability mass is on each agent's top-choice under Q, while it is split across both alternatives under P. Intuitively, this means that every agent prefers allocation Q over P.

Exercise 4.12. Provide an example of a house allocation problem in which Random Serial Dictatorship always generates an ordinally efficient random allocation.

4.5 Top-trading cycle with random endowments

Now, consider extending another of the deterministic mechanisms we have studied in previous sections, the Top-Trading Cycle. The intuition is very simple. The TTC is formally defined for housing markets, in which agents initially own a house. In this setting, we first assign the objects to agents uniformly at random and then run TTC with their initial endowments. The mechanism is attractive since the TTC has several desirable properties in the deterministic case. However, it suffers from the same shortcomings as the Serial Random Dictatorship. Indeed, they are equivalent.

Theorem 4.13 (Abdulkadiroğlu and Sönmez 1998)**.** *Let $\phi^{RSD} : \mathcal{P}(X)^n \to \mathcal{A}(X)$ and $\phi^{TTC} : \mathcal{P}(X)^n \to \mathcal{A}(X)$ be the Random Serial Dictatorship and the Top-Trading Cycle with Random Endowments mechanisms, respectively. As long as $|I| = |X| = n$, for every $(\succcurlyeq_i) \in \mathcal{P}(X)^n$, $i \in I$ and $x \in X$,*

$$\phi^{RSD}[(\succcurlyeq_i)]_{ix} = \phi^{TTC}[(\succcurlyeq_i)]_{ix}.$$

Exercise 4.14. Prove Theorem 4.13.

4.6 Probabilistic serial mechanism

A natural question is whether there exists a mechanism that is guaranteed to produce a random allocation that is ordinally efficient. This question was solved by Bogomolnaia and Moulin (2001). Their mechanism, called Probabilistic Serial, proceeds as follows.

Algorithm 4.15 (Probabilistic Serial)**.** *Given a preference profile $(\succcurlyeq_i)_{i\in I} \in \mathcal{P}(X)^n$, think of each good $x \in X$ as a cake of size 1. Let time run continuously in the interval $[0,1]$, and proceed as follows.*

- *At every instant $t \in [0,1]$, each agent eats from the cake corresponding to their favorite good among the cakes that are not finished. Eating occurs at speed 1: if an agent eats a cake between t_0 and t_1, they eat a fraction $t_1 - t_0$ of the cake. Each agent can only eat from one cake at the time.*
- *The mechanism stops at $t = 1$. The amount of cake eaten by an agent equals the probability with which they receive the good. That is, the mechanism outputs a random allocation P where P_{ix} is equal to the total share of cake x eaten by agent i.*

Example 4.16. *Let $I = \{1,2,3,4\}$ and $X = \{w,x,y,z\}$. Assume preference are given by:*

$$\succ_1, \succ_2 \colon x, y, z, w, \quad \text{and} \quad \succ_3, \succ_4 \colon y, x, w, z.$$

At time $t = 0$, agents 1 and 2 start "eating" from object x, while agents 3 and 4 start eating object y. At time $t = 1/2$, objects x and y have been completely eaten. Agents 1 and 2 then start eating shares of object z, while 3 and 4 shares of w. At time $t = 1$ all goods have been completely eaten. The resulting random allocation is given by:

$$P = \begin{bmatrix} P_{1x} & P_{1y} & P_{1z} & P_{1w} \\ P_{2x} & P_{2y} & P_{2z} & P_{2w} \\ P_{3x} & P_{3y} & P_{3z} & P_{3w} \\ P_{4x} & P_{4y} & P_{4z} & P_{4w} \end{bmatrix} = \begin{bmatrix} 1/2 & 0 & 1/2 & 0 \\ 1/2 & 0 & 1/2 & 0 \\ 0 & 1/2 & 0 & 1/2 \\ 0 & 1/2 & 0 & 1/2 \end{bmatrix}.$$

Notably, this random allocation P stochastically dominates the random allocation generated by the Random Serial Dictatorship. Indeed, as the theorem below shows, this is a general property of the Probabilistic Serial mechanism.

Theorem 4.17 (Bogomolnaia and Moulin 2001)**.** *For every profile of preferences, the Probabilistic Serial mechanism produces an ordinally efficient random allocation.*

Before proving Theorem 4.17, we prove a useful Lemma. Given a preference profile $(\succcurlyeq_i)$ and a random allocation P, define a directed graph as follows. Each object corresponds to a node. Connect nodes $y \to x$ if there is an agent i such that $x \succ_i y$ and $P_{iy} > 0$. Denote the resulting graph by $G(\succcurlyeq, P)$.

Lemma 4.18. *If random allocation P is stochastically dominated by another random allocation, then $G(\succcurlyeq, P)$ has a cycle.*

Proof. Let $P, Q \in \mathcal{A}(I, X)$ and assume Q stochastically dominates P. Hence, there exist an agent i_1 for which Q gives higher probability to more preferred alternatives than P; that is to say, there exist $x, x_1 \in X$ s.th.

$$x_1 \succ_{i_1} x, \quad Q(i_1, x_1) > P(i_1, x) \quad \text{and} \quad Q(i_1, x) < P(i_1, x).$$

This implies $x \to x_1$. Since

$$\sum_{i \in I} Q(i, x_1) = \sum_{i \in I} P(i, x_1) = 1,$$

there exists another agent i_2 for which Q assigns object x_1 with lower probability than under P, i.e.,

$$Q(i_2, x_1) < P(i_2, x_1).$$

Since Q dominates P, there must be another object x_2 such that $x_2 \succ_{i_2} x_1$ and Q assigns x_2 to i_2 with higher probability than P, i.e.,

$$Q(i_2, x_2) > P(i_2, x_2).$$

Hence, $x_1 \to x_2$. Following the same logic, since $Q(i_2, x_2) > P(i_2, x_2)$, we can find another agent i_3 for which Q assigns x_2 with lower probability than P, i.e., $Q(i_3, x_2) < P(i_3, x_2)$. And, similarly, since Q dominates P, there must exist an object x_3 such that $x_3 \succ_{i_3} x_2$ and $Q(i_3, x_3) > P(i_3, x_3)$. Which implies $x_2 \to x_3$. Proceeding inductively, we obtain a sequence of objects where each one is connected to its

predecessor through an arrow. Since the set of objects is finite, we must eventually find a cycle. Q.E.D.

Sketch of Proof of Theorem 4.17. The intuition behind the Lemma is that agents in a cycle would like to exchange probabilities. To see this clearly, consider a cycle of two objects $x \to y \to x$. This implies that there exists two agents, i_1 and i_2, such that one prefers x to y and is getting y with positive probability, while the other one prefers y to x and is getting x with positive probability. Concretely, assume $x \succ_{i_1} y$, $P(i_1, y) > 0$, $y \succ_{i_2} x$, and $P(i_2, x) > 0$. Intuitively, these two agents would like to exchange the probabilities with which they get their least preferred object. The key to the proof is to realize that these sort of situations are not possible under the Probabilistic Serial mechanism.

We show this using this simple example of two agents. Let $t_1 \in [0, 1]$ be the first time agent i_1 eats a share of object y. Since they prefer x to y, it must be that x was not available at time t_1. Similarly, let $t_2 \in [0, 1]$ be the first time agent i_2 eats a share of object x. Then, object y must not have been available at time t_2. Since y was available at time t_1, then $t_1 < t_2$. However, since x was available at time t_2, we also have $t_2 < t_1$, which is clearly a contradiction. Q.E.D.

Besides being ordinally efficient, another key property of the Probabilistic Serial mechanism is that it is envy-free in the sense that no agent would rather have the assignment of another agent.

Definition 4.19. *Given a preference profile $(\succcurlyeq_i)_{i \in I}$ and a random allocation P, we say that agent i **envies** agent j if agent i prefers the random assignment of j under P in the first-order stochastic dominance sense. That is, i envies j under P if, for every $x \in X$,*

$$\sum_{y \in X : y \succcurlyeq_i x} P_{iy} \geq \sum_{y \in X : y \succcurlyeq_i x} P_{jy}.$$

*A random allocation is envy-free if no agent envies another one. A random mechanism $\phi : \mathcal{P}(X)^n \to \mathcal{A}(X)$ is **envy-free** if $\phi[(\succcurlyeq_i)]$ is an envy-free random allocation for every $(\succcurlyeq_i) \in \mathcal{P}(X)^n$.*

Theorem 4.20. *The Probabilistic Serial mechanism is envy-free.*

Exercise 4.21. Prove Theorem 4.20.

Notes

The analysis of TTC with Random Endowments is from Abdulkadiroğlu and Sönmez (1998). The Probabilistic Serial mechanism was developed and analyzed by Bogomolnaia and Moulin (2001). They offer further characterizations for the Random Serial Dictatorship and the Probabilistic Serial mechanisms, and also provide an impossibility result. For a simple exposition of the the Birkhoff-von Neumann Theorem along concrete examples, see Chapter 12 of Haeringer (2017). The discussion over first-order stochastic dominance and utility representation is standard in the literature, e.g., Mas-Colell, Whinston, and Green (1995).

5 Marriage market

In this section, we turn to the analysis of two-sided matching markets. In a two-sided matching market, agents are divided into two sides. Each agent is assumed to have preferences over the agents on the other side of the market. Following the seminal contribution of Gale and Shapley (1962), we frame the model in terms of *marriages*. There are two finite and disjoint sets of agents, one of men and one of women. The men are assumed to have preferences over women, and the women over the men. The main question is, how do we match men and women in a way in which no one wishes to "divorce" and marry someone who would also like to marry them.[7]

Definition 5.1. *A **marriage market** is a tuple $(M, W, (\succcurlyeq_m)_{m\in M}, (\succcurlyeq_w)_{w\in W})$, where M amd W are finite, nonempty, and disjoint sets of **men** and **women**, and $(\succcurlyeq_m)_{m\in M}$ and $(\succcurlyeq_w)_{w\in W}$ are preference profiles with $\succcurlyeq_m \in \mathcal{P}(W \cup \{m\})$ for every $m \in M$, and $\succcurlyeq_w \in \mathcal{P}(M \cup \{w\})$ for every $w \in W$. Often, we denote marriage markets simply by $(M, W, \succcurlyeq)$, where $(\succcurlyeq)$ is shorthand for $((\succcurlyeq_m)_{m\in M}, (\succcurlyeq_w)_{w\in W})$.*

Note that we allow for agents to have a preference for themselves, which amounts to remaining single. If, say, man m has preferences such that $w_1 \succ_m m \succ_m w_2$, this means that m prefers w_1 over being single, but would rather remain single than marrying w_2. Accordingly, we allow for agents to remain single in a matching by matching them with themselves.

Definition 5.2. *A **matching** is a function $\mu : M \cup W \to M \cup W$ such that, for all $m \in M$ and $w \in W$, (i) $\mu(m) \in W \cup \{m\}$, (ii) $\mu(w) \in M \cup \{w\}$, and (iii) $\mu(m) = w$ if and only if $\mu(w) = m$. Denote the set of all matchings by $\mathcal{M}(M, W)$.*

Exercise 5.3. Give 10 real-life examples (different to the ones mentioned in footnote 7) of two-sided matching markets that have the same (or similar) structure as a marriage market.

[7] While we stick with the original formulation in terms of a marriage market, we recognize that it is notably gendered and not inclusive. It restricts attention to monogamous and heterosexual marriages between binary agents. Indeed, in their original formulation, David Gale and Lloyd Shapley (1962) admitted to having "abandoned reality altogether and entered the world of mathematical make-believe." Aside from the labels "men," "women," and "marriage," the key content of the model is that agents are split into two sides, and each agent has preferences over agents on the other side. Examples of marriage markets include: workers and firms, schools or colleges and students, doctors and hospitals, advisors and students, adoptees and adoptive parents, patients and organ donors, foster children and foster homes, etc.

5.1 Stability and efficiency

Firstly, we consider the notions of Pareto efficiency and stability. The former is the same one we have used in previous sections: a matching is not efficient if we can improve someone without harming anyone else. The notion of stability is intimately related with the core. In a stable matching, no one wishes to be matched with someone whom they are not matched with *and* who would also prefer to be matched with them.

Definition 5.4. *Given a preference profile $(\succcurlyeq_i)_{i\in M\cup W}$, a matching is **Pareto efficient** if there is no matching μ' such that $\mu'(i) \succcurlyeq_i \mu(i)$ for all $i \in M \cup W$ and $\mu'(i) \succ_i \mu(i)$ for some $i \in M \cup W$.*

Definition 5.5. *Let $(\succcurlyeq_i)_{i\in M\cup W}$ be a preference profile. Woman w is **acceptable** to man m if $w \succcurlyeq_m m$, and man m is acceptable to woman w if $m \succcurlyeq_w w$. A matching is **individually rational** if everyone is matched with an acceptable partner. A pair $(m, w) \in M \times W$ **blocks** a matching μ if $w \succ_m \mu(m)$ and $m \succ_w \mu(w)$. If (m, w) block μ, they are called a **blocking pair**. A matching is **stable** if it is individually rational and admits no blocking pair. Denote by $\mathcal{S}(M, W, \succcurlyeq)$ the set of all stable matchings in $(M, W, \succcurlyeq)$.*

Exercise 5.6. Evaluate: the set of stable matchings coincides with the core.

Our first result shows that every stable matching is Pareto efficient: requiring stability is at least as restrictive as requiring Pareto optimality.

Proposition 5.7. *Every stable matching is Pareto efficient.*

Proof. By contradiction, suppose μ' Pareto dominates $\mu \in \mathcal{S}(M, W, \succcurlyeq)$. Wlog, suppose $\mu'(m) \succ_m \mu(m)$. Let $w' = \mu'(m)$. Then, $\mu'(w') \succcurlyeq_{w'} \mu(w')$ since μ' Pareto dominates μ. Note that we actually have $\mu'(w') \succ_{w'} \mu(w')$ since $\mu'(w') \neq \mu(w')$. Then (m, w') block μ, which is a contradiction. Q.E.D.

Exercise 5.8. Evaluate: every Pareto efficient matching is stable.

Exercise 5.9. Evaluate: the First and Second Welfare Theorems are satisfied in a marriage market.

Stability is a key criterion both in *centralized* and *decentralized* settings. In a centralized setting, agents do not have incentives to participate in mechanisms that assign non-stable matchings. If the matching prescribed by a mechanism is not stable,

agents will have incentives to deviate from the matching prescribed by the mechanism. In this sense, stability is a requirement for a mechanism to be used and respected by agents. In a decentralized setting, in which agents are left to match with one another through their own devices, stability is important in the same sense as the core. It is reasonably to assume that if agents match and unmatch freely with one another, they will continue to do so until they reach a stable matching. The first question we address is whether stable matchings always exist, and how to find them. The answer was provided by David Gale and Lloyd Shapley in 1962.

Algorithm 5.10 (Gale and Shapley 1962)**.** *Consider the men-proposing version. Initially, all men are active and no agent is provisionally matched. Proceed in steps as follows.*

- *All active men propose to their most preferred woman among the ones they have not proposed to previously.*
- *Each woman considers the set of men who have just proposed to her, and their provisional partner if they have one. Women become provisionally matched to their favorite man among this set. All men who are not provisionally matched become active.*
- *Stop if there are no active men, or if all active men have proposed to all acceptable women.*

Note: the woman-proposing version of the algorithm is analogous.

The Gale-Shapley Algorithm is also known as the Deferred Acceptance (DA) algorithm. As we show next, it is the key to showing that the set of stable matchings $\mathcal{S}(M, W, \succcurlyeq)$ is nonempty in every marriage market. A key aspect of the Gale-Shapley algorithm is that both sides of the market go through their ranking lists in opposite directions. In the men-proposing version, men propose to women in the order of their preference rankings from top to bottom. They start proposing to their most preferred woman, and continue to propose to women in the order of their ranking as long as their proposals are not accepted. By contrast, the provisional matches of women go from bottom from to top: every time a woman accepts a proposal, it is from a man who is better than her previous match. In the women-proposing version of the algorithm, the opposite obtains: women go from top to bottom, and men go from bottom to top.

Theorem 5.11. (*Gale and Shapley 1962*) *The outcome of the Gale-Shapley algorithm is a stable matching.*

Proof. Let μ be the output of the men-proposing Gale-Shapley algorithm. Men only propose to acceptable women, and women only accept offers from acceptable men. Therefore, μ is individually rational. Let $m \in M$ and $w \in W$ be such that $w \succ_m \mu(m)$. Then, m proposed to w in some iteration of the algorithm. Since $\mu(m) \neq w$, w accepted the proposal of some man m' with $m' \succ_w m$. Then, $\mu(w) \succcurlyeq_w m' \succ_w m$. Hence, (m, w) are not a blocking pair. Therefore, μ is stable. Q.E.D.

Exercise 5.12. Let $(M, W, \succcurlyeq)$ be a marriage market. Evaluate: (i) in a Pareto efficient matching of $(M, W, \succcurlyeq)$, someone must be matched to their most preferred partner; (ii) in a stable matching of $(M, W, \succcurlyeq)$, someone must be matched to their most preferred partner.

Exercise 5.13. Evaluate: The men- and women-proposing versions of the Gale-Shapley algorithm always result in the same stable matching.

Definition 5.14. *Consider a marriage market $(M, W, \succcurlyeq)$. The men are said to have **aligned preferences** if every man has the same preference relation, i.e., $\succcurlyeq_m = \succcurlyeq_{m'}$ for every $m, m' \in M$. Similarly, women are said to have aligned preferences if all of them have the same preference relation.*

Exercise 5.15. Let $(M, W, \succcurlyeq)$ be a marriage market. (i) How many matchings are in $\mathcal{S}(M, W, \succcurlyeq)$ if men have aligned preferences? (ii) How many matchings are in $\mathcal{S}(M, W, \succcurlyeq)$ if both men and women have aligned preferences? (iii) How many rounds does it take for the men-proposing Gale-Shapley algorithm to converge if men have aligned preferences? (iv) And if (only) women have aligned preferences? (v) And if both men and women have aligned preferences?

Exercise 5.16. Evaluate: (i) if there is a unique stable matching in a marriage market, then both sides have aligned preferences. (ii) If there is a unique stable matching, then at least one side has aligned preferences.

Next, we show that, even though finding stable matchings is not a simple problem at first sight, randomly breaking blocking pairs actually leads to a stable matchings with probability one.

Theorem 5.17 (Roth and Vande Vate 1990)**.** *Let μ be an arbitrary matching in $(M, W, \succcurlyeq)$. There exists a finite sequence of matchings $\mu_1, \mu_2, \ldots, \mu_k$, such that $\mu_1 = \mu$, μ_k is stable, and for each $i = 1, \ldots, k-1$, there is a blocking pair (m_i, w_i) for μ_i such that μ_{i+1} is obtained from μ_i by satisfying the blocking pair (m_i, w_i).*

Proof. Let $\mu \in \mathcal{M}(M, W)$. If μ is stable, we are finished. Otherwise, select a nonempty set of agents $S \subseteq M \cup W$ such that there are no blocking pairs for μ contained in S, and μ does not match any agent in S to any agent not in S. (For example, S may contain a pair of agents matched under μ, or a single agent for that matter.) Select an agent not in S, say, woman w. If no man in S is part of a blocking pair with w, simply add w to S and do not change matching μ. Otherwise, select the man m in S whom woman w prefers the most among all the ones with whom she forms a blocking pair. Update the matching μ by matching woman w and man m (and breaking any other matches they were involved in). If there is a woman w' in S to whom m used to be matched, she may form a new blocking pair with some other man m' in S. If so, choose the blocking pair most preferred by w' to form the new matching. Continue this process within the set $S \cup \{w\}$, where women "propose" to men as in the deferred acceptance algorithm. On each stage, update the matching by forming each subsequent blocking pair. By the same reason for which the deferred acceptance algorithm converges to a stable matching, the process will terminate with a matching μ_i in which there are no blocking pairs within $S_i = S \cup \{w\}$. Now, continue the process iteratively, with the set S_i growing at each stage. Since the selected set has no blocking pairs at the end of each stage, the process will eventaully converge to astable matching when $S_k = M \cup W$. Q.E.D.

5.2 Opposition of interests

Theorem 5.18 (Gale and Shapley 1962)**.** *Let μ_M and μ_W be the outcomes of the men- and women-proposing Gale-Shapley algorithms, respectively. Then, for every $\mu \in \mathcal{S}(M, W, \succcurlyeq)$,*

$$\begin{aligned} &\forall m \in M, \quad \mu_M(m) \succcurlyeq_m \mu(m) \succcurlyeq_m \mu_W(m); \\ &\forall w \in W, \quad \mu_W(w) \succcurlyeq_w \mu(w) \succcurlyeq_w \mu_M(w). \end{aligned}$$

Proof. We prove first that $\mu_M(m)$ is the best partner for m out of

$$A_m = \{w \in W : \exists\, \mu \in S(M, W, \succcurlyeq) \text{ s.t. } w = \mu(m)\},$$

the set of women who are matched to m in some matching in $S(M, W, \succcurlyeq)$.

If μ_M does not give every man m his best partner from A_m, then there must be a first step in the Gale-Shapley algorithm in which a man m proposes to a woman

in A_m and is rejected. Before that step, all men who have been rejected, have been so by women the are never matched to in a stable matching.

Let m be this first man, and suppose that he is rejected by $w \in A_m$. Let m' be the man accepted by w instead of m: so $m' \succ_w m$. Recall that m's proposal to w is the first proposal to a stable partner rejected in the course of the algorithm. So m''s proposals to women at least as good as w cannot have been to any women in $A_{m'}$. Thus w is at least as good as any partner in $A_{m'}$.

Now, since we have assumed that $w \in A_m$, there is a matching $\mu \in S(M, W, \succcurlyeq)$ such that $w = \mu(m)$. We have established that w must be at least as good for m' as every partner in $A_{m'}$, including $\mu(w')$. So $w \succ_{m'} \mu(m')$. We also have that $m' \succ_w m = \mu(w)$. Therefore (m', w) for a blocking pair to μ, which yields a contradiction.

We now turn to the proof of the statement that μ_M is the worst matching for women. Let $\mu \in S(M, W, \succcurlyeq)$ and suppose (towards a contradiction) that there is some w for which $\mu_M(w) \succ_w \mu(w)$. By the result we have shown for μ_M, and because preferences are linear orders, we know that $\mu_M(w) \succ_{\mu_M(w)} \mu(\mu_M(w))$, i.e., whoever is matched with w under μ_M, $\mu_M(w)$, ranks $\mu_M(w)$ above whoever they are matched with in any other stable matching, in particular, μ. In other words, there exists a man m such that $w \succ_m \mu(m)$ and $m \succ_w \mu(w)$. This means that (m, w) are a blocking pair for μ, contradicting that $\mu \in S(M, W, \succcurlyeq)$. Q.E.D.

Theorem 5.18 shows that there are two matchings in the set of stable matchings, μ_M and μ_W, over which men and women disagree. Every man prefers their partner under matching μ_M over the one they get under every other stable matching. Moreover, all men agree that the worst matching among all the stable matchings is μ_W. Women have the opposite preferences: all of them find their partner under μ_W as the best partner under every stable matching, and the one under μ_M as the worst. Furthermore, Theorem 5.18 also shows that these two matchings, μ_M and μ_W, are precisely the ones generated by the Gale-Shapley algorithm when either of the two sides proposes. The matchings μ_M and μ_W are known as the **M-optimal** and **W-optimal** stable matchings, respectively. As we shall see in the rest oft his subsection, this "opposition of interest" goes beyond these two matchings.

Define the binary relations $\succ_M$ and $\succ_W$ over $\mathcal{M}(M, W)$ as follows:

- $\mu \succ_M \mu'$ if $\mu(m) \succcurlyeq_m \mu'(m)$ for all $m \in M$;

- $\mu \succ_W \mu'$ if $\mu(w) \succcurlyeq_w \mu'(w)$ for all $w \in W$.

Exercise 5.19. Evaluate: the binary relations $\succ_M$ and $\succ_W$ are complete and transitive.

Theorem 5.20 (Knuth 1976)**.** *If μ and μ' are stable matchings, then $\mu \succ_M \mu'$ if and only if $\mu' \succ_W \mu$.*

Proof. Let μ and μ' be stable matchings such that $\mu \succ_M \mu'$. Towards a contradiction, suppose it is not true that $\mu' \succ_W \mu$. Then, there exists $w \in W$ such that $\mu(w) \succ_w \mu'(w)$. Then, man $m = \mu(w)$ is matched to another woman under μ', which he prefers less to w (since $\mu \succ_M \mu'$). Then, we have $m \succ_w \mu'(w)$ and $w \succ_m \mu'(m)$, meaning that (m, w) block μ', which is a contradiction. Q.E.D.

Theorem 5.20 states that all men agree on the ranking of two stable matchings if and only if all women agree in the opposite direction. In this sense, the opposition of interest goes beyond the "extreme" stable matchings μ_M and μ_W. It permeates the whole set of stable matchings.

Definition 5.21. *Let $(M, W, \succcurlyeq)$ be a marriage market. The set of **stable partners** of $i \in M \cup W$, denoted by P_i, is the set of agents who i is matched to in some stable matching, i.e.,*

$$P_i = \{j \in M \cup W : \exists\, \mu \in \mathcal{S}(M, W, \succcurlyeq), \mu(i) = j\}\,.$$

Theorem 5.22. *Consider the matching formed by matching every woman to their most preferred stable partner. The resulting matching is well-defined, stable, and equal to the W-optimal matching. Furthermore, the same matching results from matching every man to their least preferred stable partner.*

Exercise 5.23. Prove Theorem 5.22.

The above results suggest that stable matchings can be ordered. To formalize this idea, for any two matchings μ, μ', define the **join** of μ and μ' as the matching $\mu \vee_M \mu'$ such that, for every $m \in M$ and $w \in W$,

$$\mu \vee_M \mu'(m) = \begin{cases} \mu(m) & \text{if } \mu(m) \succ_m \mu'(m) \\ \mu'(m) & \text{if } \mu(m) \succ_m \mu'(m) \\ \mu(m) & \text{otherwise} \end{cases} \quad \& \quad \mu \vee_M \mu'(w) = \begin{cases} \mu'(w) & \text{if } \mu(w) \succ_w \mu'(w) \\ \mu(m) & \text{if } \mu'(w) \succ_w \mu(w) \\ \mu(m) & \text{otherwise} \end{cases}$$

Note that $\mu \vee_M \mu'(i)$ stands for the agent matched with i in matching $\mu \vee_M \mu'$. Similarly, define the **meet** of μ and μ' as the matching $\mu \wedge_M \mu'$ such that, for every $m \in M$ and $w \in W$,

$$\mu \wedge_M \mu'(m) = \begin{cases} \mu'(m) & \text{if } \mu(m) \succ_m \mu'(m) \\ \mu(m) & \text{if } \mu(m) \succ_m \mu'(m) \\ \mu(m) & \text{otherwise} \end{cases} \quad \& \quad \mu \wedge_M \mu'(w) = \begin{cases} \mu(w) & \text{if } \mu(w) \succ_w \mu'(w) \\ \mu'(m) & \text{if } \mu'(w) \succ_w \mu(w) \\ \mu(m) & \text{otherwise} \end{cases}$$

We can define $\mu \vee_W \mu'$ and $\mu \wedge_W \mu'$ analogously.

A simple way to visualize the join (or meet) of two matchings, say $\mu \vee_M \mu'$, is to think of each man m as pointing to his most preferred woman between $\mu(m)$ and $\mu'(m)$. And each woman w pointing to her least preferred man between $\mu(w)$ and $\mu'(w)$. Then, match every man to the woman they are pointing to and every woman to the man they are pointing to. The first question to ask is whether a matching defined in this way is well-defined. That is, is it the case that a man always points to a woman who is pointing at him? The Theorem below shows that this is indeed the case when μ and μ' are stable matchings. However, it is not true in general.

Exercise 5.24. Show with an example that the join and meet between two (non-stable) matchings may fail to be a matching.

Theorem 5.25 (Conway)**.** *If μ and μ' are stable matchings, then both $\mu \vee_M \mu'$ and $\mu \wedge_M \mu'$ are stable matchings.*

Proof. First, we show that $\mu \vee_M \mu'$ is a matching by showing $\mu \vee_M \mu'(m) = w$ if and only if $\mu \vee_M \mu'(w) = m$. (i) Let $\mu \vee_M \mu'(m) = w$. Wlog, assume $w = \mu(m)$, so that $w \succ_m \mu'(m)$. We want to show that $\mu \vee_M \mu'(w) = m$, i.e., that $\mu'(w) \succ_w m$. Note that this must be the case, since $m \succ_w \mu'(w)$ would imply that (m, w) block μ'.

(ii) Let $\mu \vee_M \mu'(w) = m$. We show that $\mu \vee_M \mu'(m) = w$. Let M' be the set of men who are matched to a woman in at least one of matchings μ and μ', i.e.,

$$M' = \{m \in M : \mu \vee_M \mu'(m) \in W\}.$$

Define the set W' as the set of women who are matched to some man under both μ and μ', i.e.,

$$W' = \{w \in W : \mu \vee_M \mu'(w) \in M\}.$$

We show that $W' = \mu \vee_M \mu'(M')$, where $\mu \vee_M \mu'(M')$ denotes the set of women w such that $w = \mu \vee_M \mu'(m)$ for some $m \in M'$. First, we show (a) $\mu \vee_M \mu'(M') \subseteq W'$, and, second, we show (b) $|W'| \leq |\mu \vee_M \mu'(M')|$.

(a) Let $w \in \mu \vee_M \mu'(M')$, then $w = \mu \vee_M \mu'(m)$ for some $m \in M'$. By (i) above, $m = \mu \vee_M \mu'(w)$, which implies $w \in W'$.

(b) First, note that $|W'| = |\mu(W')|$. Now, we show (b.1) $|\mu(W')| \leq |M'|$, and (b.2) $|M'| = |\mu \vee_M \mu'(M')|$ to conclude this part of the proof. To show (b.1), actually note that $\mu(W') \subseteq M'$ since $m \in \mu(W')$ implies $m = \mu(w)$ for some $w \in W' \subseteq W$. To prove (b..2), note that $\mu \vee_M \mu'(m) = \mu \vee_M \mu'(m') = w$ implies $m = m'$ by (i) above (hence the mapping $M' \mapsto \mu \vee_M \mu'(M')$ is one-to-one).

Recall that we have $\mu \vee_M \mu'(w) = m$, and we want to show $\mu \vee_M \mu'(m) = w$. By hypothesis, $w \in W'$, so by what we just proved there exists $m' \in M'$ such that $\mu \vee_M \mu'(m') = w$. To conclude this part of the proof, we show that $m = m'$. Towards a contradiction, assume $m \neq m'$. Wlog, assume $m = \mu(w)$ and $m' = \mu'(w)$. Then, $m' = \mu'(w) \succ_w \mu(m) = m$ and $w' = \mu'(m) \succ_{m'} \mu(m')$, which is a contradiction since (m', w) would block μ. Therefore, $\mu \vee_M \mu'$ is a matching. By an analogous proof, one can show that $\mu \wedge_M \mu'$ is also a matching.

To finalize the proof of the theorem, we show that $\mu \vee_M \mu'$ is stable. Towards a contradiction, suppose (m, w) block $\mu \vee_M \mu'$. Then, $w \succ_m \mu \vee_M \mu'(m)$, which implies $w \succ_m \mu(m)$ and $w \succ_m \mu'(m)$. Similarly, $m \succ_w \mu \vee_M \mu'(w)$. Hence, if $\mu \vee_M \mu'(w) = \mu(w)$, (m, w) block μ. Alternatively, if $\mu \vee_M \mu'(w) = \mu'(w)$, they bock μ'. Either way, we obtain a contradiction since both μ and μ' are stable. The symmetric argument shows that $\mu \wedge_M \mu'$ is also stable. Q.E.D.

Theorem 5.25 characterizes the structure of the set of stable matchings. Namely, it shows that whenever we have two stable matchings, μ and μ', we can find another two stable matchings by following a simple procedure. If we match every man m to their most preferred woman between $\mu(m)$ and $\mu'(m)$, this results in another stable matching, denoted $\mu \vee_M \mu'$. Similarly, the matching obtained by matching every man to their least preferred woman between $\mu(m)$ and $\mu'(m)$ is also stable, and denoted $\mu \wedge_M \mu'$. The results above also imply that the opposite obtains if we follow the same procedure for women. That is, matching every woman w to their most

preferred man between $\mu(w)$ and $\mu'(w)$ yields matching $\mu \wedge_M \mu'$, and to their least preferred one yields $\mu \vee_M \mu'$.

Exercise 5.26. Show that if there is a unique stable matching, then $\mu_M = \mu_W = \mu_M \vee_M \mu_W = \mu_M \wedge_M \mu_W$.

Exercise 5.27. Consider a marriage market. Assume there exists a man and a woman that are matched to each other in the output of both the man- and woman-proposing Gale-Shapely algorithm. Show that they are matched to each other in every stable matching.

Exercise 5.28. Show that if an agent is single in one stable matching, then they are single in every stable matching. That is, the set of unmatched agents is the same in every stable matching.

In more formal terms, Theorem 5.25 shows that the set of stable matchings has a *lattice* structure. While we define formally the notion of a lattice in the next subsection, it is fairly simple to illustrate. A lattice is a partially ordered set in which every pair of elements has a "join" and a "meet", which are also elements of the lattice. Figure 2 presents several examples of lattice structures. In each of the examples, the order is sideways: from left to right, or right to left. At the extremes, each lattice has the M-optimal and W-optimal stable matchings. The matchings in between may or may not be ordered within themselves. For example, in the middle-right example with five stable matchings, matching μ_1 is more preferred by all men (and less preferred by all women) to matchings μ_2 and μ_2'. Indeed, $\mu_1 = \mu_2 \vee_M \mu_2'$. However, matchings μ_2 and μ_2' are not ordered, meaning that not every man (or every woman) agrees on which one is better.

Exercise 5.29. Give an example of a marriage market in which the set of stable matchings contains four matchings and has the lattice structure in the upper-right corner of Figure 2.

5.3 Stable matchings as fixed points

A **partially ordered set (poset)** is a set endowed with a **partial order**, which is a reflexive, antisymmetrtic, and transitive binary relation.[8] Given a poset, $(X, \leq)$, (i)

[8] A binary relation is reflexive if every element is related to itself.

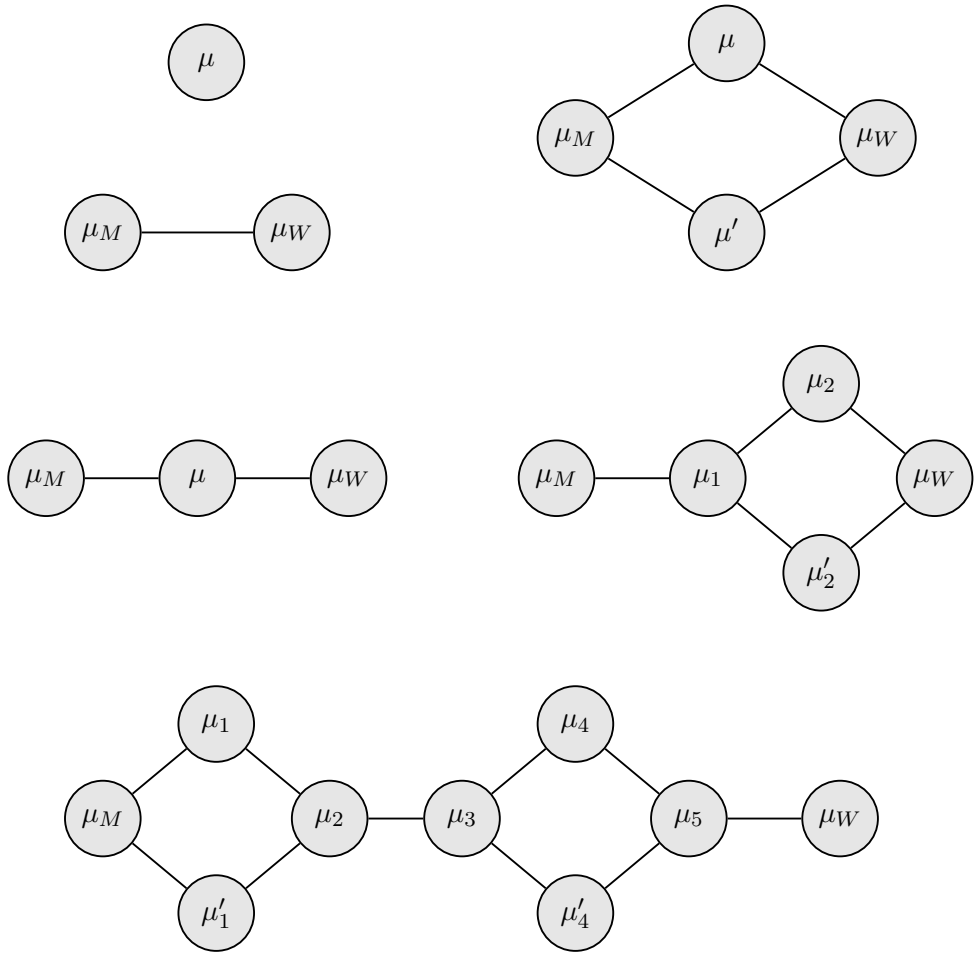

Figure 2: Examples of lattice structures

the **upper bound** of a subset $S \subseteq X$ is an element $u \in X$ such that $s \leq u$ for every $s \in S$; (ii) the **least upper bound** or **join** of $S \subseteq X$ is the element $\vee S \in X$ such that $\vee S$ is an upper bound of S and $\vee S \leq u$ for every upper bound u of S; (iii) the **lower bound** of a subset $S \subseteq X$ is an element $l \in X$ such that $l \leq s$ for every $s \in S$; (iv) the **greatest lower bound** or **meet** of $S \subseteq X$ is the element $\wedge S \in X$ such that $\wedge S$ is a lower bound of S and $l \leq \wedge S$ for every lower bound l of S.

The join of a two-element subset $\{x, y\} \subseteq X$ is denoted by $x \vee y$; its meet by $x \wedge y$. A poset $(X, \leq)$ is called a **lattice** if every two-element subset $\{x, y\} \subseteq X$ has a join and a meet. If every subset of the poset has a join and a meet, then it is called a **complete lattice**.

Exercise 5.30. Show that the poset $([0,1], \leq)$, where $[0,1]$ is the unit-interval in the real line and $\leq$ the usual "less than or equal" relation on the reals, is a lattice.

Exercise 5.31. Define $\leq$ on $\mathbb{R}^2$ as follows. For $x, y \in \mathbb{R}^2$, let $x \leq y$ if $x_1 \leq y_1$ and $x_2 \leq y_2$, where $x = (x_1, x_2)$ and $y = (y_1, y_2)$. (i) Show that the poset $([0,1]^2, \leq)$ is a lattice, where $x \in [0,1]^2$ if and only if $0 \leq \min\{x_1, x_2\}$ and $\max\{x_1, x_2\} \leq 1$. (ii) Let $A \subseteq \mathbb{R}^2_+$ be such that $(x, y) \in A$ if and only if $x + y \leq 1$. Show that $(A, \leq)$ is *not* a lattice.

Theorems 5.11 and 5.25 imply that the set of stable matchings $\mathcal{S}(M, W, \succcurlyeq)$ and the binary relation $\succ_M$ (or $\succ_W$) form a nonempty lattice. In this section, we will obtain the same result via a fixed-point argument.

Definition 5.32. *A **prematching** is a function $\nu : M \cup W \to M \cup W$ such that for all $m \in M$ and $w \in W$, (i) $\mu(m) \in W \cup \{m\}$, and (ii) $\mu(w) \in M \cup \{w\}$.*

Prematchings are also known as **fantasies** since a man (or woman) can be prematched to a woman (or man) that is not prematched to them. Similarly, in a prematching, two men (or women) can be prematched to the same woman (or man). A simple way to picture fantasies is to imagine them as every man pointing to a woman and every woman pointing to a man. While matchings must to be reciprocal (if a man points to a woman, she must be pointing back at him), fantasies need not be reciprocal. All matchings are fantasies, but not conversely.

For prematching ν, $m \in M$ and $w \in W$, define the following sets:

$$A(m, \nu) = \{w \in W : m \succcurlyeq_w \nu(w)\},$$
$$A(w, \nu) = \{m \in M : w \succcurlyeq_m \nu(m)\}.$$

That is, $A(m, \nu)$ is the set of women willing to match with m, given their fantasy in ν. Define a function T mapping fantasies to fantasies, by $(T\nu)(m)$ being the optimal choice for $\succcurlyeq_m$ in $A(m, \nu) \cup \{m\}$, for any $m \in M$; and similarly by $(T\nu)(w)$ being the optimal choice for $\succcurlyeq_w$ in $A(w, \nu) \cup \{w\}$, for any $w \in W$. Under fantasy $T\nu$, every man points to his most preferred woman among the ones who were willing to match with him under fantasy ν. And the same for women. Finally, for any two fantasies

ν and ν', say that ν is **less than** ν', denoted by $\nu \leq \nu'$ if

$$\forall m \in M, \quad \nu'(m) \succcurlyeq_m \nu(m);$$
$$\forall w \in W, \quad \nu(w) \succcurlyeq_w \nu'(w).$$

Lemma 5.33. *T is monotone increasing, $\nu \leq \nu'$ implies $T\nu \leq T\nu'$.*

Proof. Let $\nu \leq \nu'$. We show that $A(m,\nu) \subseteq A(m,\nu')$ and $A(w,\nu) \supseteq A(w,\nu')$, for all m and w. To see this, note that if $w \in A(m,\nu)$ then

$$m \succcurlyeq_w \nu(w) \succcurlyeq_w \nu'(w),$$

so $w \in A(m,\nu')$. Similarly for $A(w,\nu) \supseteq A(w,\nu')$. Since the best element from a larger set cannot be worse than from a smaller, $(T\nu')(m) \succcurlyeq_m (T\nu)(m)$ and $(T\nu)(w) \succcurlyeq_w (T\nu')(w)$. Thus $T\nu \leq T\nu'$. Q.E.D.

Lemma 5.34. *A matching μ is stable if and only if it is a fixed point of T.*

Proof. ($\Rightarrow$) Let μ be a stable matching. We must show that $\mu = T\mu$. Suppose, by way of contradiction, that there is an element of $A(m,\mu) \cup \{m\}$ that is strictly better than $\mu(m)$. By individual rationality, $\mu(i) \succcurlyeq_i i$ for all $i \in M \cup W$. So this element cannot be m. Suppose that $w \in A(m,\mu)$ is such that $w \succ_m \mu(m)$. Then, $w \neq \mu(m)$, which implies that $m \neq \mu(w)$ as μ is a matching. Then, $w \in A(m,\mu)$ implies that $m \succ_w \mu(m)$ because $\succcurlyeq$ is a strict preference. Then (m,w) would form a blocking pair, a contradiction.

($\Leftarrow$) First, we show that a fixed point is a matching, not just a prematching. Let $\nu = T\nu$. Suppose, by way of contradiction, that there is (m,w) with $w = \nu(m)$ and $m \neq \nu(w)$. Then, $w \in A(m,\mu)$ so $m \succcurlyeq_w \nu(w)$. Since $m \neq \nu(w)$, $m \succ_w \nu(w)$. But $m \in A(w,\mu)$ since $\nu(m) = m$. A contradiction, as ν is a fixed point of T.

Second, we show that ν is stable. By construction, ν is individually rational since $\nu(i) = (T\nu)(i) \succcurlyeq_i i$. If $w \succ_m \nu(m)$ then $m \in A(w,\nu)$, which implies $\nu(w) = (T\nu)(w) \succ_w m$ since $\nu(w) \neq m$. So (m,w) are not a blocking pair. Q.E.D.

In light of the previous lemmas, the existence of a stable matching and the underlying lattice structure of the set of stable matchings can be obtained as a consequence of the following fixed-point theorem.

Theorem 5.35 (Tarski's fixed point theorem)**.** *The set of fixed points of a monotone function on a lattice is a nonempty and complete lattice.*

Finally it is worth pointing out that this formalism also leads to a natural algorithm for finding a stable matching. Let $\bar{\nu}$ be a prematching where $\bar{\nu}(w) = w$ for all w and $\nu(m)$ is the best alternative in $W \cup \{m\}$ for all m. Define a sequence of prematchings ν^k by letting $\nu^0 = \bar{\nu}$ and $\nu^{k+1} = T\nu^k$. Then, there is K such that $\mu = \nu^K$ is a stable matching. To observe why the sequence ν^k converges to a stable matching, note that by monotonicity

$$T(T\bar{\nu}) \leq T\bar{\nu} \leq \bar{\nu}.$$

Hence, we obtain a decreasing sequence

$$\cdots \nu^{k+1} \leq \nu^k \leq \cdots \leq \bar{\nu}.$$

Since the set of prematchings is finite, the sequence must eventually be constant. So there is K large enough such that $T\nu^k = \nu^{k+1} = \nu^k$, a fixed point of T. In fact, the sequence converges to the M-optimal stable matching. By defining $\bar{\nu}$ analogously, we reach the W-optimal stable matching.

5.4 Incentives in the marriage market

Given two finite and disjoint sets M and W, let $\mathcal{P}$ be the set of all strict preference profiles $\succcurlyeq$ such that $(M, W, \succcurlyeq)$ is a matching market. Given $\succcurlyeq \in \mathcal{P}$, extend each $\succcurlyeq_i$ to a preference over $\mathcal{M}(M, W)$ by $\mu \succcurlyeq_i \mu'$ if and only if $\mu(i) \succcurlyeq_i \mu'(i)$.

A **matching mechanism** is a function $\phi : \mathcal{P} \to \mathcal{M}(M, W)$. A matching mechanism is **strategy-proof** if, for every $\succcurlyeq, \succcurlyeq' \in \mathcal{P}$, $i \in M \cup W$, and $\succcurlyeq_i'$,

$$\phi(\succcurlyeq) \succcurlyeq_i \phi(\succcurlyeq_i', \succcurlyeq_{-i}).$$

A matching mechanism ϕ is **Pareto efficient** if for every $\succcurlyeq \in \mathcal{P}$, $\phi(\succcurlyeq)$ is Pareto efficient in $(M, W, \succcurlyeq)$. A matching mechanism ϕ is **stable** if for every $\succcurlyeq \in \mathcal{P}$, $\phi(\succcurlyeq) \in \mathcal{S}(M, W, \succcurlyeq)$.

Theorem 5.36. *There is an efficient and strategy-proof mechanism, but there is no stable and strategy-proof mechanism.*

Proof. Running a serial dictatorship is strategy-proof and Pareto efficient. Indeed, let $M = \{m_1, \ldots, m_n\}$. For each $\succcurlyeq$, let $\mu(m_1)$ be the top alternative in $\succcurlyeq_{m_1}$; let $\mu(m_2)$ be the top alternative in $\succcurlyeq_{m_2}$, once $\mu(m_1)$ is no longer available; let $\mu(m_3)$ be the top alternative in $\succcurlyeq_{m_3}$, once $\mu(m_1)$ and $\mu(m_2)$ are no longer available; and so on. The mechanism does not depend on the women's preferences. It is strategy-proof because no man can obtain anything better for himself than reporting the truth, given what the other men are reporting.

To see that there is no strategy-proof stable mechanism, consider the following particular market. Let $M = \{m_1, m_2\}$ and $W = \{w_1, w_2\}$. Let $\succcurlyeq_{m_i}$ rank w_i over w_{3-i} over m_i. Let $\succcurlyeq_{w_i}$ rank m_{3-i} over m_i. There are two stable matchings: $\mu_M(m_i) = w_i$ for $i = 1, 2$, and $\mu_W(m_i) = w_{3-i}$ for $i = 1, 2$. Let ϕ be a stable mechanism. Then, $\phi(\succcurlyeq)$ must coincide with either μ_M or μ_W. Say wlog that it coincides with μ_M.

Consider the preference $\succcurlyeq'_{w_1}$ that ranks m_2 over w_1 over m_1; thus making m_1 unacceptable. There is a single stable matching in $(M, W, (\succcurlyeq_{m_1}, \succcurlyeq_{m_2}, \succcurlyeq'_{w_1}, \succcurlyeq_{w_2}))$ and it coincides with μ_W. So ϕ is not strategy-proof as $m_2 = \mu_W(w_1) \succ_{w_1} \mu_M(w_1) = m_1$. Q.E.D.

Theorem 5.37. *The men-proposing Gale-Shapley mechanism is (group) strategy-proof for the men.*

Exercise 5.38. Read the proof of Theorem 5.37 in Chapter 4 of Roth and Sotomayor (1990).

5.5 Marriage with transferable utility

We turn to a model of a two-sided matching market with transfers due to Shapley and Shubik (1971). We stick to their original formulation in terms of *buyers* and *sellers*. Nonetheless, note that buyers and sellers can be thought of as men and women who divide the "spoils of marriage."

Definition 5.39. *A **matching market with transfers** is a tuple (B, S, α), where B and S are finite, nonempty, and disjoint sets of **buyers** and **sellers**, and $\alpha = (\alpha_{ij})_{i \in B, j \in S}$ is a **surplus matrix** with $\alpha_{ij} \geq 0$ for all $(i, j) \in B \times S$.*

Each buyer seeks to buy one and only one unit of an indivisible good. Each seller has one unit to sell, but sellers are different from each other, and offer potentially

different goods. If $i \in B$ buys from $j \in S$, they generate surplus α_{ij}. You may think of α_{ij} as the sum of the payoffs of i and j from exchanging (or marrying) with one another. When i buys from j, i obtains some utility u_i and j gets some profit v_j, so that $u_i + v_i = \alpha_{ij}$. For example, if the cost for j is zero, i's utility is $u_i = \alpha_{ij} - v_j$, where v_j is the price (and therefore the profit) that j gets from i. Hence, i buys from j if $\alpha_{ij} - v_j \geq \alpha_{is} - v_s$ for all $s \in S$. Similarly, j sells to i if $\alpha_{ij} - u_i \geq \alpha_{bj} - u_b$ for all $b \in B$.

Definition 5.40. *A **matching** is a matrix $x = (x_{ij})_{i \in B, j \in S}$ such that, for all $(i,j) \in B \times S$, $x_{ij} \geq 0$,*

$$\sum_{s \in S} x_{is} \leq 1 \quad \textit{and} \quad \sum_{b \in B} x_{bj} \leq 1.$$

If $x_{ij} \in \{0,1\}$ we can interpret $x_{ij} = 1$ as i buying from j, and $x_{ij} = 0$ as i *not* buying from j. The above inequalities mean that each buyer buys from at most one seller, and each seller sells to at most one buyer. However, in principle, x_{ij} may be anywhere in the interval $[0,1]$. Matchings in which with some $x_{ij} \in (0,1)$ are called **fractional matchings**. As we shall see later on, even though fractional matchings are possible in this model, we will be able to rule them out and focus on matchings with $x_{ij} \in \{0,1\}$ for every $(i,j) \in B \times S$.

Definition 5.41. *An **assignment** is a pair of vectors $u = (u_i)_{i \in B}$ and $v = (v_j)_{j \in S}$ such that $u_i \geq 0$, $v_j \geq 0$, and there exists a matching x satisfying $\sum_{i \in B} u_i + \sum_{j \in S} v_j = \sum_{i \in B, j \in S} \alpha_{ij} x_{ij}$. In such case, we say that matching x **supports** assignment (u,v).*

An assignment is a redistribution of the total surplus in the economy through a matching. Note that one matching may support distinct assignments. This is because a matching only incorporates information about who trades with whom, while an assignment consists of the net payoffs (after transfers) of every agent. Similarly, an assignment can be supported by distinct matchings since buyers and sellers may reach the same utilities by trading with different parties.

Definition 5.42. *An assignment (u,v) is in the **core** if $u_i + v_j \geq \alpha_{ij}$ for every $(i,j) \in B \times S$.*

The notion of the core arises from not allowing blocking pairs. A pair (i,j) can **block** an assignment if $u_i + v_j < \alpha_{ij}$ by trading amongst themselves and sharing the

surplus α_{ij}. Assignments in the core can also be seen in terms of individual payoff optimisation. That is, (u, v) is in the core if and only if, for all $(i, j) \in B \times S$,

$$u_i = \max\{\alpha_{is} - v_s : s \in S\}, \quad \text{and} \quad v_j = \max\{\alpha_{bj} - u_b : b \in B\}.$$

We focus on the relationship between two key notions. On the one hand, we consider efficient matchings, the ones that maximise the total surplus of the economy. On the other hand, we consider core assignments. As we shall see, both are dual notions: every core assignment is supported by an efficient matching, and every efficient matching supports a core assignment.[9]

Definition 5.43. *A matching x is **efficient** if it maximizes the total surplus in the economy; that is, if it solves the following problem:*

$$\begin{aligned}
(\star) \qquad & \max_{(x_{ij})} \sum_{i \in B, j \in S} x_{ij}\alpha_{ij} \\
\text{subject to} \quad & x_{ij} \geq 0 \qquad \forall\, (i,j) \in B \times S \\
& \sum_{j \in S} x_{ij} \leq 1 \quad \forall\, i \in B \\
& \sum_{i \in B} x_{ij} \leq 1 \quad \forall\, j \in S.
\end{aligned}$$

Efficient matchings always exist. Problem $(\star)$ is a maximization of a continuous function over a compact set. The existence of a maximizer follows from the Weierstrass Theorem. The following result states formally what we mean by efficient matchings and core assignments to be dual. As we shall see, the notion of duality comes from inspecting the dual problem of the linear program $(\star)$.

Theorem 5.44 (Shapley and Shubik 1971)**.** *For every efficient matching x, there exists a core assignment (u, v) such that $\sum_i u_i + \sum_j v_j = \sum_{i,j} \alpha_{ij} x_{ij}$. Likewise, every core assignment is supported by an efficient matching.*

Proof. The proof relies on analyzing the model by means of linear programming duality. Concretely, we show that there is a duality between problem $(\star)$, and that of finding a core assignment.

[9] Note how we recover a type of Second Welfare Theorem when we introduce transfers to a matching model.

We set up the Lagrangean of ($\star$), and use the minmax theorem. Let u_i be the Lagrange multiplier associated to the constraint that $\sum_{j\in S} x_{ij} \leq 1$. Let v_j be the one associated to $\sum_{i\in B} x_{ij} \leq 1$. Then, the Lagrangean is given by:

$$\mathcal{L}(x;(u,v)) = \sum_{i\in B, j\in S} x_{ij}\alpha_{ij} + \sum_{i\in B} u_i \left(1 - \sum_{j\in S} x_{ij}\right) + \sum_{j\in S} v_j \left(1 - \sum_{i\in B} x_{ij}\right).$$

By the minmax theorem, we have

$$\max_x \min_{u,v} \mathcal{L}(x;(u,v)) = \min_{u,v} \max_x \mathcal{L}(x;(u,v)).$$

Note that

$$\begin{aligned}\mathcal{L}(x;(u,v)) &= \sum_{i,j} x_{ij}\alpha_{ij} + \sum_i u_i \left(1 - \sum_j x_{ij}\right) + \sum_j v_j \left(1 - \sum_i x_{ij}\right) \\ &= \sum_{i,j} x_{ij}\left(\alpha_{ij} - u_i - v_j\right) + \sum_i u_i + \sum_j v_j.\end{aligned}$$

Hence,

$$\max_x \min_{u,v} \mathcal{L}(x;(u,v)) = \min_{u,v} \max_x \sum_i u_i + \sum_j v_j + \sum_{i,j} x_{ij}(\alpha_{ij} - u_i - v_j).$$

Then, x_{ij} are Lagrange multipliers in the problem of minimizing $\sum_i u_i + \sum_j v_j$ subject to $(\alpha_{ij} - u_i - v_j) \leq 0$, which results in the **dual** problem:

($\star\star$)
$$\begin{aligned}&\min_{(u_i),(v_j)} \sum_{i\in B} u_i + \sum_{j\in S} v_j \\ \text{subject to}\quad & u_i + v_j \geq \alpha_{ij} \quad \forall (i,j) \in B \times S \\ & u_i \geq 0 \quad \forall i \in B \\ & v_j \geq 0 \quad \forall j \in S.\end{aligned}$$

By duality, the value functions coincide at the solutions of problems ($\star$) and ($\star\star$):

$$\sum_{i\in B} u_i + \sum_{j\in S} v_j = \sum_{i\in B, j\in S} x_{ij}\alpha_{ij}.$$

Thus, we have shown that for every efficient matching x, there exists a core assignment (u,v) such that $\sum_i u_i + \sum_j v_j = \sum_{i,j} \alpha_{ij}x_{ij}$. Duality between ($\star$) and ($\star\star$)

also implies that core assignments can only be supported by efficient matchings. To see this clearly, let (u, v) be the solution to $(\star\star)$, and x the efficient matching that supports it. Let (u', v') be another core assignment, and, towards a contradiction, assume it is supported by some matching x' that is *not* efficient. Since (u', v') is feasible in problem $(\star\star)$, we must have $\sum_i u'_i + \sum_j v'_j \geq \sum_i u_i + \sum_j v_j$. However, this implies $\sum_{i,j} \alpha_{ij} x'_{ij} \geq \sum_{i,j} \alpha_{ij} x_{ij}$, which implies either that x' is efficient or x is not efficient, a contradiction. Q.E.D.

Exercise 5.45. Note that the Lagrangean $\mathcal{L}(x; (u, v))$ in the proof of Theorem 5.44 does not include the non-negativity constraints $x_{ij} \geq 0$ for every $(i, j) \in B \times S$. Show that this omission does not affect the conclusion of the theorem.

In principle, nothing in Theorem 5.44 precludes the solution of $(\star)$ to be integer. The next result shows that this is indeed the case.

Proposition 5.46. *There is a solution to the surplus maximization problem in which $x_{ij} \in \{0, 1\}$ and where, if $x_{ij} = 1$, then $u_i + v_j = \alpha_{ij}$.*

Proof. The extreme points of the set of matchings consist of matrices with 0-1 values. The primal problem is a linear programming problem, so a solution always exists that is an extreme point. Furthermore, the complementary slackness conditions of $(\star\star)$ imply that

$$0 = \sum_{i,j} x_{ij}(\alpha_{ij} - u_i - v_j), \quad \text{and} \quad \alpha_{ij} - u_i - v_j \leq 0.$$

So $x_{ij} > 0$ implies $u_i + v_j = \alpha_{ij}$. Q.E.D.

Suppose that $\alpha_{ij} > 0$ for all i, j, and that $|B| = |S|$. Then, the matching constraints will hold with equality at a solution, and all agents will be matched to someone.

Notably, the structure of the core in a matching market with transfers is similar to the one of the set of stable matchings in a model without transfers. The next result shows that the core also has a lattice structure when we have transfers.

Proposition 5.47. *Let (u, v) and (u', v') be assignments in the core. Let $\bar{u}_i = \max\{u_i, u'_i\}$ and $\underline{v}_j = \min\{v_j, v'_j\}$. Then $(\bar{u}, \underline{v})$ is an assignment in the core.*

Proof. Note that for any $i \in B$ and $j \in S$, $\bar{u}_i + \underline{v}_j \geq \alpha_{ij}$ (since wlog $\underline{v}_j = v_j$, so $\bar{u}_i + \underline{v}_j \geq u_i + v_j \geq \alpha_{ij}$), and $\bar{u}_i, \underline{v}_j \geq 0$.

Choose one optimal matching x that is an extreme point. Then $x_{ij} = 1$ means that $u_i + v_j = u'_i + v'_j$. Then $\bar{u}_i = u'_i$ if and only if $\underline{v}_j = v'_j$. So $\bar{u}_i + \underline{v}_j = u_i + v_j$. Thus,

$$\sum_{i \in B} u_i + \sum_{j \in S} v_j = \sum_{i \in B, j \in S} x_{ij}(u_i + v_j) = \sum_{i \in B, j \in S} x_{ij}(\bar{u}_i + \underline{v}_j) = \sum_{i \in B} \bar{u}_i + \sum_{j \in S} \underline{v}_j.$$

Thus $(\bar{u}, \underline{v})$ satisfies the constraints of the dual program, and has the same value for the objective function. Q.E.D.

An analogous result is true if we take the maximum of v_j and v'_j and the minimum of u_i and u'_i. It means that buyers share some interests with other buyers, and sellers with other sellers. There are common interests for agents on the same side of the market, and opposing interest for agents on opposite side of the market. As a consequence, we have the following corollary.

Corollary 5.48. *There exists core assignments (u^*, v_*) and (u_*, v^*) such that for any core assignment (u, v), for every $i \in B$ and $j \in S$,*

$$u^*_i \geq u_i \geq u_{*i}$$
$$v^*_j \geq v_j \geq v_{*i}$$

Think of (u^*, v_*) and (u_*, v^*) as core assignments with minimal and, respectively maximal, prices. That is, these assignments may be thought of as the buyer- and seller-optimal assignments.

Notes

The seminal contribution is due to Gale and Shapley (1962), who developed the DA algorithm to prove nonemptiness of the set of stable matchings in a marriage market. They also established the existence of the M-optimal and W-optimal extremal stable matchings. Knuth (1976) further analyzed the structure of the set of stable matchings. They attribute Theorem 5.25 to John Horton Conway. In general, the discussion about the structure of the set of stable matchings draws heavily from Chapters 2 and 3 of Roth and Sotomayor (1990). The characterization of stable

matchings as fixed points of an increasing function is due to Adachi (2000). For a thorough treatment of the use of posets and lattices in economics, see the monograph by Topkis (1998) on monotone comparative statics. The brief discussion on the incentive properties of the DA algorithm is based on Chapter 4 of Roth and Sotomayor (1990). See that chapter for more results and detailed proofs. The model of matching with transferable utility is due to Shapley and Shubik (1971), who refer to the problem as The Assignment Game. Standard treatments on optimization with inequality constraints abound in the literature, e.g., Sundaram (1996) and de la Fuente (2000). However, linear programming and duality tend not to be included in the standard "toolkit" for economists. For a detailed treatment on the use of linear programming methods in economics, see Vohra (2005) (especially Chapter 4) or the monograph by Galichon (2016) on optimal transport methods (especially Appendix B).

Additional exercises

Exercise 5.49. A **roommate problem** is given by $(I, (\succcurlyeq_i)_{i\in I})$, where I is a finite set of agents and $\succcurlyeq_i \in \mathcal{P}(I)$ for every $i \in I$. A matching in this setting is a function $\mu : I \to I$ such that $\mu(i) = j$ if and only if $\mu(j) = i$ for every $i, j \in I$. Define what is a stable matching in this setting. Evaluate: the set of stable matchings in a roommate problem is non-empty.

Exercise 5.50. A **three-sided matching problem** is given by $(M, W, C, \succcurlyeq)$, where M, W, and C are finite, nonemtpy and disjoint sets of men, women and children, respectively. Furthermore, $\succcurlyeq$ includes a strict preference relation for every $i \in M \cup W \cup C$, where $\succcurlyeq_m \in \mathcal{P}((W \times C) \cup \{m\})$ for every $m \in M$, $\succcurlyeq_w \in \mathcal{P}((M \times C) \cup \{w\})$ for every $w \in W$, and $\succcurlyeq_c \in \mathcal{P}((M \times W) \cup \{c\})$ for every $c \in C$. That is, agents have preferences over pairs of agents from the other sides. Define a matching and stability in this economy. Show via a counterexample that the set of stable matchings may be empty.

Exercise 5.51. A **many-to-one matching market** is given by $(F, W, \succcurlyeq)$, where F and W are finite, noempty and disjoint sets of firms and workers, and $\succcurlyeq$ includes $\succcurlyeq_f \in \mathcal{P}(2^W)$ for every $f \in F$, and $\succcurlyeq_w \in \mathcal{P}(F)$ for every $w \in W$. That is, firms have preferences over sets of workers, while workers have preferences over individ-

ual firms.[10] Consider a notion of blocking between firms and subsets of workers. Define stability in this economy, and show via counterexample that the set of stable matchings may be empty. What assumption would you add to guarantee that stable matchings always exist?

[10]The power set of set X, denoted by 2^X, is the set containing all the subsets of X.

6 The medical match

In this section, we review one of the first and most most famous applications of market design methods: the assignment of medical interns into residency programs. To this date, more than 40,000 medical interns are allocated to more than 30,000 residency programs every year in the U.S. The assignment is done through a centralized clearinghouse, known as the National Resident Matching Program (NRMP), or simply known among doctors as "The Match." The success of the NRMP in the U.S. has led to the adoption of similar clearinghouses in other countries, such as Canada and the U.K.

The history of the NRMP is both an intellectual delight and an example of how economic theory can guide market design in practice, what Alvin Roth famously calls "economic engeneering" (2002). First, we will briefly review this history. The three main lessons to draw are: (i) the importance of stability as a condition for the survival of an institutional design; (ii) how real-life markets are shaped by a collection of regulations that are the result of trial-and-error and multiple idiosyncratic factors, and how all of these can at times result in desirable institutional designs, but also in market inefficiencies at others; (iii) how economists have a lot to learn from looking closely at how real-life markets work. Second, we will study closely the algorithm underlying the original design of The Match in the 1950s and its relation with the Gale-Shapley algorithm.

6.1 A brief history of unraveling

The system through which medical interns are allocated to medical residency programs in the U.S. underwent multiple changes in the 1940s. In 1951, it reached a design which persisted for more than four decades, until the end of the 1990s. At that time, prompted for calls for reform, a group led by economists undertook a further redesign of the system.

Until 1945, medical interns were assigned to residency programs in a decentralized fashion. As in typical entry-level labor markets, interns were free to apply to residency programs, which in turn accepted the applications from the interns of their preference. Medical residency programs (a.k.a. hospitals) would also seek the interns which they preferred the most and offered them binding agreements to

enrol in their programs upon graduation. By 1945, it was clear to the administrators of the Association of American Medical Colleges (AAMC) that the market suffered from what now economists refer to as *unraveling*.

Prior to the mid-1940s, hospitals competed in a typical "arms race" for the best medical interns. The main way in which this competition took place was through the dates of the binding agreements that hospitals offered students in order to "lock them" into their residency programs. By offering early binding agreements to students, hospitals tried to guarantee high-quality incoming classes. During the first decades of the twentieth century, hospitals offered binding agreements to students earlier and earlier in their careers. Initially, these biding agreements were signed a few months before students in the senior class graduated. However, as hospitals started undercutting each other's agreements, the dates at which students had to decide which residency program they would enrol upon graduation became all but absurd. By the mid-1940s, agreements were typically signed up to two years before students graduated from medical school. This was clearly inefficient. On the one hand, students did not know which program or specialty they would want to study after graduating. In some cases, they had not even taken the necessary classes to make up their minds. On the other hand, the earlier the agreements were offered by hospitals to students, the less hospitals knew about the quality and aptitudes of the students. Students who appeared to be very promising after two years of medical school, would turn out to be not so successful by the end of it. Given these clear inefficiencies, in 1945 the AAMC decided to stop the unraveling by imposing a minimum date at which medical schools were allowed to disclose student records to hospitals.

At first, the new minimum date served its purpose in that hospitals were not able to lock in students early on through binding agreements. However, another market inefficiency turned up. Given the chaotic application process, which consists of applications and interviews, it was common for hospitals to "jump the gun" and offer so-called "exploding offers" to students. In order to lock in students, hospitals would make offers to students with very short deadlines right after the date in which records were released. By forcing a student to decide quickly on whether or not to enrol in a residency program, a hospital minimised the probability that another hospital, which the student might prefer, would also made them an offer. Students faced tough career decisions. They could play it "safe" by accepting an early

offer, even if it was not from their most preferred hospital. Or they could "risk" it and decline such offers, in the hope that other hospitals which they preferred more would make them an offer later. Either way, students were likely to end up in a less preferred residency program while another program they liked better had an opening for them. Therefore, it became more and more common for students to back out from offers they had previously accepted, which clearly hospitals found annoying.

Between 1945 and 1951, the AAMC implemented a series of regulations which aimed to solve this problem. They largely consisted on regulating the time at which offers could be made, and the time which they should give interns to make up their minds. After some (unsuccessful) experimentation with different sorts of rules, in 1951, the AAMC resolved to fully centralize the process into a matching clearinghouse. Under the new procedure, students and hospitals would communicate and exchange information as before via interviews, but then both would submit rank-ordered lists of their preferences over the hospitals and applicants they were considering. The final allocation of interns to residency programs would be decided through a matching algorithm.

In 1951, the AAMC performed a trial-run of the new procedure. It was not used to actually match students and hospitals on that year, but as a basis for the next year. Despite some caveats, the trial-run was deemed to be successful, and the AAMC decided to fully implement the matching mechanism with a few tweaks the following year. A key aspect of the market that allowed for this type of organization was that the salaries and responsibilities for medical interns were mostly standard across all programs and not an important part of contract negotiations. Importantly, the matching procedure was to be voluntary. Students were free to opt out and contract directly with hospitals. The matching algorithm, which was used until the end of the 1990s, is known as the NIMP algorithm (for National Intern Matching Program), which used to be the name of the program at the time.

In a remarkable "discovery" of the economics discipline, some decades later it was noted that the NIMP and the Gale-Shapley algorithms (1962), though written distinctly, are actually equivalent. Notably, this was unknown to both the administrators of the NIMP, and to David Gale and Lloyd Shapley until the 1970s.[11] The

[11] According to anecdotal evidence reported by Roth (1984), it was until 1976 when David Gale first heard of the labor market for medical interns and sent a copy of Gale and Shapley (1962) to an administrator of the NIMP.

NIMP algorithm was used until the late 1990s. At the time, the mechanism faced strong opposition from students, who claimed, amongst other things, that the mechanism was open to "gaming." Furthermore, as years went by, it became more common for couples of medical interns to look for medical residency programs that were geographically close to one another. For this reason, a group of economists led by Alvin Roth undertook a partial redesign of the matching algorithm in the mid-1990s. Though the main aspects of the Gale-Shapley algorithm remain in place to date, the redesign focused on (i) changing the algorithm from the program-proposing to the applicant-proposing version of the Gale-Shapley algorithm, and (ii) the way in which the algorithm deals with couples who have interdependent preferences.

6.2 NIMP algorithm

In this subsection, we study formally the NIMP algorithm, as well as the algorithm used in the trial run of 1951. For simplicity, we restrict attention to the case in which every hospital has exactly one position. As discussed above, the algorithm was modified to incorporate several complaints brought up by students after the trial run. As we shall see below, the main concern was that the algorithm was not strategy-proof for students. But not only this, it was also not a stable mechanism. After adjusting the algorithm, AAMC administrators came up with the NIMP algorithm, which always generates a stable matching. At the time, it was mistakenly claimed that it also was strategy-proof for students. Remarkably, this appears to have remained unknown for several decades until economists studied the algorithm formally.

Definition 6.1. *A **one-to-one medical match** consists on a tuple $(S, H, \succcurlyeq)$, where S and H are finite, nonempty, and disjoint sets of **students** and **hospitals**, respectively. The preference profile $\succcurlyeq$ contains a linear order for each agent in the market over the agents on the other side (including the possibility of remaining single).*

Note that every one-to-one medical match is a marriage market, and vice versa. We denote a generic student by $s_j \in S$, and a generic hospital by $h_i \in H$. Also, for convenience, and according to the nature of the NRMP, we express linear orders as rank order lists (instead of preference relations).

Algorithm 6.2 (NIMP trial-run)**.** *Students submit a rank ordering of hospitals. Hospitals submit a ranking dividing student into five groups: rank 1, rank 2, ..., rank 5; each*

group containing as many students as the number of positions the hospital is offering. The algorithm proceeds in consecutive stages as follows.

- *1:1 stage. Students and hospitals are matched if they give each other a rank of 1.*
- *1:2 stage. The remaining students and hospitals are matched if the student has ranked the hospital 1 and the hospital has ranked the student 2.*
- *2:1 stage. Among the remaining students and hospitals, match students who ranked hospitals 2, and hospitals who ranked students 1.*
- *2:2 stage. . . . , followed by 1:3 stage, and so on.*

Proposition 6.3. *The NIMP trial-run algorithm is not stable, nor strategy-proof for students.*

Proof. Consider the following example with three students and three hospitals. The preferences are given as follows:

$$\begin{aligned} s_1 &: h_1, h_2, h_3; & h_1 &: s_2, s_3, s_1; \\ s_2 &: h_2, h_3, h_1; & h_2 &: s_1, s_2, s_3; \\ s_3 &: h_1, h_3, h_2; & h_3 &: s_3, s_2, s_1. \end{aligned}$$

Suppose that everyone submits their true preferences to the NIMP trial-run algorithm. In the 1:1 stage, there are no matches. No one is ranked 1 by whom they rank first. At the 1:2 stage, it matches (s_2, h_2) and (s_3, h_1). And, eventually, it also matches (s_1, h_3). First, note that this matching is not stable since s_1 and h_2 form a blocking pair. Second, note that, if s_1 ranked h_2 as their top choice, above h_1, then (s_1, h_2) would be matched in the 1:1 stage. Therefore, s_1 has incentives to misreport their preferences. This shows that the NIMP trial-run algorithm is neither stable nor strategy-proof for students. Q.E.D.

Exercise 6.4. Evaluate: the NIMP trial-run algorithm is (i) Pareto efficient, and (ii) strategy-proof for hospitals.

Now, we turn to the NIMP algorithm, which was first used in 1952, and remained without any change until the redesign of the NRMP in the mid-1990s.

Algorithm 6.5 (NIMP)**.** *Students submit a rank ordering of hospitals, and indicate which of them are unacceptable. Likewise, hospitals submit a rank ordering of students and indicate which of them are unacceptable. First, from the ranking list of each hospital, remove every student who marked the hospital as unacceptable. Likewise, from the ranking list of each student, remove every hospital that marked the student as unacceptable. The edited lists are rank orderings of mutually acceptable parties. Initially, no one is tentatively matched. The algorithm proceeds in consecutive stages as follows.*

- *1:1 step. Check whether there are students and hospitals who rank each other as their top choice and are not tentatively matched. If no such matches are found, proceed to the 2:1 step. If any such matches are found, proceed to the tentative-assignment-and-update phase.*

  ...

- *k:1 step. Check whether there are students and hospitals such that the student ranks the hospital as their k-th choice, the hospital ranks the student as their top choice, and they are not tentatively matched. If no such matches are found, proceed to the k+1:1 step. If any such matches are found, proceed to the tentative-assignment-and-update phase.*

- *Tentative-assignment-and-update phase. Assume the algorithm entered this phase from the k:1 step. Assign tentatively the k:1 matches, i.e., tentatively match every student and hospital such that the hospital is k-th in the student's list and the student is the hospital's top choice. Any new tentative matches replace previous tentative matches. Then, update the rankings of the students and hospitals as follows.*

  - *Consider the ranking of student s_j. From this ranking, delete every hospital that student s_j ranks lower than their current tentative match. That is, if s_j is tentatively matched to their k-th choice, their ranking now only includes their first k choices.*
  - *Consider the ranking of hospital h_i. Delete student s_j from this ranking if hospital h_i was just deleted from the ranking of student s_j. That is, the updated ranking list of h_i only includes students who have not been tentatively assigned to a hospital they prefer over h_i. Note that if a top-ranked choice is deleted in a hospital's ranking list, students are moved up the ranking.*
  - *After updating the ranking lists, return to the 1:1 step.*

- *Terminate the algorithm when no new tentative matches can be found, at which point the current tentative matches become final.*

Example 6.6. *Consider the following example with four students and four hospitals. The preferences are given as follows (where we omit unacceptable parties from rank ordered lists).*

$$\begin{aligned} &s_1 : h_1, h_2, h_3, h_4 && h_1 : s_3, s_2, s_1 \\ &s_2 : h_2, h_3 && h_2 : s_1, s_2, s_3, s_4 \\ &s_3 : h_4, h_3 && h_3 : s_4, s_3, s_2 \\ &s_4 : h_2, h_1, h_4, h_3 && h_4 : s_2, s_1, s_4, s_3. \end{aligned}$$

We run the NIMP algorithm. In the first step, from each agent's ranking list delete every party who does not find them acceptable. This yields the following updated ranking lists:

$$\begin{aligned} &s_1 : h_1, h_2, h_4 && h_1 : s_1 \\ &s_2 : h_2, h_3 && h_2 : s_1, s_2, s_4 \\ &s_3 : h_4, h_3 && h_3 : s_4, s_3, s_2 \\ &s_4 : h_2, h_4, h_3 && h_4 : s_1, s_4, s_3. \end{aligned}$$

In the 1:1 step, match (s_1, h_1). Update the ranking lists as follows:

$$\begin{aligned} &s_1 : h_1 && h_1 : s_1 \\ &s_2 : h_2, h_3 && h_2 : s_2, s_4 \\ &s_3 : h_4, h_3 && h_3 : s_4, s_3, s_2 \\ &s_4 : h_2, h_4, h_3 && h_4 : s_4, s_3. \end{aligned}$$

In the 1:1 step, match (s_2, h_2). Update the ranking lists as follows:

$$\begin{aligned} &s_1 : h_1 && h_1 : s_1 \\ &s_2 : h_2 && h_2 : s_2, s_4 \\ &s_3 : h_4, h_3 && h_3 : s_4, s_3 \\ &s_4 : h_2, h_4, h_3 && h_4 : s_4, s_3. \end{aligned}$$

In the 1:1 step, we find no (new) matches. In the 2:1 step, match (s_4, h_4)*. Update the ranking lists as follows:*

$s_1 : h_1$ $\qquad$ $h_1 : s_1$

$s_2 : h_2$ $\qquad$ $h_2 : s_2$

$s_3 : h_4, h_3$ $\qquad$ $h_3 : s_3$

$s_4 : h_2, h_4$ $\qquad$ $h_4 : s_4, s_3.$

In the 1:1 step, we find no (new) matches. In the 2:1 step, match (s_3, h_3)*. Then, we find no new tentative matches, at which point the algorithm terminates. The resulting matching is given by:* (s_1, h_1)*,* (s_2, h_2)*,* (s_3, h_3)*, and* (s_4, h_4)*.*

Now, run the hospital-proposing Gale-Shapley algorithm. Recall the original preference profile:

$s_1 : h_1, h_2, h_3, h_4$ $\qquad$ $h_1 : s_3, s_2, s_1$

$s_2 : h_2, h_3$ $\qquad$ $h_2 : s_1, s_2, s_3, s_4$

$s_3 : h_4, h_3$ $\qquad$ $h_3 : s_4, s_3, s_2$

$s_4 : h_2, h_1, h_4, h_3$ $\qquad$ $h_3 : s_2, s_1, s_4, s_3.$

Round 1: Proposals: $h_1 \to s_3$*,* $h_2 \to s_1$*,* $h_3 \to s_4$*,* $h_4 \to s_2$

Accepted: $h_2 \to s_1$*,* $h_3 \to s_4$

Round 2: Proposals: $h_1 \to s_2$*,* $h_4 \to s_1$

Accepted: $h_2 \to s_1$*,* $h_3 \to s_4$

Round 3: Proposals: $h_1 \to s_1$*,* $h_4 \to s_4$

Accepted: $h_1 \to s_1$*,* $h_4 \to s_4$

Round 4: Proposals: $h_2 \to s_2$*,* $h_3 \to s_3$

Accepted: $h_1 \to s_1$*,* $h_2 \to s_2$*,* $h_3 \to s_3$*,* $h_4 \to s_4$

Both the NIMP algorithm and the hospital-proposing Gale-Shapley algorithm reach the same matching. As we know from the previous section, this matching is the hospital-optimal stable matching. As the next theorem shows, the NIMP algorithm and the hospital-proposing Gale-Shapley algorithm do not coincide by chance. They are, indeed, equivalent algorithms.

Theorem 6.7. *The NIMP algorithm and the hospital-proposing Gale-Shapley algorithm are equivalent* (*they always generate the same matching*).

Exercise 6.8. Prove Theorem 6.7.

Exercise 6.9. The first step of the NIMP algorithm consists in deleting from each agent's rank order list the parties on the other side who find such agent unacceptable. Show with an example that we may obtain a matching that is not stable if we skip the first step in the NIMP algorithm.

Notes

The history of the NRMP, including the NIMP algorithm and its 1951 trial-run, is surveyed in Roth (1984) and Chapter 1 of Roth and Sotomayor (1990). Roth and Xing (1994) provide a comprehensive treatment of unraveling in several real-life markets. For a full account of the redesign of the NRMP in the 1990s, see Roth and Peranson (1997; 1999).

7 School choice

In this section we study the problem of assigning public school seats to K-12 students. Traditionally, children are assigned to schools according to where they live. However, in several parts of the world this has been deemed unfair in recent years. While wealthier parents can decide to move to a city or neighborhood with good schools, parents without such means had no choice of school, and had to send their children to schools assigned to them by the district. Today, several states in the U.S. offer inter-district and intra-district school choice programs. We now study the problem of assigning students to schools as a formal two-sided matching problem.

Definition 7.1. *A **school choice problem** is a tuple $(I, S, Q, \succ_I, \succ_S)$, where I and S are nonempty, finite, and disjoint sets of **students** and **schools**, respectively; $Q = (q_s)_{s \in S}$ is a vector of **capacities**, one for each school, with $q_s \in \mathbb{N}$ for each $s \in S$; $\succ_I = (\succ_i)_{i \in I}$ is a **preference profile** for students, where $\succ_i \in \mathcal{P}(S \cup \emptyset)$ for each $i \in I$, and $\succ_S = (\succ_s)_{s \in S}$ is a **priority profile** for schools, where $\succ_s \in \mathcal{P}(I)$ for each $s \in S$.*

The capacity q_s indicates the number of available seats at school s. Priorities have the following interpretation: the ranking $i \succ_s j$ means that student i has higher priority than student j in school s. In practice, priorities are determined by combining test scores, home address, whether there are siblings already enrolled in the school, etc. In some cases they might even have a random component. You should think about priorities as part of a school's admission procedure. In principle, $\succ_s$ only incorporates information that is publicly available. Students' preferences describe their ranking over schools. The option $\emptyset$ corresponds to being unmatched. For example, the ranking $s \succ_i \emptyset \succ_i s'$ means that student i prefers school s to being unmatched, but prefers being unmatched than being assigned to school s'.

Definition 7.2. *Let $(I, S, Q, \succ_I, \succ_S)$ be a school choice problem. A **matching** is a function $\mu : I \to S \cup \{\emptyset\}$ such that $|\mu^{-1}(s)| \leq q_s$ for every $s \in S$. Denote the set of all matchings by $\mathcal{M}(I, S, Q)$.*

In this context, a matching indicates the school to which each student is assigned. The inverse mapping μ^{-1} indicates the set of students that are assigned to each school. The requirement $|\mu^{-1}(s)| \leq q_s$ means that the number of students matched to a school under μ must be less than or equal to its capacity.

We consider two key properties of matchings in school choice problems.

Definition 7.3. *A matching $\mu \in \mathcal{M}(I, S, Q)$* ***eliminates justified envy*** *if there is no pair $(i, s) \in I \times S$ such that $i \succ_s j$ and $s \succ_i \mu(i)$ for some student j with $\mu(j) = s$.*

Notice the resemblance of justified envy with stability. In this context, justified envy refers to envy between students. Student i has justified envy for j if i has a higher priority over j in the school to which j is matched to, while i is matched to a school they prefer less. Underlying this notion is the idea that if the priority of student i for school s is violated in favor of a different student j, then student i has incentives to seek legal action against the school. In the context of school choice, legal and political concerns appear to strongly favor mechanisms that avoid such situations.

Definition 7.4. *A matching $\mu \in \mathcal{M}(I, S, Q)$ is* ***non-wasteful*** *if, for every $(i, s) \in I \times S$, $s \succ_i \mu(i)$ implies $|\mu^{-1}(s)| = q_s$.*

Non-wastefulness simply means that there are no seats vacant while there are students who prefer those seats to their current assignments.

Exercise 7.5. How are students assigned to high schools in the place where you went to high school? Does the mechanism create matchings that eliminate justified envy and are non-wasteful?

7.1 The Boston mechanism

Until recently, one of the most commonly used school choice mechanisms was the one used by the Boston Public Schools (BPS) in Massachusetts. The Boston school choice mechanism is defined as follows.

Algorithm 7.6 (Boston)**.** *Each school determines a priority ordering over students. (In the case of Boston, priorities depend on home address, whether the student has a sibling already attending a school, and a lottery number to break ties.) Each student submits a preference ranking of the schools. The algorithm proceeds in steps as follows.*

- ***Step 1:*** *For each school, consider the students who have listed it as their top choice. Assign the seats of each school to these students one at a time following its priority order. Proceed until either there are no seats left or until there is no student left who has listed it as their top choice.*

- ***Step k:*** *For each school that still has available seats, consider the students who have listed it as their k-th choice. Assign the remaining seats of each school to these students one at a time following its priority order. Proceed until either there are no seats left or until there is no student left who has listed it as their top choice.*
- *Stop if all students have been assigned or there are no seats left.*

One major problem with the mechanism is that is not strategy-proof. Even if a student has a very high priority at school s, unless they list it as their top choice, they will lose their priority to students who ranked school s at the top of their list. For this reason, the Boston mechanism gives parents strong incentives to inflate their ranking of schools where they have high priority. As observed by Glazerman and Meyer (1994),

> *It may be optimal for some families to be strategic in listing their school choices. For example, if a parent thinks that their favorite school is oversubscribed and they have a close second favorite, they may try to avoid "wasting" their first choice on a very popular school and instead list their number two school first.*

This incentive for families to strategize their applications makes the mechanism difficult to predict. It also favors families who are more experienced with the mechanism.

Exercise 7.7. Give an example in which at least one student has incentives to misreport their true preferences to the Boston mechanism.

The Boston algorithm is also known as the Immediate Acceptance (IA) algorithm. This is to contrast it with the Deferred Acceptance (DA) algorithm by Gale and Shapley. Note that the Boston algorithm works as a student-proposing Gale-Shapley algorithm in which acceptances are *final* instead of tentative.

Exercise 7.8. Evaluate: the Boston mechanism (i) eliminates justified envy, and (ii) is non-wasteful.

Exercise 7.9. Show that in the Boston mechanism, if student i is matched to a school they rank worse than school s, then all the seats of school s are taken by students who rank s at least as high as i.

7.2 Deferred acceptance and Pareto efficiency

A first alternative to the Boston mechanism is the students-proposing deferred acceptance algorithm. This mechanism works almost verbatim as described for the marriage market, but with one change: at every step, a school s is tentatively matched to the best q_s students who have proposed or who were tentatively matched to it in the previous round, and rejects the rest.

The deferred acceptance mechanism eliminates justified envy. This follows from the same argument we used to show that it always produces a stable matching in marriage markets. And, it is also strategy-proof for the students. No student can gain by misreporting their preferences. Due to these desirable features, DA was adopted by the school choice programs of New York City (in 2003) and Boston (in 2005). However, a disadvantage of DA is that it does not guarantee that the resulting allocation is Pareto efficient *for the students*. Consider the following example.

Example 7.10. *There are 3 students* $\{1, 2, 3\}$ *and 3 schools* $\{A, B, C\}$*, each of which has only one seat. Priorities and preferences are as follows:*

$$\succ_1\colon B, A, C \qquad \succ_2\colon A, B, C \qquad \succ_3\colon A, B, C$$
$$\succ_A\colon 1, 3, 2 \qquad \succ_B\colon 2, 1, 3 \qquad \succ_C\colon 2, 1, 3$$

Verify that the student-proposing DA matches 1 *to* A*,* 2 *to* B*, and* 3 *to* C*. Since students 1 and 2 prefer schools* A *and* B *more than* C*, and schools* A *and* B *prefer student 1 or 2 over 3, student 3 is matched to school* C *in every stable matching.*[12] *However, stability forces students 1 and 2 to "share" schools* A *and* B *in an inefficient way. It forces* 1 *to be matched to* A *rather than to* B*. The allocation is Pareto dominated by the matching where we switch the assignments of* 1 *and* 2*, which makes all students better off.*

Allocations that are not Pareto efficient for students are hard to justify in practice. In this context, the public or the designer might wish to prioritize efficiency for one side of the market over the other.

Exercise 7.11. Evaluate: the Boston mechanism is Pareto efficient for students.

[12] An alternative way of observing this is by noting that C is the most preferred stable partner of 3.

7.3 Two notions of Pareto efficiency

The example above shows that DA can result in allocations that are not Pareto efficient for the students. Now we investigate whether there is a mechanism that does better than DA in this sense.

Definition 7.12. *A mechanism ϕ **Pareto dominates** a mechanism ψ if (i) for all profile of preferences, ϕ results in a matching that all students prefer at least as much as the matching obtained by ψ, and (ii) for some profile of students' preferences, some of the students are strictly better off under ϕ than ψ.*

Even if a mechanism can result in allocations that are not Pareto-efficient, it is not clear that we can find another mechanism that Pareto dominates it *and* has desirable properties. Indeed, the next result shows that for any mechanism ϕ that is strategy-proof for students and non-wasteful, there is no other strategy-proof mechanism that Pareto dominates it. In particular, the result applies to the student-proposing DA algorithm, being both strategy-proof and non-wasteful.

Proposition 7.13. *If ϕ is a strategy-proof (for students) and non-wasteful mechanism, then there is no strategy-proof mechanism that Pareto dominates ϕ.*

Proof. Let ϕ be a strategy-proof (for students) and non-wasteful mechanism. The proof is divided into two parts. Fix a preference profile $\succ_I \in \mathcal{P}(S \cup \{\emptyset\})^{|I|}$. Let $\mu = \phi(\succ_I)$ be the matching produced by the mechanism ϕ given preferences $\succ_I$, and suppose ν is matching that satisfies $\nu(i) \succcurlyeq_i \mu(i)$ for all i.

First, we will show that the same set of agents is matched under the two matchings. Clearly, if i is matched under μ then i must also be matched under ν (otherwise, i would find it profitable to report $s = \mu(i)$ as unacceptable in ϕ). Conversely, suppose i is matched under ν but not under μ. Because ϕ is non-wasteful, it must be that under ν, i is matched to some school s that was fully assigned under μ. So, it must be that under ν some other student i_1 is now matched to some other school s_1. Since preferences are strict, i_1 is made strictly better off. This implies that s_1 was full under μ. Hence some other student i_2 is no longer matched to i_1 but to another school that was full under μ. Proceeding this way we obtain a sequence of students who are made strictly better off. Since this sequence must stop, we must find at least one student j who is matched to a school that under μ was not at full capacity.

Because ϕ is non-wasteful and preferences are strict then j is made strictly worse off. A contradiction.

Now suppose there exists a mechanism ψ that Pareto dominates ϕ. We prove that ψ is *not* strategy-proof for students. There exists a profile $\succ_I$ of students' preferences such that $\psi[\succ_I](i) \succcurlyeq_i \phi[\succ_I](i)$ for all i and the preference is strict for some j. Let $s = \psi[\succ_I](j)$. Consider that j reports $\succ'_j$, the preference ranking in which s is the only acceptable school. Then, strategy-proofness of ϕ implies $\phi(\succ'_j, \succ_{-j})(j) = \emptyset$ (otherwise, j would have incentives to misreport in ϕ). Since ϕ is dominated by ψ, by what we showed above, the same students must be matched under both $\phi(\succ'_j, \succ_{-j})$ and $\psi(\succ'_j, \succ_{-j})$. Then, $\psi(\succ'_j, \succ_{-j})(j) = \emptyset$. However, this implies that ψ is not strategy-proof since, when j's true preference is $\succ'_j$, they would rather lie and report $\succ_j$. Q.E.D.

7.4 The school choice TTC

An alternative to obtain matchings that are Pareto efficient for students is to use a modified version off the Top-trading Cycle (TTC).

Algorithm 7.14 (School choice TTC)**.** *Assign a counter for each school which keeps track of how many seats are still available at the school. Initially, set the counter of each school to be equal to its capacity. Proceed in steps as follows.*

- *Each student points to their favorite school. Each school that has not run out of counters points to the student who has the highest priority for the school.*

  Since the number of students and schools are finite, there is at least one cycle. Moreover, every school and every student can be part of at most one cycle.

  Every student in a cycle is assigned a seat at the school they point to and is removed. The counter of each school in a cycle is reduced by one. Remove schools whose counter reaches zero.

- *Repeat until no more students are assigned.*

The mechanism favors students with high priority, but in a novel way. A student is allowed to use their high priority in one school to get into another school, as long as this leads to a Pareto improvement.

Proposition 7.15. *The school choice TTC mechanism is strategy-proof for students and Pareto efficient.*

Exercise 7.16. Prove Proposition 7.15

Exercise 7.17. Prove that the school choice TTC is non-wasteful.

A drawback from TTC is that it does not eliminate justified envy. Consider the following example.

Example 7.18. *Consider the same school choice problem as in Example 7.10, which we reproduce below for convenience:*

$$\succ_1: B, A, C \quad \succ_2: A, B, C \quad \succ_3: A, B, C$$
$$\succ_A: 1, 3, 2 \quad \succ_B: 2, 1, 3 \quad \succ_C: 2, 1, 3$$

As we argued, the student-proposing DA matches 1 to A, 2 to B, and 3 to C. Note that the TTC, in turn, assigns 1 to B, 2 to A, and 3 to C. As we noted, this matching does not eliminate justified envy since 3 envies 2. Even though the outcome of the TTC Pareto dominates the one of the student-proposing DA, according to Proposition 7.13, this cannot always be the case (since TTC is strategy-proof and non-wasteful). Indeed, consider the following preference profile:

$$\succ_1: B, C, A \quad \succ_2: A, B, C \quad \succ_3: A, B, C$$
$$\succ_A: 1, 3, 2 \quad \succ_B: 2, 3, 1 \quad \succ_C: 2, 1, 3$$

Even though agent 1 ranks A as the worst choice now, they still have the highest priority in A (which matters for TTC). Verify that the student-proposing DA matches 1 to C, 2 to B, and 3 to A. By contrast, TTC matches 1 to B, 2 to A, and 3 to C. Therefore, agents 1 and 2 would rather use the TTC to assign school seats, while student 3 would rather use the student-proposing DA. In this case, no matching Pareto dominates the other.

The example above shows how there is a trade-off between eliminating justified envy and Pareto efficiency for students. The student-proposing DA eliminates justified envy but may generate matchings that are *not* Pareto efficient amongst students. Conversely, the school choice TTC always generates matchings that are Pareto efficient amongst students, but may fail to eliminate justified envy.

As argued in Kesten (2010), the idea of trading priorities is not without problems from a practical perspective. In his May 25, 2005, memorandum to the School Committee, regarding his take on a trading-based mechanism, the (then) Superintendent of Boston, Thomas Payzant, writes (p. 3):

> *There may be advantages to this approach... It may be argued, however, that certain priorities, e.g., sibling priority, apply only to students for particular schools and should not be traded away.*

The Boston Public Schools (BPS) Strategic Planning Team's May 11, 2005, Recommendation Report further states (pp. 23, 38):

> *The trading mechanism can have the effect of "diluting" priorities' impacts, if priorities are to be "owned" by the district as opposed to being "owned" by parents; it shifts the emphasis onto the priorities and away from the goals the BPS is trying to achieve by granting these priorities in the first place; and could lead to families believing they can strategize by listing a school they don't want in hopes of a trade.*

Therefore, when designing school choice systems, administrators have been forced to take a stance and decide whether they favor a mechanism that eliminates justified envy, such as DA, or that is Pareto efficient for students, such as TTC.

Additional exercises

Exercise 7.19. Consider a school choice problem in which all students have the same ranking $\succ_i = \succ^*$ over schools, and that this ranking is known. In this case, what mechanism would you suggest for assigning students to schools? Why?

Exercise 7.20. In some countries, such as India, China, and Turkey, and some schools in the United States, students take a centralized exam that determines common school priorities over students. Consider the school choice problem again, but assume that all schools have the same priority $\succ_s = \succ^*$ over students, and that this ranking is known. In this case, what mechanism would you suggest? Why?

Notes

The seminal contribution in the school choice literature is Abdulkadiroğlu and Sönmez (2003). For a discussion of the tension between stability and efficiency, including the shortcomings of using TTC in practice, see Kesten (2010). The literature on school choice has exploded since the original paper by Abdulkadiroğlu and Sönmez. It includes theoretical and applied papers using a wide range of methods: experiments, observational studies, structural estimation, etc. For an overview of the literature, see Pathak (2011). For a discussion on the practical aspects of designing school choice mechanisms, see Pathak (2017).

References

Abdulkadiroğlu, Atila, and Tayfun Sönmez. 1998. "Random Serial Dictatorship and the Core from Random Endowments in House Allocation Problems." *Econometrica* 66 (3): 689–701. (Cited on pages 43, 47).

———. 1999. "House Allocation with Existing Tenants." *Journal of Economic Theory* 88 (2): 233–260. (Cited on pages 27, 28).

———. 2003. "School Choice: A Mechanism Design Approach." *American Economic Review* 93 (3): 729–747. (Cited on page 87).

Adachi, Hiroyuki. 2000. "On a characterization of stable matchings." *Economics Letters* 68 (1): 43–49. (Cited on page 68).

Bogomolnaia, Anna, and Hervé Moulin. 2001. "A New Solution to the Random Assignment Problem." *Journal of Economic Theory* 100 (2): 295–328. (Cited on pages 44, 45, 47).

de la Fuente, Angel. 2000. *Mathematical Methods and Models for Economists.* Cambridge University Press. (Cited on page 68).

Diamantaras, D., E. Cardamone, K.A.C. Campbell, S. Deacle, and L.A.A. Delgado. 2009. *A Toolbox for Eoreonomic Design.* Palgrave Macmillan. (Cited on page 14).

Gale, David, and Lloyd S. Shapley. 1962. "College Admissions and the Stability of Marriage." *The American Mathematical Monthly* 69 (1): 9–15. (Cited on pages 48, 50, 52, 67, 72).

Galichon, Alfred. 2016. *Optimal Transport Methods in Economics.* Princeton University Press. (Cited on page 68).

Glazerman, Steven, and Robert H. Meyer. 1994. "Public School Choice in Minneapolis." In *Midwest approaches to school reform,* Proceedings of a conference held at the Federal Reserve Bank of Chicago, edited by T.A. Downes and W. A. Testa, 110–26. (Cited on page 81).

Haeringer, Guillaume. 2017. *Market Design: Auctions and Matching.* The MIT Press. (Cited on pages 4, 14, 47).

Kesten, Onur. 2010. "School Choice with Consent." *The Quarterly Journal of Economics* 125 (3): 1297–1348. (Cited on pages 86, 87).

Knuth, Donald E. 1976. *Marriages Stables.* Montreal: Les Presses de l'Université de Montreal. (Cited on pages 54, 67).

Kreps, David M. 1988. *Notes On The Theory Of Choice.* Underground classics in economics. Avalon Publishing. (Cited on page 14).

Ma, Jinpeng. 1994. "Strategy-proofness and the strict core in a market with indivisibilities." *International Journal of Game Theory* 23:75–83. (Cited on pages 23, 28).

Mas-Colell, Andreu, Michael D. Whinston, and Jerry R. Green. 1995. *Microeconomic Theory.* Oxford University Press. (Cited on pages 14, 47).

Pathak, Parag A. 2011. "The Mechanism Design Approach to Student Assignment." *Annual Review of Economics* 3 (1): 513–536. (Cited on page 87).

———. 2017. "What Really Matters in Designing School Choice Mechanisms." In *Advances in Economics and Econometrics: Eleventh World Congress,* edited by Bo Honoré, Ariel Pakes, Monika Piazzesi, and Larry Samuelson, 1:176–214. Econometric Society Monographs. Cambridge University Press. (Cited on page 87).

Roth, Alvin E. 1982. "Incentive compatibility in a market with indivisible goods." *Economics Letters* 9 (2): 127–132. (Cited on pages 21, 28).

———. 1984. "The Evolution of the Labor Market for Medical Interns and Residents: A Case Study in Game Theory." *Journal of Political Economy* 92 (6): 991–1016. (Cited on pages 72, 78).

———. 2002. "The Economist as Engineer: Game Theory, Experimentation, and Computation as Tools for Design Economics." *Econometrica* 70 (4): 1341–1378. (Cited on page 70).

Roth, Alvin E., and Elliott Peranson. 1997. "The Effects of the Change in the NRMP Matching Algorithm." *JAMA* 278 (9): 729–732. (Cited on page 78).

———. 1999. "The Redesign of the Matching Market for American Physicians: Some Engineering Aspects of Economic Design." *American Economic Review* 89 (4): 748–780. (Cited on page 78).

Roth, Alvin E., and Andrew Postlewaite. 1977. "Weak versus strong domination in a market with indivisible goods." *Journal of Mathematical Economics* 4 (2): 131–137. (Cited on pages 20, 27).

Roth, Alvin E., Tayfun Sönmez, and M. Utku Ünver. 2004. "Kidney Exchange." *The Quarterly Journal of Economics* 119 (2): 457–488. (Cited on pages 31, 36).

———. 2005. "Pairwise kidney exchange." *Journal of Economic Theory* 125 (2): 151–188. (Cited on page 36).

Roth, Alvin E., and Marilda A. Oliveira Sotomayor. 1990. *Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis.* Econometric Society Monographs. Cambridge University Press. (Cited on pages 4, 62, 67, 68, 78).

Roth, Alvin E., and John H. Vande Vate. 1990. "Random Paths to Stability in Two-Sided Matching." *Econometrica* 58 (6): 1475–1480. (Cited on page 51).

Roth, Alvin E., and Xiaolin Xing. 1994. "Jumping the Gun: Imperfections and Institutions Related to the Timing of Market Transactions." *The American Economic Review* 84 (4): 992–1044. (Cited on page 78).

Shapley, Lloyd, and Herbert Scarf. 1974. "On cores and indivisibility." *Journal of Mathematical Economics* 1 (1): 23–37. (Cited on pages 20, 27).

Shapley, Lloyd S., and Martin Shubik. 1971. "The assignment game I: The core." *International Journal of Game Theory* 1 (1): 111–130. (Cited on pages 62, 64, 68).

Sundaram, Rangarajan K. 1996. *A First Course in Optimization Theory.* Cambridge University Press. (Cited on page 68).

Topkis, Donald M. 1998. *Supermodularity and Complementarity.* Princeton University Press. (Cited on page 68).

Vohra, Rakesh V. 2005. *Advanced Mathematical Economics.* Routledge Advanced Texts in Economics and Finance. London: Routledge. (Cited on page 68).