CHAPTER ONE

Two-Sided Markets: Stable Matching

Federico Echenique, Nicole Immorlica, and Vijay V. Vazirani

1.1 Introduction

The field of matching markets was initiated by the seminal work of Gale and Shapley on stable matching. Stable matchings have remarkably deep and pristine structural properties, which have led to polynomial time algorithms for numerous computational problems as well as quintessential game-theoretic properties. In turn, these have opened up the use of stable matching to a host of important applications.

This chapter will deal with the following four aspects:

1. Gale and Shapley's deferred acceptance algorithm for computing a stable matching; we will sometimes refer to it as the DA algorithm;
2. the incentive compatibility properties of this algorithm;
3. the fact that the set of all stable matchings of an instance forms a finite, distributive lattice, and the rich collection of structural properties associated with this fact;
4. the linear programing approach to computing stable matchings.

A general setting. A setting of the stable matching problem which is particularly useful in applications is the following (this definition is quite complicated because of its generality, and can be skipped on the first reading).

Definition 1.1. Let W be a set of n workers and F a set of m firms. Let c be a *capacity function* $c\colon F \to \mathbb{Z}_+$ giving the maximum number of workers that can be matched to a firm; each worker can be matched to at most one firm. Also, let $G = (W, F, E)$ be a bipartite graph on vertex sets W, F and edge set E. For a vertex v in G, let $N(v)$ denote the set of its neighbors in G. Each worker w provides a strict preference list $l(w)$ over the set $N(w)$ and each firm f provides a strict preference list $l(f)$ over the set $N(f)$. We will adopt the convention that each worker and firm prefers being matched to one of its neighbors to remaining unmatched, and it prefers remaining unmatched to being matched to a non-neighbor.[1] If a worker or firm remains unmatched, we will say that it is matched to $\perp$.

[1] An alternative way of defining preference lists, which we will use in Section 1.3.2 is the following. Each worker w has a preference list over $F \cup \{\perp\}$, with firms in $N(w)$ listed in the preference order of w, followed by $\perp$, followed by $(F \setminus N(w))$ listed in arbitrary order. Similarly, each firm f's preference list is over $W \cup \{\perp\}$.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

We wish to study all four aspects stated for this setting. However, it would be quite unwise and needlessly cumbersome to study the aspects directly in this setting. It turns out that the stable matching problem offers a natural progression of settings, hence allowing us to study the aspects gradually in increasing generality.

1. **Setting I.** Under this setting $n = m$, the capacity of each firm is one and graph G is a complete bipartite graph. Thus in this setting each side, consisting of workers or firms, has a total order over the other side. This simple setting will be used for introducing the core ideas.
2. **Setting II.** Under this setting n and m are not required to be equal and G is arbitrary; however, the capacity of each firm is still one. The definition of stability becomes more elaborate, hence making all four aspects more difficult in this setting. Relying on the foundation laid in Setting I, we will present only the additional ideas needed.
3. **Setting III.** This is the general setting defined in Definition 1.1. We will give a reduction from this setting to Setting II, so that the algorithm and its consequences carry over without additional work.

1.2 The Gale–Shapley Deferred Acceptance Algorithm

In this section we will define the notion of a stable matching for all three settings and give an efficient algorithm for finding it.

1.2.1 The DA Algorithm for Setting I

In this setting, the number of workers and firms is equal, i.e., $n = m$, and each firm has unit capacity. Furthermore, each worker and each firm has a total order over the other side.

Notation. If *worker w prefers firm f to f'* then we represent this as $f \succ_w f'$; a similar notation is used for describing the preferences of a firm.

We next recall a key definition from graph theory. Let $G = (W, F, E)$ be a graph with equal numbers of workers and firms, i.e., $|W| = |F|$. Then, $\mu \subseteq E$ is a *perfect matching* in G if each vertex of G has exactly one edge of μ incident at it. If so, μ can also be viewed as a bijection between W to F. If $(w,f) \in \mu$ then we will say that μ *matches w to f* and use the notation $\mu(w) = f$ and $\mu(f) = w$.

Definition 1.2. Worker w and firm f form a *blocking pair* with respect to a perfect matching μ, if they prefer each other over their partners in μ, i.e., $w \succ f \mu(f)$ and $f \succ_w \mu(w)$.

If (w,f) form a blocking pair with respect to perfect matching μ then they have an incentive to secede from matching μ and pair up by themselves. The significance of the notion of stable matching, defined next, is that no worker–firm pair has an incentive to secede from this matching. Hence such matchings lie in the *core* of the particular instance; this key notion will be introduced in Chapter 3. For now, recall from cooperative game theory that the core consists of solutions under which no subset of the agents can gain more (i.e., with each agent gaining at least as much

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

and at least one agent gaining strictly more) by seceding from the grand coalition. Additionally, in Chapter 3 we will also establish that stable matchings are efficient and individually rational.

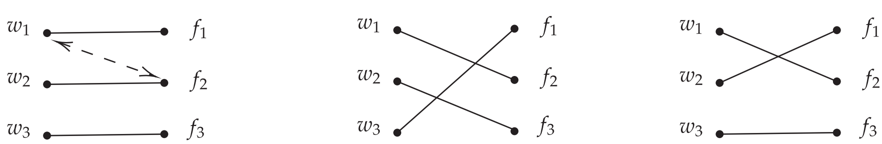
Definition 1.3. A perfect matching μ with no blocking pairs is called a *stable matching*.

It turns out that every instance of the stable matching problem with complete preference lists has at least one stable matching. Interestingly enough, this fact follows as a corollary of the deferred acceptance algorithm, which finds in polynomial time one stable matching among the $n!$ possible perfect matchings in G.

Example 1.4. Let I be an instance of the stable matching problem with three workers and three firms and the following preference lists:

$$\begin{array}{ll} w_1: f_2, f_1, f_3 & f_1: w_1, w_2, w_3 \\ w_2: f_2, f_3, f_1 & f_2: w_1, w_2, w_3 \\ w_3: f_1, f_2, f_3 & f_3: w_1, w_3, w_2 \end{array}$$

Figure 1.1 shows three perfect matchings in instance I. The first matching is unstable, with blocking pair (w_1, f_2), and the last two are stable (this statement is worth verifying).



(a) An unstable perfect matching (b) Stable matching 1 (c) Stable matching 2

Figure 1.1

We next present the deferred acceptance algorithm[2] for Setting I, described in Algorithm 1.8. The algorithm operates iteratively, with one side proposing and the other side acting on the proposals received. We will assume that workers propose to firms. The initialization involves each worker marking each firm in its preference list as *uncrossed*.

Each iteration consists of three steps. First, each worker proposes to the best uncrossed firm on its list. Second, each firm that got proposals tentatively accepts the best proposal it received and rejects all other proposals. Third, each worker who was rejected by a firm crosses that firm off its list. If in an iteration each firm receives a proposal, we have a perfect matching, say μ, and the algorithm terminates.

The following observations lead to a proof of correctness and running time.

Observation 1.5. *If a firm gets a proposal in a certain iteration, it will keep getting at least one proposal in all subsequent iterations.*

[2] The reason for this name is provided in Remark 1.11.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

Observation 1.6. *As the iterations proceed, for each firm the following holds: once it receives a proposal, it tentatively accepts a proposal from the same or a better worker, according to its preference list.*

Lemma 1.7. *Algorithm 1.8 terminates in at most n^2 iterations.*

Proof In every iteration other than the last, at least one worker will cross a firm off its preference list. Consider iteration number $n^2 - n + 1$, assuming the algorithm has not terminated so far. Since the total size of the n preference lists is n^2, there is a worker, say w, who will propose to the last firm on its list in this iteration. Therefore by this iteration w has proposed to every firm and every firm has received a proposal. Hence, by Observation 1.5, in this iteration every firm will get a proposal and the algorithm will terminate with a perfect matching. □

Algorithm 1.8. Deferred acceptance algorithm

Until all firms receive a proposal, do:

1. $\forall w \in W$: w proposes to its best uncrossed firm.
2. $\forall f \in F$: f tentatively accepts its best proposal and rejects the rest.
3. $\forall w \in W$: If w got rejected by firm f, it crosses f off its list.

Output the perfect matching, and call it μ.

Example 1.9. The Figures 1.2 shows the two iterations executed by Algorithm 1.8 on the instance given in Example 1.4. In the first iteration, w_2 will get rejected by f_2 and will cross it from its list. In the second iteration, w_2 will propose to f_3, resulting in a perfect matching.

Figure 1.2

Theorem 1.10. *The perfect matching found by the DA algorithm is stable.*

Proof For the sake of contradiction assume that μ is not stable and let (w, f') be a blocking pair. Assume that $\mu(w) = f$ and $\mu(f') = w'$ as shown in Figure 1.3. Since (w, f') is a blocking pair, w prefers f' to f and therefore must have proposed to f' and been rejected in some iteration, say i, before eventually proposing to f. In iteration i, f' must have tentatively accepted the proposal from a worker it likes better than w. Therefore, by Observation 1.6, at the

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

termination of the algorithm, $w' \succ_{f'} w$. This contradicts the assumption that (w, f') is a blocking pair. □

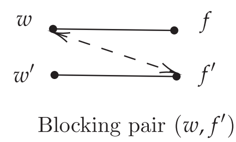


Figure 1.3 Blocking pair (w, f').

Remark 1.11. The Gale–Shapley algorithm is called the *deferred acceptance algorithm* because firms do not immediately accept proposals received by them – they defer them and accept only at the end of the algorithm when a perfect matching is found. In contrast, under the immediate acceptance algorithm, each firm immediately accepts the best proposal it has received; see Chapter 3.

Our next goal is to prove that the DA algorithm, with workers proposing, leads to a matching that is favorable for workers and unfavorable for firms. We first formalize the terms "favorable" and "unfavorable."

Definition 1.12. Let S be the set of all stable matchings over (W, F). For each worker w, the *realm of possibilities* $R(w)$ is the set of all firms to which w is matched in S, i.e., $R(w) = \{f \mid \exists \mu \in S \text{ s.t. } (w, f) \in \mu\}$. The *optimal firm* for w is the best firm in $R(w)$ with respect to w's preference list; it will be denoted by optimal(w). The *pessimal firm* for w is the worst firm in $R(w)$ with respect to w's preference list and will be denoted by pessimal(w). The definitions of these terms for firms are analogous.

Lemma 1.13. *Two workers cannot have the same optimal firm, i.e., each worker has a unique optimal firm.*

Proof Suppose that this is not the case and suppose that two workers w and w' have the same optimal firm, f. Assume without loss of generality that f prefers w' to w. Let μ be a stable matching such that $(w, f) \in \mu$ and let f' be the firm matched to w' in μ. Since $f =$ optimal(w') and w' is matched to f' in a stable matching, it must be the case that $f \succ_{w'} f'$. Then (w', f) forms a blocking pair with respect to μ, leading to a contradiction. See Figure 1.4. □

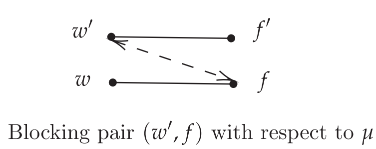


Figure 1.4 Blocking pair (w', f) with respect to μ.

Corollary 1.14. *Matching each worker to its optimal firm results in a perfect matching, say μ_W.*

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

Lemma 1.15. *The matching μ_W is stable.*

Proof Suppose that this is not the case and let (w,f') be a blocking pair with respect to μ_W, where $(w,f),(w',f') \in \mu_W$. Then $f' \succ_w f$ and $w \succ_{f'} w'$.

Since optimal$(w') = f'$, there is a stable matching, say μ', s.t. $(w',f') \in \mu'$. Assume that w is matched to firm f'' in μ'. Now since optimal$(w) = f$, $f \succ_w f''$. This together with $f' \succ_w f$ gives $f' \succ_w f''$. Then (w,f') is a blocking pair with respect to μ', giving a contradiction. See Figure 1.5. $\square$

(a) Blocking pair (w, f') with respect to μ_W (b) Blocking pair (w, f') with respect to μ'

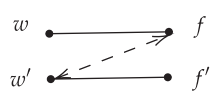
Figure 1.5

Proofs similar to those of Lemmas 1.13 and 1.15 show that each worker has a unique pessimal firm and the perfect matching that matches each worker to its pessimal firm is also stable.

Definition 1.16. The perfect matching that matches each worker to its optimal (pessimal) firm is called the *worker-optimal (-pessimal) stable matching*. The notions of *firm-optimal (-pessimal) stable matching* are analogous. The worker and firm optimal stable matchings will be denoted by μ_W and μ_F, respectively.

Theorem 1.17. *The worker-proposing DA algorithm finds the worker-optimal stable matching.*

Proof Suppose that this is not the case; then there must be a worker who is rejected by its optimal firm before proposing to a firm it prefers less. Consider the first iteration in which a worker, say w, is rejected by its optimal firm, say f. Let w' be the worker that firm f tentatively accepts in this iteration; clearly, $w' \succ_f w$. By Lemma 1.13, optimal$(w') \neq f$ and, by the assumption made in the first sentence of this proof, w' has not yet been rejected by its optimal firm (and perhaps never will be). Therefore, w' has not yet proposed to its optimal firm; let the latter be f'. Since w' has already proposed to f, we have that $f \succ_{w'} f'$. Now consider the worker-optimal stable matching μ; clearly, (w,f), $(w',f') \in \mu$. Then (w',f) is a blocking pair with respect to μ, giving a contradiction. See Figure 1.6. $\square$



Blocking pair (w', f) with respect to μ

Figure 1.6 Blocking pair (w', f) with respect to μ.

Lemma 1.18. *The worker-optimal stable matching is also firm pessimal.*

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

Proof Let μ be the worker-optimal stable matching and suppose that it is not firm pessimal. Let μ' be a firm-pessimal stable matching. Now, for some $(w,f) \in \mu$, pessimal$(f) \neq w$. Let pessimal$(f) = w'$; clearly, $w \succ_f w'$. Let $w =$ pessimal(f'); then $(w,f'), (w',f) \in \mu'$. Since optimal$(w) = f$ and w is matched to f' in a stable matching, $f \succ_w f'$. Then (w,f) forms a blocking pair with respect to μ', giving a contradiction. □

1.2.2 Extension to Setting II

Recall that in this setting each worker and firm has a total preference order over only its neighbors in the graph $G = (W, F, E)$ and $\perp$, with $\perp$ the least preferred element in each list; matching a worker or firm to $\perp$ is equivalent to leaving it unmatched.

In this setting, a stable matching may not be a perfect matching in G even if the number of workers and firms is equal; however, it will be a maximal matching. Recall that a matching $\mu \subseteq E$ is *maximal* if it cannot be extended with an edge from $E - \mu$. As a result of these changes, in going from Setting I to Setting II, the definition of stability also needs to be enhanced.

Definition 1.19. Let μ be any maximal matching in $G = (W, F, E)$. Then the pair (w,f) forms a *blocking pair* with respect to μ if $(w,f) \in E$ and one of the following holds:

- **Type 1.** w,f are both matched in μ and prefer each other to their partners in μ.
- **Type 2a.** w is matched to f', f is unmatched, and $f \succ_w f'$.
- **Type 2b.** w is unmatched, f is matched to w', and $w \succ_f w'$.

Observe that, since $(w,f) \in E$, w and f prefer each other to remaining unmatched. Therefore they cannot both be unmatched in μ – this follows from the maximality of the matching.

The only modification needed to Algorithm 1.8 is to the termination condition; the modification is as follows. Every worker is either tentatively accepted by a firm or has crossed off all firms from its list. When this condition is reached, each worker in the first category is matched to the firm that tentatively accepted it and the rest remain unmatched. Let μ denote this matching. We will still call this the deferred acceptance algorithm. It is easy to see that Observations 1.5 and 1.6 still hold and that Lemma 1.7 holds with a bound nm on the number of iterations.

Lemma 1.20. *The deferred acceptance algorithm outputs a maximal matching in G.*

Proof Assume that $(w,f) \in E$ but that so far worker w and firm f are both unmatched in the matching found by the algorithm. During the algorithm, w must have proposed to f and been rejected. Now, by Observation 1.5, f must be matched, giving a contradiction. □

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

Theorem 1.21. *The maximal matching found by the deferred acceptance algorithm is stable.*

Proof We need to prove that neither type of blocking pair exists with respect to μ. For the first type, the proof is identical to that in Theorem 1.10 and is omitted. Assume that (w,f) is a blocking pair of the second type. There are two cases:

Case 1. w is matched, f is not, and w prefers f to its match, say f'. Clearly w will propose to f before proposing to f'. Now, by Observation 1.5, f must be matched in μ, giving a contradiction.

Case 2. f is matched, w is not, and f prefers w to its match, say w'. Clearly w will propose to f during the algorithm. Since f prefers w to w', it will not reject w in favor of w', hence giving a contradiction. □

Notation. If worker w or firm f is unmatched in μ then we will denote this as $\mu(w) = \perp$ or $\mu(f) = \perp$. We will denote the sets of workers and firms matched under μ by $W(\mu)$ and $F(\mu)$, respectively.

Several of the definitions and facts given in Setting I carry over with small modifications; we summarize these next. The definition of the *realm of possibilities* of workers and firms remains the same as before; however, note that in Setting II, some of these sets could be the singleton set $\{\perp\}$. The definitions of *optimal and pessimal firms for a worker* also remain the same, with the change that they will be $\perp$ if the realm of possibilities is the set $\{\perp\}$. Let $W' \subseteq W$ be the set of workers whose realm of possibilities is non-empty. Then, via a proof similar to that of Lemma 1.13, it is easy to see that two workers in W' cannot have the same optimal firm, i.e., every worker in W' has a unique optimal firm.

Next, match each worker in W' to its optimal firm, leaving the remaining workers unmatched. This is defined to be the *worker-optimal matching*; we will denote it by μ_W. Similarly, define the *firm-optimal matching*; this will be denoted by μ_F. Using ideas from the proof of Lemma 1.15, it is easy to show that the worker-optimal matching is stable. Furthermore, using Theorem 1.17 one can show that the deferred acceptance algorithm finds this matching. Finally, using Lemma 1.18, one can show that the worker-optimal stable matching is also firm pessimal.

Lemma 1.22. *The numbers of workers and firms matched in all stable matchings are the same.*

Proof Each worker w prefers being matched to one of the firms that is its neighbor in G over remaining unmatched. Therefore, all workers who are unmatched in μ_W will be unmatched in all other stable matchings as well. Hence for an arbitrary stable matching μ we have $W(\mu_W) \supseteq W(\mu) \supseteq W(\mu_F)$. Thus $|W(\mu_W)| \geq |W(\mu)| \geq |W(\mu_F)|$. A similar statement for firms is $|F(\mu_W)| \leq |F(\mu)| \leq |F(\mu_F)|$. Since the number of workers and firms matched in any stable matching is equal, $|W(\mu_W)| = |F(\mu_W)|$ and $|W(\mu_F)| = |F(\mu_F)|$. Therefore the cardinalities of all sets given above are equal, hence establishing the lemma. □

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

Finally, we present the *rural hospital theorem*[3] for Setting II.

Theorem 1.23. *The set of workers matched is the same under all stable matchings; similarly for firms.*

Proof As observed in the proof of Lemma 1.22, $W(\mu_W) \supseteq W(\mu) \supseteq W(\mu_F)$. By Lemma 1.22 these sets are of equal cardinality. Hence they must all be the same set as well. □

1.2.3 Reduction from Setting III to Setting II

We will first give a definition of a blocking pair that is appropriate for Setting III. We will then give a reduction from this setting to Setting II, thereby allowing us to carry over the algorithm and its consequences to this setting directly. Finally, we will prove the rural hospital theorem for Setting III.

Definition 1.24. Given a graph $G = (V, E)$ and an upper bound function $b\colon V \to \mathbb{Z}_+$, a set $\mu \subseteq E$ is a *b-matching* if the number of edges of μ incident at each vertex $v \in V$ is at most $b(v)$. Furthermore, μ is a *maximal b-matching* if μ cannot be extended to a valid b-matching by adding an edge from $E - \mu$.

In Setting III, firms have capacities given by $c\colon F \to \mathbb{Z}_+$. For the graph $G = (W, F, E)$ specified in the instance given in Setting III, define an upper bound function $b\colon W \cup F \to \mathbb{Z}_+$ as follows. For $w \in W$, $b(w) = 1$ and for $f \in F$, $b(f) = c(f)$. Let μ be a maximal b-matching in $G = (W, F, E)$ with upper bound function b. We will say that firm f is *matched to capacity* if the number of workers matched to f is exactly $c(f)$ and it is *not matched to capacity* if f is matched to fewer than $c(f)$ workers. Furthermore, if a set $S \subseteq W$ of workers is matched to firm f under μ, with $|S| \le c(f)$, then we will use the notation $\mu(f) = S$ and for each $w \in S$, $\mu(w) = f$.

Definition 1.25. Let μ be a maximal b-matching in $G = (W, F, E)$ with upper bound function b. For $w \in W$ and $f \in F$, (w, f) forms a *blocking pair* with respect to μ if $(w, f) \in E$ and one of the following hold:

- **Type 1.** f is matched to capacity, w is matched to f', and there is a worker w' that is matched to f such that $w \succ_f w'$ and $f \succ_w f'$.
- **Type 2a.** f is not matched to capacity, w is matched to f', and $f \succ_w f'$.
- **Type 2b.** w is unmatched, w' is matched to f, and $w \succ_f w'$.

Reduction to Setting II. Given an instance I of Setting III, we show below how to reduce it in polynomial time to an instance I' of Setting II in such a way that there

[3] The name of this theorem has its origins in the application of stable matching to the problem of matching residents to hospitals. The full scope of the explanation given next is best seen in the context of the extension of this theorem to Setting III, given in Section 1.2.3. In this application it was found that certain hospitals received very poor matches and even remained underfilled; moreover, this persisted even when a hospital-optimal stable matching was used. It turned out that these unsatisfied hospitals were mostly in rural areas and were preferred least by most residents. The question arose whether there was a "better" way of finding an allocation. The rural hospital theorem clarified that *every* stable matching would treat underfilled hospitals in the same way, i.e., give the same allocation.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

is a bijection ϕ between the sets of stable matchings of I and I' such that ϕ and ϕ^{-1} can be computed in polynomial time.

Let I be given by (W, F, E, c) together with preference lists $l(w), \forall w \in W$ and $l(f), \forall f \in F$. Instance I' will be given by (W', F', E') together with preference lists $l'(w), \forall w \in W'$, and $l'(f), \forall f \in F'$, where:

- $W' = W$.
- $F' = \cup_{f \in F} \{f^{(1)}, \ldots, f^{(c(f))}\}$; i.e., corresponding to firm $f \in I$, I' will have $c(f)$ firms, namely $f^{(1)}, \ldots, f^{(c(f))}$.
- Corresponding to each edge $(w, f) \in E$, E' has edges $(w, f^{(i)})$ for each $i \in [1 \ldots c(f)]$.
- $\forall w \in W'$, $l'(w)$ is obtained by replacing each firm, say f, in $l(w)$ by the ordered list $f^{(1)}, \ldots, f^{(c(f))}$. More formally, if $f \succ_w f'$ then for all $1 \le i \le c(i)$ and $1 \le j \le c(j)$ we have $f^{(i)} \succ_w f'^{(j)}$ and for all $1 \le i < j \le c(i)$ we have $f^{(i)} \succ_w f^{(j)}$.
- $\forall f \in F$ and $i \in [1 \ldots c(f)]$, $l'(f^{(i)})$ is the same as $l(f)$.

Lemma 1.26. *Let μ be a stable matching for instance I of Setting III. Then the following hold:*

- *If firm f is matched to $k < c(f)$ workers then $f^{(1)}, \ldots, f^{(k)}$ must be matched and $f^{(k+1)}, \ldots, f^{(c(f))}$ must remain unmatched.*
- *If $(f^{(i)}, w), (f^{(j)}, w') \in \mu$ with $i < j$ then $w \succ_f w'$.*

Proof For contradiction assume that $f^{(i)}$ is unmatched and $f^{(j)}$ is matched, to w say, in μ, where $i < j$. Clearly, $f^{(i)} \succ_w f^{(j)}$ and $w \succ_{f^{(i)}} \perp$. Therefore $(w, f^{(i)})$ is a blocking pair; see Figure 1.7.

The proof of the second statement is analogous, with $\perp$ replaced by w'. □

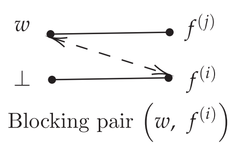


Figure 1.7 Blocking pair $\left(w,\ f^{(i)}\right)$.

Theorem 1.27. *There is a bijection between the sets of stable matchings of I and I'.*

Proof We will first define a mapping ϕ from the first set to the second and then prove that it is a bijection.

Let μ be a stable matching of instance I. Assume that a set $S \subseteq W$ of workers is matched in μ to firm f and worker $w \in S$ and w is the ith most preferred worker in S with respect to $l(f)$. Then, under $\phi(\mu)$ we will match w to $f^{(i)}$. This defines $\phi(\mu)$ completely.

For contradiction assume that $\phi(\mu)$ is not stable and let $(w, f^{(i)})$ be a blocking pair with respect to $\phi(\mu)$. Assume that $f^{(i)}$ is matched to w', where either $w' \in W'$ or $w' = \perp$. Clearly $f^{(i)}$ prefers w to w'; therefore, by construction, if w is matched to $f^{(j)}$ then $j < i$, contradicting the fact that $(w, f^{(i)})$ is a blocking pair. Hence w is either unmatched or is matched to $f'^{(k)}$ for some $f' \neq f$ and some k.

In the first case, under μ, w is unmatched, w' is matched to f, $w \succ_f w'$, and $f \succ_w \perp$. In the second case, under μ, w is matched to f', w' is matched to f, $w \succ_f w'$, and $f \succ_w f'$. Therefore, in both cases (w,f) is a blocking pair with respect to μ, giving a contradiction.

Finally we observe that ϕ has an inverse map $\phi^{-1}(\phi(\mu)) = \mu$. Let μ' be a stable matching of instance I'. If $\mu'(w) = f^{(i)}$ then $\phi^{-1}(\mu')$ matches w to f. The stability of $\phi^{-1}(\mu')$ is easily shown, in particular, because each f has the same preference lists as $f^{(i)}$ for $i \in [1, \ldots, c(f)]$. □

As a consequence of Theorem 1.27, we can transform the given instance I to an instance I' of Setting II, run the deferred acceptance algorithm on it, and transform the solution back to obtain a stable matching for I. Clearly, all notions established in Setting II following the algorithm, such as the realm of possibilities and the worker-optimal and firm-optimal stable matchings, also carry over to the current setting.

Finally, we present the rural hospital theorem for Setting III.

Theorem 1.28. *The following hold for an instance in Setting III:*

1. *Over all the stable matchings of the given instance the set of matched workers is the same and the number of workers matched to each firm is also the same.*
2. *Assume that firm f is not matched to capacity in the stable matchings. Then, the set of workers matched to f is the same over all stable matchings.*

Proof 1. Let us reduce the given instance, say I, to an instance I' in Setting II and apply Theorem 1.23. Then we obtain that the set of matched workers is the same over all the stable matchings of I', hence yielding the same statement for I as well. We also obtain that the set of matched firms is the same over all the stable matchings of I'. Applying the bijection ϕ^{-1} to I' we find that the number workers matched to each firm is also the same over all stable matchings.

2. Let μ_W and μ_F be the worker-optimal and firm-optimal stable matchings for instance I, respectively. Assume for contradiction that firm f is not filled to capacity and $\mu_W(f) \neq \mu_F(f)$. Then there is a worker w who is matched to f in μ_W but not in μ_F. Since μ_W is worker optimal, w prefers f to its match in μ_F. Since f is not filled to capacity, (w,f) forms a blocking pair of Type 2a (see Definition 1.25) with respect to μ_F, giving a contradiction. □

1.3 Incentive Compatibility

In this section we will study incentive compatibility properties of the deferred acceptance algorithm for all three settings. Theorem 1.17 shows that if workers propose to firms, then the matching computed is worker optimal in the sense that each worker is matched to the best firm in its realm of possibilities. However, for an individual worker this "best" firm may be very low in the worker's preference list; see end-of-chapter Exercise 1.7. If so, the worker may have an incentive to cheat, i.e., to manipulate its preference list in order to get a better match.

A surprising fact about the DA algorithm is that this worker will not be able to get a better match by falsifying its preference list. Hence its best strategy is to report its

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

true preference list. Moreover, this holds no matter what preference lists the rest of the workers report. Thus the worker-proposing DA algorithm is *dominant-strategy incentive compatible (DSIC)* for workers.

This ground-breaking result on incentive compatibility opened up the DA algorithm to a host of consequential applications. An example is its use for matching students to public schools in big cities, such as New York and Boston, with hundreds of thousands of students seeking admission each year into hundreds of schools. Previously, Boston was using the immediate acceptance algorithm, which did not satisfy incentive compatibility. It therefore led to much guessing and gaming, making the process highly stressful for the students and their parents. With the use of the student-proposing DA algorithm, each student is best off simply reporting her true preference list. For further details on this application, see Chapter 8.

The proof of Theorem 1.32, showing that Setting I is DSIC, is quite non-trivial and intricate, and more complexity is introduced in Setting II. In this context, the advantage of partitioning the problem into the three proposed settings should become evident.

The rest of the picture for the incentive compatibility of the DA algorithm is as follows. In Setting I, the worker-proposing DA algorithm is not DSIC for firms; see Exercise 1.7. The picture is identical in Settings II and III for the worker-proposing DA algorithm. However, Setting III is asymmetrical for workers and firms, since firms have maximum capacities. For this setting, Theorem 1.35 will establish that there is no mechanism that is DSIC for firms.

1.3.1 Proof of DSIC for Setting I

The following lemma will be critical to proving Theorem 1.32; it guarantees a blocking pair with respect to an arbitrary perfect matching μ. Observe that the blocking pair involves a worker who does not improve its match in going from μ_W to μ.

Lemma 1.29 (Blocking lemma). *Let μ_W be the worker-optimal stable matching under preferences $\succ$ and let μ be an arbitrary perfect matching, not necessarily stable. Further, let W' be the set of workers who prefer their match under μ to their match under μ_W, i.e., $W' = \{w \in W \mid \mu(w) \succ_w \mu_W(w)\}$, and assume that $W' \neq \emptyset$. Then $W' \neq W$ and there exist $w \in (W \setminus W')$ and $f \in \mu(W')$ such that (w, f) is a blocking pair for μ.*

Proof Clearly, for $w \in (W \setminus W')$, $\mu_W(w) \succeq_w \mu(w)$. Two cases arise naturally: whether the workers in W' get better matches in μ over μ_W by simply trading partners, i.e., whether $\mu(W') = \mu_W(W')$ or not. We will study the two cases separately.

Case 1. $\mu(W') \neq \mu_W(W')$. Since $|\mu(W')| = |\mu_W(W')| = |W'|$ and $(\mu(W') \setminus \mu_W(W')) \neq \emptyset$, therefore $W' \neq W$. Pick any $f \in (\mu(W') \setminus \mu_W(W'))$ and let $w = \mu_W(f)$. Now $w \in (W \setminus W')$, since if $f \notin \mu_W(W')$ then $\mu_W(f) \notin W'$. We will show that (w, f) is a blocking pair for μ. To this end, we will identify several other workers and firms.

Let $w' = \mu(f)$; since $f \in \mu(W')$, $w' \in W'$. Let $f'' = \mu_W(w')$; clearly, $f'' \in \mu_W(W')$. Finally let $f' = \mu(w)$; since $w \in (W \setminus W')$, $f' \in (F \setminus \mu(W'))$. Figure 1.8 should be helpful in visualizing the situation.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

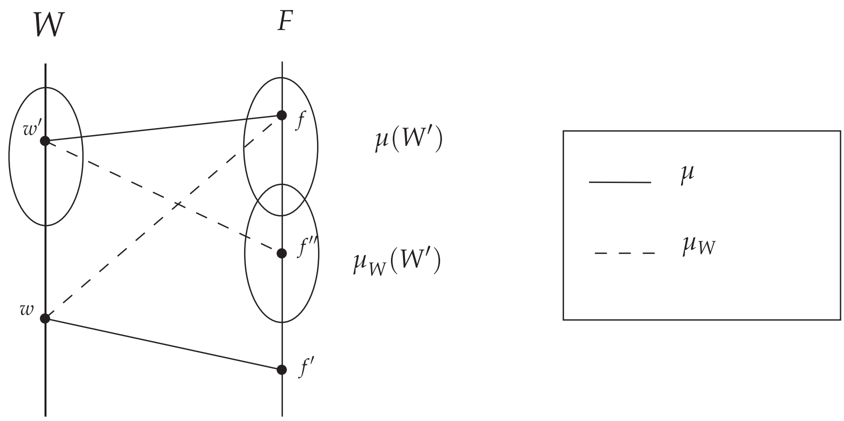


Figure 1.8 Blocking pair $\left(w, f^{(i)}\right)$.

Finally we need to show that $f \succ_w f'$ and $w \succ_f w'$. The first assertion follows on observing that $w \in (W \setminus W')$ and $f \neq f'$. Assume that the second assertion is false, i.e., that $w' \succ_f w$. Now $f \succ_{w'} f''$, since $w' \in W'$, $f = \mu(w')$, and $f'' = \mu_W(w')$. But this implies that (w', f) is a blocking pair for stable matching μ_W. The contradiction proves that $w \succ_f w'$. Hence (w, f) is a blocking pair for μ.

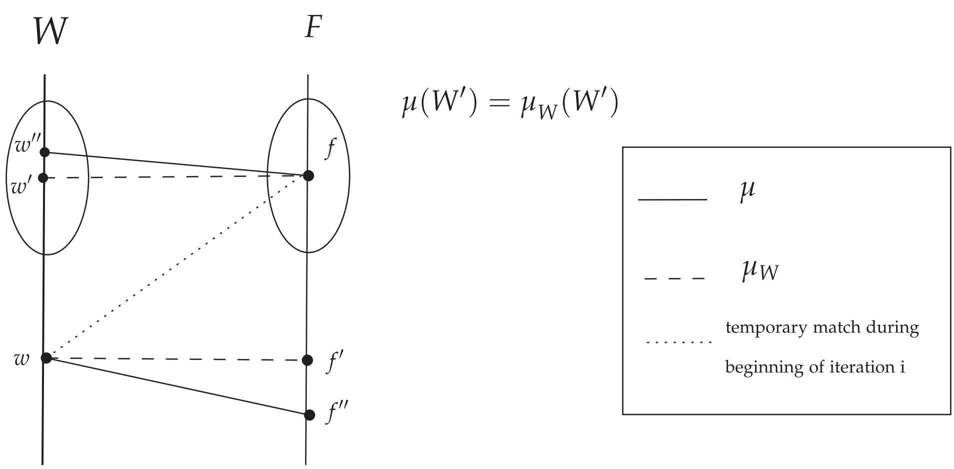


Figure 1.9

Case 2. $\mu(W') = \mu_W(W')$. In this case, unlike the previous one, we will crucially use the fact that μ_W is the matching produced by the DA algorithm with workers proposing. Let i be the last iteration of the DA algorithm in which a worker, say $w' \in W'$, first proposes to its eventual match; let the latter be $f \in \mu_W(W')$. Let $w'' = \mu(f)$. Since $f \in \mu(W')$, $w'' \in W'$.

By the definition of W', $f = \mu(w'') \succ_{w''} \mu_W(w'')$. Therefore, w'' must have proposed to f before iteration i and subsequently moved on to its eventual match under μ_W no later than iteration i. Now, by Observation 1.5, f must keep getting proposals in each iteration after w'' proposed to it. In particular, assume that at the end of iteration $i-1$, f had tentatively accepted the proposal of worker w. In iteration i, f will reject w and w will propose to its eventual match in iteration $i+1$ or later. Since w' is the last worker in W' to propose to its eventual match, $w \notin W'$. Therefore $W' \neq W$. We will show that (w,f) is a blocking pair for μ. Figure 1.9 visualizes the situation.

Since f must have rejected w'' before tentatively accepting the proposal of w, $w \succ_f w''$. Let $f' = \mu_W(w)$ and $f'' = \mu(w)$. In the DA algorithm, w had proposed to f before being finally matched to f', therefore $f \succ_w f'$. Since $w \in (W \setminus W')$, $f' \succeq_w f''$. Therefore $f \succ_w f''$. Together with the assertion $w \succ_f w''$, we get that (w,f) is a blocking pair for μ. $\square$

Notation 1.30. *Assume that a worker w reports a modified list; let us denote it by* $\succ'_w$. *Also assume that all other workers and all firms report their true preference lists. Define, for all* $x \in F \cup W$,

$$\succ'_x \; = \begin{cases} \succ_x & \text{if } x \in F \text{ or } x \in W \setminus \{w\}, \\ \succ'_w & \text{if } x = w. \end{cases}$$

Let μ_W and μ'_W be the worker-optimal stable matchings under the preferences $\succ$ and $\succ'$, respectively. We will use these notions to state and prove Theorem 1.32. However, first we will give the following straightforward observation.

Observation 1.31. *Let* $(W, F, \succ)$ *be an instance of stable matching and consider alternative preference lists* $\succ'_w$ *for every worker* $w \in W$ *and* $\succ'_f$ *for every firm* $f \in F$. *Assume that we are given a perfect matching* μ *which has a blocking pair* (w,f) *with respect to the preferences* $\succ'$. *Moreover, assume that w and f satisfy* $\succ'_w = \succ_w$ *and* $\succ'_f = \succ_f$. *Then* (w,f) *is a blocking pair in* μ *with respect to* $\succ$ *as well.*

Theorem 1.32. *Let* $\succ$ *and* $\succ'$ *be the preference lists defined above and let* μ_W *and* μ'_W *be the worker-optimal stable matchings under these preferences, respectively. Then*

$$\mu_W(w) \succeq_w \mu'_W(w),$$

i.e., the match of worker w, with respect to its original preference list, does not improve if it misrepresents its list as $\succ'_w$.

Proof We will invoke Lemma 1.29; for this purpose, denote μ'_W by μ. Suppose that w prefers its match in μ to its match in μ_W. Let $W' = \{w \in W \mid \mu(w) \succ_w \mu_W(w)\}$; clearly $w \in W'$ and therefore $W' \neq \emptyset$. Now, by Lemma 1.29, there is a blocking pair (w',f) for μ with respect to the preferences $\succ$, with $w' \notin W'$; clearly, $w' \neq w$.

Since $\succ'$ and $\succ$ differ only for w, by Observation 1.31 (w',f) is a blocking pair for μ with respect to $\succ'$ as well. This contradicts the fact that μ is a stable matching with respect to $\succ'$. $\square$

1.3.2 DSIC for Setting II

While studying incentive compatibility for the case of incomplete preference lists, we will allow a worker w not only to alter its preference list over its neighbors in graph G but also to alter its set of neighbors, i.e., to alter G itself. For this reason, it will be more convenient to define the preference list of each worker over the set $F \cup \{\perp\}$ and of each firm over the set $W \cup \{\perp\}$, as stated in footnote 1 in Section 1.1.

Let $\succ$ denote the original preference lists of workers and firms and let μ_W denote the worker-optimal stable matching under preferences $\succ$. The definition of a *blocking pair* with respect to a matching μ is changed in one respect only, namely $\mu(w) = \perp$ or $\mu(f) = \perp$ is allowed. Thus (w,f) is a blocking pair if and only if $(w,f) \notin \mu$, $\mu(w) \succ_w f$, and $\mu(f) \succ_f w$.

The only change needed to the statement of the blocking lemma, stated as Lemma 1.29 for Setting I, is that μ is an arbitrary matching, i.e., it is not necessarily perfect. Once again, the proof involves the same two cases presented in Lemma 1.29. The proof of the first case changes substantially and is given below.

Proof *Case 1:* $\mu(W') \neq \mu_W(W')$. For $w \in W'$, $\mu(w) \succ_w \mu_W(w)$, therefore $\mu(w) \neq \perp$. However, $\mu_W(w) = \perp$ is allowed. Therefore $|\mu(W')| = |W'| \geq |\mu_W(W')|$. Hence, $\mu(W') \not\subset \mu_W(W')$ and, since $\mu(W') \neq \mu_W(W')$, we obtain $(\mu(W') \setminus \mu_W(W')) \neq \emptyset$. Pick any $f \in (\mu(W') \setminus \mu_W(W'))$. Clearly $\mu(f) \neq \perp$; let $w' = \mu(f)$. Let $\mu_W(w') = f''$, where $f'' = \perp$ is possible. By definition of W', $f \succ_{w'} f'' = \mu_W(w')$.

Assume for contradiction that $\mu_W(f) = \perp$. Since $\mu(f) = w'$, $w' \succ_f \perp$. Therefore we obtain $w' \succ_f \mu_W(f)$. If so, (w',f) is a blocking pair for μ_W, leading to a contradiction. Therefore $\mu_W(f) \neq \perp$. Let $\mu_W(f) = w$. Clearly, $w \notin W'$. Hence $W' \neq W$. Let $\mu(w) = f'$, where $f' = \perp$ is possible. Clearly, $f \neq f'$ and $f \succ_w f'$. This, together with the assertion $w \succ_f w'$, gives us that (w,f) is a blocking pair for μ. □

Now consider Case 2, namely $\mu(W') = \mu_W(W')$. Since, for each $w \in W'$, we have $\mu(w) \succ_w \mu_W(w)$, w cannot be unmatched under μ. Therefore, in this case, $\forall w \in W'$, w is matched in both μ and μ_W. The rest of the proof is identical to that of Lemma 1.29, other than the fact that $f' = \perp$ and $f'' = \perp$ are allowed. We will not repeat the proof here. This establishes the blocking lemma for Setting II.

As in Setting I, presented in Section 1.3.1, we will adopt the notation $\succ$ and $\succ'$ given in Notation 1.30. Let μ'_W be the worker-optimal stable matchings under the preferences $\succ'$. Note that Observation 1.6 still holds. Finally, the statement of Theorem 1.32 carries over without change to this setting and its proof is also identical, provided one uses the slightly modified statement of the blocking lemma.

1.3.3 DSIC for Setting III

For Setting III, the DSIC algorithm for workers follows easily using the reduction from Setting II to Setting III, given in Section 1.2.3, and the fact that the worker-proposing DA algorithm is DSIC for workers. However, this setting is asymmetric for workers and firms, since firms can match to multiple workers. Therefore, we need to study the incentive compatibility of the firm-proposing DA algorithm as well.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

The answer is quite surprising: not only is this algorithm not DSIC, but in fact no mechanism can be DSIC for this setting.

Example 1.33. Let the set of workers be $W = \{w_1, w_2, w_3, w_4\}$ and the set of firms be $F = \{f_1, f_2, f_3\}$. Suppose that firm f_1 has capacity 2 and firms f_2 and f_3 have unit capacities. Preferences are given by:

$$\begin{aligned}
f_1 &: w_1 \succ w_2 \succ w_3 \succ w_4 \\
f_2 &: w_1 \succ w_2 \succ w_3 \succ w_4 \\
f_3 &: w_3 \succ w_1 \succ w_2 \succ w_4 \\
w_1 &: f_3 \succ f_1 \succ f_2 \\
w_2 &: f_2 \succ f_1 \succ f_3 \\
w_3 &: f_1 \succ f_3 \succ f_2 \\
w_4 &: f_1 \succ f_2 \succ f_3
\end{aligned}$$

Consider the instance given in Example 1.33. If firms propose to workers, the resulting stable matching, μ_F, assigns $\mu_F(f_1) = \{w_3, w_4\}$, $\mu_F(f_2) = \{w_2\}$, and $\mu_F(f_3) = \{w_1\}$.

Next consider matching μ with $\mu(f_1) = \{w_2, w_4\}$, $\mu(f_2) = \{w_1\}$, and $\mu(f_3) = \{w_3\}$. This matching is strictly preferred by all firms but it is not stable since (w_1, f_1) and (w_3, f_1) are both blocking pairs. However, if f_1 misrepresents its preferences as

$$f_1 : w_2 \succ' w_4 \succ' w_1 \succ' w_3,$$

then μ becomes stable and firm optimal. Hence we get:

Lemma 1.34. *For the instance given in Example 1.33, there exists a matching μ which is not stable and which all firms strictly prefer to the firm-optimal stable matching μ_F. Moreover, there is a way for one firm to misrepresent its preferences so that μ becomes stable.*

The next theorem follows.

Theorem 1.35. *For the stable matching problem in Setting III with capacitated firms, there is no mechanism that is DSIC for firms.*

1.4 The Lattice of Stable Matchings

The notions of worker-optimal and worker-pessimal stable matchings, defined in Section 1.2.1, indicate that the set of all stable matchings of an instance has structure. It turns out that these notions form only the tip of the iceberg! Below we will define the notion of a *finite distributive lattice* and will prove that the set of stable matchings of an instance forms such a lattice. Together with the non-trivial notion of *rotation* and Birkhoff's representation theorem, this leads to an extremely rich collection of structural properties and efficient algorithms which find a use in important applications.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

1.4.1 The Lattice for Setting I

Definition 1.36. Let S be a finite set, $\geq$ be a reflexive, anti-symmetric, transitive relation on S and $\pi = (S, \geq)$ be the corresponding partially ordered set. For any two elements $a, b \in S$, $u \in S$ is said to be an *upper bound* of a and b if $u \geq a$ and $u \geq b$. Further, u is said to be a *least upper bound* of a and b if $u' \geq u$ for any upper bound u' of a and b. The notion of the *(greatest) lower bound* of two elements is analogous. The partial order π is said to be a *lattice* if any two elements $a, b \in S$ have a unique least upper bound and a unique greatest lower bound. If so, these will be called the *join* and *meet* of a and b and will be denoted by $a \vee b$ and $a \wedge b$, respectively, and the partial order will typically be denoted by $\mathcal{L}$. Finally, $\mathcal{L}$ is said to be a *finite distributive lattice*, abbreviated *FDL*, if for any three elements $a, b, c \in S$, the distributive property holds, i.e.,

$$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c) \quad \text{and} \quad a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c).$$

Birkhoff's representation theorem, mentioned above, holds for FDLs. We next define a natural partial order on the set of stable matchings of an instance and show that it forms such a lattice.

Definition 1.37. Let S_μ be the set of stable matchings of a given instance in Setting I. Define a relation $\geq$ on S_μ as follows: for $\mu, \mu' \in S_\mu$, $\mu \geq \mu'$ if and only if every worker w weakly prefers her match in μ to her match in μ', i.e.,

$$\forall w \in W: \ \mu(w) \succeq_w \mu'(w).$$

Theorem 1.41 will show that the partial order $\mathcal{L}_\mu = (S_\mu, \geq)$ is a finite distributive lattice; $\mathcal{L}_\mu$ will be called the *stable matching lattice* for the given instance.

Let μ and μ' be two stable matchings. We define the following four operations. For worker w, $\max\{\mu(w), \mu'(w)\}$ is the firm that w weakly prefers among $\mu(w)$ and $\mu'(w)$, and $\min\{\mu(w), \mu'(w)\}$ is the firm that w weakly dislikes, where "dislikes" is the opposite of the relation "prefers". For a firm f, $\max\{\mu(f), \mu'(f)\}$ and $\min\{\mu(f), \mu'(f)\}$ are analogously defined.

Define two maps $M_W: W \to F$ and $M_F: F \to W$ as follows:

$$\forall w \in W: \ M_W(w) = \max\{\mu(w), \mu'(w)\} \quad \text{and} \quad \forall f \in F: \ M_F(f) = \min\{\mu(f), \mu'(f)\}.$$

Lemma 1.38. *$\forall w \in W$, if $M_W(w) = f$ then $M_F(f) = w$.*

Proof Assume $\mu(w) \neq \mu'(w)$, since otherwise the proof is obvious. Let $\mu(w) = f$ and $\mu'(w) = f'$, and without loss of generality assume that $f \succeq_w f'$. Let $\mu'(f) = w'$; clearly $w \neq w'$. Now if $w \succeq_f w'$ then (w, f) is a blocking pair for μ', leading to a contradiction. Therefore, $w' \succeq_f w$ and hence $M_F(f) = w$; see Figure 1.10 for an illustration. □

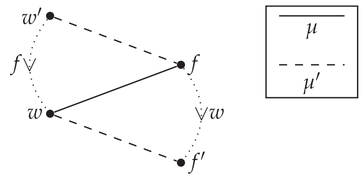


Figure 1.10 Illustration of the Figure for proof of Lemma 1.38.

Corollary 1.39. *M_W and M_F are both bijections, and $M_W = M_F^{-1}$.*

As a consequence of Corollary 1.39, M_W is a perfect matching on $W \cup F$; denote it by μ_1. Analogously, mapping each worker w to $\min\{\mu(w), \mu'(w)\}$ gives another perfect matching; denote it by μ_2. Observe that μ_2 matches each firm f to $\max\{\mu(f), \mu'(f)\}$; see Figure 1.11.

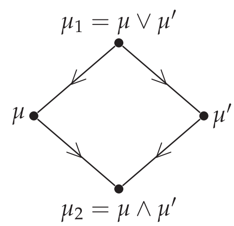


Figure 1.11 The meet and join of μ and μ'.

Lemma 1.40. *The matchings μ_1 and μ_2 are both stable.*

Proof Assume that (w,f) is a blocking pair for μ_1. Let $\mu(w)=f'$ and $\mu'(w)=f''$ and assume without loss of generality that $f' \succeq_w f''$. Then $\mu_1(w) = \max\{\mu(w), \mu'(w)\} = f'$.

Let $\mu(f)=w'$ and $\mu'(f)=w''$. By the definition of map M_F, $w'' \succeq_f w'$ and $\mu_1(f) = w'$; observe that $w \neq w'$. Since (w,f) is a blocking pair for μ_1, we have $w \succeq_f w'$. This implies that (w,f) is a blocking pair for μ, leading to a contradiction. Hence μ_1 is stable. An analogous argument shows that μ_2 is also stable; see Figure 1.12 for an illustration. □

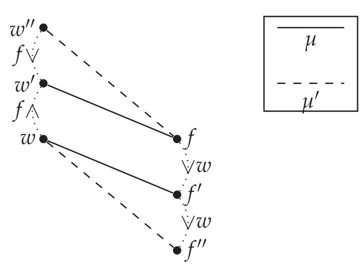


Figure 1.12 Illustration of the proof of Lemma 1.40.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

Now consider the partial order $\mathcal{L}_\mu = (S_\mu, \geq)$ defined in Definition 1.37. Clearly, μ_1 and μ_2 are an upper bound and a lower bound of μ and μ', respectively. It is easy to see that they are also the unique lowest upper bound and the unique greatest lower bound of μ and μ'. Therefore $\mathcal{L}_\mu$ supports the operations of meet and join given by

$$\mu \vee \mu' = \mu_1 \quad \text{and} \quad \mu \wedge \mu' = \mu_2,$$

Finally, it is easy to show that the operations of meet and join satisfy the distributive property; see Exercise 1.8. Hence we get:

Theorem 1.41. *The partial order $\mathcal{L}_\mu = (S_\mu, \geq)$ is a finite distributive lattice.*

Remark 1.42. If $\mu, \mu' \in S_\mu$ with $\mu > \mu'$ then, by definition, workers get weakly better matches in μ than in μ'. The discussion presented above implies that firms get weakly worse matches.

Using the finiteness of $\mathcal{L}_\mu$, it is easy to show that there are two special matchings, say $\mu_\top, \mu_\perp \in S_\mu$, which we will call *top* and *bottom* matchings, respectively, such that, $\forall \mu \in S_\mu$, $\mu_\top \geq \mu$ and $\mu \geq \mu_\perp$. These stable matchings were already singled out in Section 1.2.1: $\mu_\top$ is the worker-optimal and firm-pessimal matching and $\mu_\perp$ is the firm-optimal and worker-pessimal matching; see Figure 1.13.

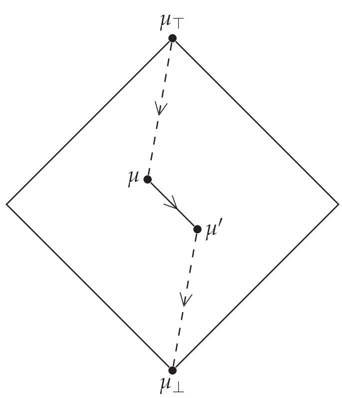


Figure 1.13 A path from the worker-optimal matching, $\mu_\top$, to the firm-optimal matching, $\mu_\perp$, in the lattice $\mathcal{L}_\mu$. The edge (μ, μ') indicates that $\mu' = \rho(\mu)$, for a rotation ρ with respect to μ.

1.4.1.1 Rotations, and Their Use for Traversing the Lattice

Several applications require stable matchings which are not as "extreme" as $\mu_\top$ and $\mu_\perp$, i.e., they treat the two sets W and F more equitably. These stable matchings can be found in the rest of the lattice. In this subsection we will define the notion of a rotation, which helps to traverse the lattice. In particular, we will prove that rotations help to traverse paths from $\mu_\top$ to $\mu_\perp$, as illustrated in Figure 1.13, with intermediate "vertices" on such a path being stable matchings and an "edge" (μ, μ') indicating that $\mu > \mu'$; see Lemma 1.45 and Corollary 1.46. By Remark 1.42, the intermediate matchings on any such path will gradually become better for firms and worse for workers. For an example of the use of rotations, see Exercise 1.13, which develops an efficient algorithm for finding a stable matching that treats workers and firms more equitably.

Definition 1.43. Fix a stable matching $\mu \neq \mu_\perp$ and define the function next: $W \rightarrow F \cup \{\boxtimes\}$ as follows: for a worker w, find its most preferred firm, say f, such that f prefers w to $\mu(f)$. If such a firm exists then next$(w) = f$ and otherwise next$(w) = \boxtimes$. A *rotation* ρ with respect to μ is an ordered sequence of pairs $\rho = \{(w_0,f_0), (w_1,f_1), \ldots, (w_{r-1},f_{r-1})\}$ such that, for $0 \leq i \leq r-1$:

- $(w_i,f_i) \in \mu$, and
- next$(w_i) = f_{(i+1) \pmod r}$

By *applying rotation ρ to μ* we mean switching the matching of $w_0, \ldots, w_{r-2}, w_{r-1}$ to $f_1, \ldots, f_{r-1}, f_0$, respectively, and leaving the rest of μ unchanged. Clearly this results in a perfect matching; let us denote it by μ'. We will denote this operation as $\mu' = \rho(\mu)$. For example, in Figure 1.14, the rotation $\{(1,1),(2,2),(3,4),(4,5)\}$ applied to μ yields μ'. Observe that the matched edge $(5,3)$ is not in the rotation and remains unchanged in going from μ to μ'.

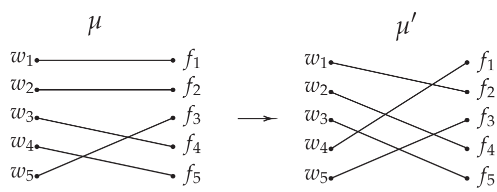


Figure 1.14

Lemma 1.44. *Let ρ be a rotation with respect to μ and let $\mu' = \rho(\mu)$. Then:*

1. *Workers get weakly worse matches and firms get weakly better matches in going from μ to μ'.*
2. *μ' is a stable matching.*
3. *$\mu > \rho(\mu)$.*

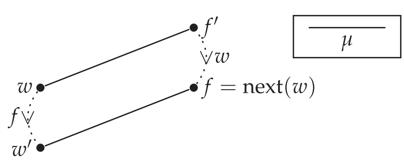


Figure 1.15

Proof 1. Suppose that next$(w) = f$; clearly $\mu(f) \neq w$. Let $\mu(w) = f'$ and $\mu(f) = w'$. By the definition of the operator next, $w \succ_f w'$. Observe that $f \succ_w f'$ is not possible, since then (w,f) would be a blocking pair for μ. Therefore $f' \succ_w f$. Hence, after the rotation, the matching of f is improved and that of w becomes worse. Clearly, this holds for all workers and firms in the rotation; see Figure 1.15.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

2. Assume that μ' has a blocking pair, namely (w, f), where $\mu'(w) = f'$ and $\mu'(f) = w'$. From this blocking pair, we can infer that $w \succ_f w'$ and $f \succ_w f'$. Now there are two cases:

Case (a). $\mu(w) = f'$. Assume that $\mu(f) = w''$; by part (1) of this lemma we have $w' \succ_f w''$. Since $w \succ_f w'$, we get that $w \succ_f w''$, hence showing that (w, f) is a blocking pair for μ and leading to a contradiction; see Figure 1.16.

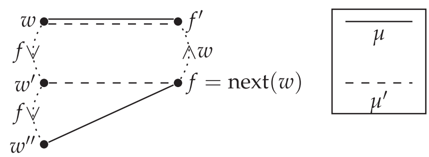


Figure 1.16 Illustration of proof of Case (a) in Lemma 1.44.

Case (b). $\mu(w) \neq f'$. If $\mu(f) \neq w'$ then let $\mu(f) = w''$. Since f improves its match after the rotation, we have $w' \succeq_f w''$ and, and since $w \succ_f w''$, we get that $w \succeq_f w''$. Therefore, whether or not $\mu(f) = w'$, we have that $w \succ \mu(f)$.Clearly, $\text{next}(w) = f'$. Since f prefers w to its match in μ and $f \neq \text{next}(w)$, we get that $f' \succ_w f$, hence contradicting the above-stated assertion that $f \succ_w f'$; see Figure 1.17.

3. This follows from the previous two statements. Assume that $\text{next}(w) \neq \perp$, i.e., $\text{next}(w)$ is a firm. We note that this does not guarantee that w will be in a rotation; see Exercise 1.10. For that to happen, the "cycle must close", i.e., for some r, $\text{next}(w_{r-1}) = f_0$. □

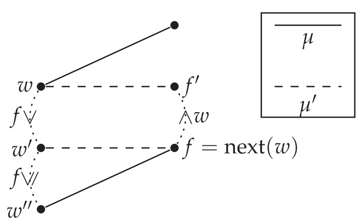


Figure 1.17 Illustration of the proof of Case (b) in Lemma 1.44.

The next lemma will justify Figure 1.13 via Corollary 1.46.

Lemma 1.45. *Let $\mu > \mu'$. Then there exists a rotation ρ with respect to μ such that $\mu > \rho(\mu) \geq \mu'$.*

Proof Define a map $g\colon W \to W \cup \{\boxtimes\}$ as follows:

$$g(w) = \begin{cases} \mu(\text{next}(w)) & \text{if next}(w) \in F, \\ \boxtimes & \text{otherwise.} \end{cases}$$

Let $W' = \{w \in W \mid \mu(w) \succ \mu'(w)\}$. We will prove that the range of g when restricted to W' is W' and, for $w \in W'$, $g(w) \neq w$.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

Let $w \in W'$. We will first prove that $\text{next}(w) \in F$, hence showing that $g(w) \in W$. Let $\mu'(w) = f'$ and $\mu(f') = w''$. Since $\mu > \mu'$, μ is weakly better than μ' for workers. Therefore $f' \succ_{w''} \mu'(w'')$. If $w'' \succ_{f'} w$ then (w'', f') will be a blocking pair for μ', leading to a contradiction. Therefore, $w \succ_{f'} w''$. Hence there is a firm that likes w better than its own match under μ. Among such firms, let f be one that w prefers most. Then $\text{next}(w) = f$ and $g(w) = \mu(f) = w'$, say.

Next, we will show that $\mu(w') \succ \mu'(w')$, hence obtaining $w' \in W'$; see Figure 1.18. Suppose that this is not the case. Since $\mu > \mu'$ and μ is not strictly better than μ' for w', we must have that $\mu'(w') = \mu(w') = f$. Let $\mu'(w) = f'$. Since $\mu > \mu'$, we have that $w \succ_{f'} \mu(f')$. Therefore f' is a firm that prefers w to its match under μ. However, since f is the most preferred such firm for worker w, we get $f \succ_w f'$. Furthermore, since $\mu(f) = w'$, $w \succ_f w'$. Therefore (w, f) is a blocking pair for μ', leading to a contradiction. Therefore, $\mu(w') \succ \mu'(w')$ and hence $w' \in W'$. Clearly w' is distinct from w, hence giving $g(w) \neq w$.

Finally, we will use the map $g\colon W' \to W'$ to complete the proof. Start with any worker $w \in W'$ and repeatedly apply g until a worker is encountered for a second time. This gives us a "cycle", i.e., a sequence of workers $w_0, w_1, \ldots, w_{r-1}$ such that, for $0 \leq i \leq r-1$, $g(w_i) = w_{i+1 (\text{mod}\, r)}$. Then $\rho = \{(w_0, f_0), (w_1, f_1), \ldots, (w_{r-1}, f_{r-1})\}$ is a rotation with respect to μ, where $f_i = \mu(w_i)$ for $0 \leq i \leq r-1$. Clearly, $\mu > \rho(\mu) \geq \mu'$. $\square$

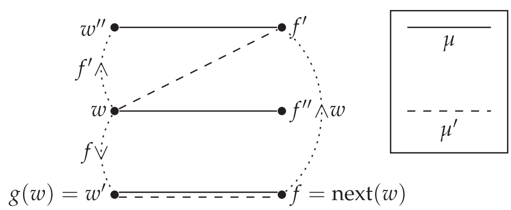


Figure 1.18 Illustration of why $w' \in W'$ in the proof of Lemma 1.45.

Corollary 1.46. *The following hold:*

1. *Let μ be a stable matching such that $\mu \neq \mu_\perp$. Then there is a rotation ρ with respect to μ.*
2. *Start with $\mu_\top$ as the "current matching" and keep applying an arbitrary rotation with respect to the current matching. This process will terminate at $\mu_\perp$.*

Let $G_\mu = (S_\mu, E_\mu)$ be a directed graph with vertex set S_μ and $(\mu, \mu') \in E_\mu$ if and only if there is a rotation ρ with respect to μ such that $\rho(\mu) = \mu'$. Then any path from $\mu_\top$ to $\mu_\perp$ is obtained by the process given in Corollary 1.46.

Definition 1.47. Let ρ be a rotation with respect to a stable matching μ and let $\rho(\mu) = \mu'$. Then the inverse of the map ρ is denoted by ρ^{-1}. We will call ρ^{-1} the *inverse rotation* with respect to the stable matching μ'. Clearly, $\rho^{-1}(\mu') = \mu$.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

Inverse rotations help traverse paths from $\mu_\perp$ to $\mu_\top$ in G_μ, and a combination of rotations and inverse rotations suffices to find a path from any one matching to any other matching in G_μ.

1.4.1.2 Rotations Correspond to Join-Irreducible Stable Matchings

Definition 1.48. A stable matching μ is said to be *join-irreducible* if $\mu \neq \mu_\perp$ and μ is not the join of any two stable matchings. Let μ and μ' be two stable matchings. We will say that μ' is the *direct successor* of μ if $\mu > \mu'$ and there is no stable matching μ'' such that $\mu > \mu'' > \mu'$; if so, we will denote this by $\mu \rhd \mu'$.

Let μ be a join-irreducible stable matching. By Lemma 1.44 and Corollary 1.46 there is a unique rotation ρ with respect to μ. Let $\rho(\mu) = \mu'$. Clearly μ' is the unique stable matching such that $\mu \rhd \mu'$.

Lemma 1.49. *Let ρ_1 and ρ_2 be two distinct rotations with respect to the stable matching μ, and let $\mu_1 = \rho_1(\mu)$ and $\mu_2 = \rho_2(\mu)$. Then the following hold:*

1. *ρ_1 and ρ_2 cannot contain the same worker–firm pair.*
2. *ρ_1 and ρ_2 are rotations with respect to μ_2 and μ_1, respectively.*

Proof 1. Clearly, $\mu > \mu_1$ and $\mu > \mu_2$. Consider the set W' and the map $g\colon W' \to W'$ defined in the proof of Lemma 1.45. It is easy to see that ρ_1 and ρ_2 correspond to two disjoint "cycles" in this map, hence giving the lemma.

2. Since ρ_1 and ρ_2 correspond to disjoint "cycles", applying one results in a matching to which the other can be applied. □

Notation 1.50. *We will denote the set of all rotations used in a lattice $\mathcal{L}_\mu$ by $\mathcal{R}_\mu$, and the set of join-irreducible stable matchings of $\mathcal{L}_\mu$ by $\mathcal{J}_\mu$.*

Lemma 1.51. *Let $\rho \in \mathcal{R}_\mu$. Then there is a join-irreducible stable matching, say μ, such that ρ is the unique rotation with respect to μ.*

Proof Let $S_\rho = \{\nu \in S_\mu \mid \rho \text{ is a rotation with respect to } \nu\}$. Let μ be the meet of all matchings in S_ρ. Clearly $\mu \in S_\rho$ and hence ρ is a rotation with respect to μ. Suppose that μ is not join-irreducible and let ρ' be another rotation with respect to μ. By the second part of Lemma 1.49, ρ is a rotation with respect to $\rho'(\mu)$, therefore $\rho'(\mu) \in S_\rho$. This implies that $\rho'(\mu) \geq \mu$. But $\mu > \rho'(\mu)$, giving a contradiction. Uniqueness follows from the fact that μ is join-irreducible. □

Lemma 1.52. *Two rotations ρ and ρ' cannot contain the same worker–firm pair.*

Proof Suppose that rotations ρ and ρ' do both contain the same worker–firm pair, say (w,f). By Lemma 1.51, there are two join-irreducible stable matchings μ and ν such that ρ and ρ' are rotations with respect to these matchings. Let $\mu \rhd \mu'$ and $\nu \rhd \nu'$.

Consider the matching $\mu \vee \nu$. Since $(w,f) \in \mu$ and $(w,f) \in \nu$, we have $(w,f) \in (\mu \vee \nu)$. On the other hand, since (w,f) is in rotations ρ and ρ', w is matched to a

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

worse firm than f in μ' and in ν'. Therefore w is matched to a worse firm than f in $\mu' \vee \nu'$. But since μ and ν are join-irreducible, $(\mu' \vee \nu') = (\mu \vee \nu)$, leading to a contradiction. Hence ρ and ρ' cannot contain the same worker–firm pair. □

Lemmas 1.51 and 1.52, together with the facts that there are n^2 worker–firm pairs and each rotation has at least two worker–firm pairs, give:

Corollary 1.53. *The following hold:*

1. *There is a bijection $f\colon \mathcal{J}_\mu \to \mathcal{R}_\mu$ such that if $f(\mu) = \rho$ then ρ is the unique rotation with respect to μ.*
2. $|\mathcal{R}_\mu| \leq n^2/2$.

1.4.1.3 Birkhoff's Representation Theorem

Finite distributive lattices arise in diverse settings; Definition 1.54 below gives perhaps the simplest of these. A consequence of Birkhoff's representation theorem is that each FDL is isomorphic to a canonical FDL. In this section we will prove this for stable matching lattices; the general statement follows along similar lines.

Definition 1.54. Let S be a finite set and $\mathcal{F}$ be a family of subsets of S which is closed under union and intersection. Denote the partial order $(\mathcal{F}, \supseteq)$ by $\mathcal{L}_\mathcal{F}$. Then $\mathcal{L}_\mathcal{F}$ is an FDL with meet and join given by

$$A \wedge B = A \cap B \quad \text{and} \quad A \vee B = A \cup B,$$

for any two sets $A, B \in \mathcal{F}$; $\mathcal{L}_\mathcal{F}$ will be called a *canonical finite distributive lattice*.

Definition 1.55. The projection of $\mathcal{L}_\mu$ onto $\mathcal{J}_\mu$ is called a *join-irreducible partial order* and is denoted by $\pi_\mu = (\mathcal{J}_\mu, \geq)$. We will say that $S \subseteq \mathcal{J}_\mu$ is a *lower set of* π_μ if it satisfies the following: if $\mu \in S$ and $\mu > \mu'$ then $\mu' \in S$.

Let $\mathcal{F}_\pi$ be the family of subsets of $\mathcal{J}_\mu$ consisting of all lower sets of π_μ. It is easy to see that $\mathcal{F}_\pi$ is closed under union and intersection, and therefore that $\mathcal{L}_\pi = (\mathcal{F}_\mu, \supseteq)$ is a canonical FDL.

Theorem 1.56. *The lattice $\mathcal{L}_\mu$ is isomorphic to $\mathcal{L}_\pi$, i.e., there is a bijection $f_\mu\colon S_\mu \to \mathcal{F}_\pi$ such that $\mu \succeq \mu'$ if and only if $f_\mu(\mu) \supseteq f_\mu(\mu')$.*

Proof Define a function $f_\mu\colon S_\mu \to \mathcal{F}_\pi$ as follows. For $\mu \in S_\mu$, $f(\mu)$ is the set of all join-irreducible stable matchings ν such that $\mu \succ \nu$; let this set be S. Then S is a lower set of π_μ since if $\mu_1, \mu_2 \in \mathcal{J}_\mu$, with $\mu_1 \in S$ and $\mu_1 \succ \mu_2$ then $\mu \succ \mu_2$, therefore giving that $\mu_2 \in S$. Hence $S \in \mathcal{F}_\pi$. Next define a function $g\colon \mathcal{F}_\pi \to S_\mu$ as follows. For a lower set S of π_μ, $g(S)$ is the join of all join-irreducibles $\nu \in S$.

We first show that the compositions $g \bullet f$ and $f \bullet g$ both give the identity function, thereby showing that f and g are both bijections. Then $f_\mu = f$ is the required bijection.

Let $\mu \in S_\mu$ and let $f(\mu) = S$. For the first composition, we need to show that $g(S) = \mu$. There exist $j_1, \ldots, j_k$, join-irreducibles of $\mathcal{L}_\mu$, such that $\mu = (j_1 \vee \cdots \vee j_k)$.[4] Clearly, $j_1, \ldots, j_k \in S$ and therefore $g(S) \succeq (j_1 \vee \ldots \vee j_k)$. Furthermore, $\mu \succeq g(S)$. Therefore, $\mu \succeq g(S) \succeq (j_1 \vee \ldots \vee j_k) = \mu$. Hence $g(S) = \mu$.

Let S be a lower set of π_μ, let $j_1, \ldots, j_k$ be the join-irreducible stable matchings in S, and let μ be $g(S)$, i.e., the join of these join-irreducibles. Let j be a join-irreducible of $\mathcal{L}_\mu$ such that $\mu \succeq j$. For the second composition, we need to show that $j \in S$, since then $f(\mu) = S$. Now,

$$j \wedge \mu = j \wedge (j_1 \vee \cdots \vee j_k) = (j \wedge j_1) \vee \cdots \vee (j \wedge j_k),$$

where the second equality follows from the distributive property. Since j is a join-irreducible, it cannot be the join of two or more elements. Therefore $j = j \wedge j_i$ for some i. But then $j_i \succeq j$, therefore giving $j \in S$.

Finally, the definitions of f and g give that $\mu \succeq \mu'$ if and only if $f_\mu(\mu) \supseteq f_\mu(\mu')$. $\square$

Observe that an instance may have exponentially many, in n, stable matchings, hence leading to an exponentially large lattice. On the other hand, by Corollary 1.53 it follows that π_μ, which encodes this lattice, has a polynomial sized description. The precise way in which π_μ encodes $\mathcal{L}_\mu$ is clarified in Exercise 1.11.

Corollary 1.57. *There is a succinct description of the stable matching lattice.*

1.4.2 The Lattice for Settings II and III

The entire development on the lattice of stable matchings for Setting I, presented in Section 1.4.1, can be easily ported to Setting II with the help of the rural hospital theorem, Theorem 1.23, which proves that the sets of workers and firms that have been matched are the same in all stable matchings.

Let these sets be $W' \subseteq W$ and $F' \subseteq F$. Let $w \in W'$. Clearly, it suffices to restrict the preference list of w to F' only. If this list is not complete over F', simply add the missing firms of F' at the end; since w is never matched to these firms, their order does not matter. Applying this process to each worker and each firm yields an instance of stable matching over W' and F' which is in Setting I. The lattice for this instance is also the lattice for the original instance in Setting II.

The lattice for Setting III requires a reduction from Setting III to Setting II, given in Section 1.2.3, and the Rural Hospital theorem for Setting III, given in Theorem 1.28. The latter proves that if a firm is not matched to capacity in a stable matching then it is matched to the same set of workers in all stable matchings. However, a firm that is matched to capacity may be matched to different sets of workers in different stable matchings.

[4] Observe that every $\mu \in S_\mu$ can be written as the join of a set of join-irreducible stable matching as follows: If μ is a join-irreducible stable matchings, there is nothing to prove. Otherwise, it is the join of two or more matchings. Continue the process on each of those matchings.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

Assume that firm f is matched to capacity in μ_1 and μ_2. A new question that arises is the following: which of these two matchings does f prefer? Observe that questions of this sort have natural answers for workers and firms in the previous settings (and for workers in Setting III) and that these answers were a key to formulating the lattice structure. The new difficulty is the following: if the two sets $\mu_1(f)$ and $\mu_2(f)$ are interleaved in complicated ways, when viewed with respect to the preference order of f we will have no grounds for declaring one matching better than the other. Lemma 1.60 shows that if μ_1 and μ_2 are stable matchings then such complications do not arise.

Definition 1.58. Fix a firm f. For $W' \subseteq W$, define $\min(W')$ to be the worker whom f prefers the least among the workers in W'. For $W_1 \subseteq W$ and $W_2 \subseteq W$, we will say that f *prefers* W_1 to W_2 if f prefers every worker in W_1 to every worker in W_2 and we will denote this as $W_1 \gg_f W_2$. Thus

$$W_1 = \{w \in W \mid \mu_1(w) \succ_w \mu_2(w)\} \quad \text{and} \quad W_2 = \{w \in W \mid \mu_2(w) \succ_w \mu_1(w)\}.$$

Also let

$$F_1 = \{f \in F \mid \min(\mu_2(f) - \mu_1(f)) \gg_f \min(\mu_1(f) - \mu_2(f))\} \quad \text{and}$$
$$F_2 = \{f \in F \mid \min(\mu_1(f) - \mu_2(f)) \gg_f \min(\mu_2(f) - \mu_1(f))\}.$$

Lemma 1.59. *Let* $(w,f) \in (\mu_1 - \mu_2)$. *Then*

1. $w \in W_1 \implies f \in F_1$
2. $w \in W_2 \implies f \in F_2$.

Lemma 1.60. *Let μ_1 and μ_2 be two stable matchings and f be a firm that is matched to capacity in both matchings. Then one of these possibilities must hold:*

1. $\mu_1(f) - \mu_2(f) \gg_f \mu_2(f)$ *(see Figure 1.19) or*
2. $\mu_2(f) - \mu_1(f) \gg_f \mu_1(f)$.

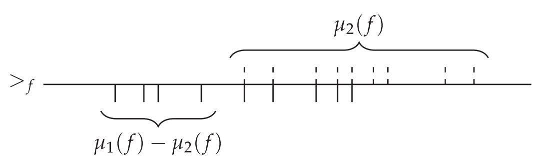


Figure 1.19 Illustration of the first possibility in Lemma 1.60. The horizontal line indicates the preference list of f, in decreasing order from left to right. The markings below the line indicate workers in set $\mu_1(f)$ and the dashed markings above the line indicate workers in $\mu_2(f)$.

Definition 1.61. Let μ_1 and μ_2 be two stable matchings and f be a firm that is matched to capacity in both matchings. Then f prefers μ_1 to μ_2 if the first possibility in Lemma 1.60 holds and it prefers μ_2 to μ_1 otherwise. We will denote these as $\mu_1 \succ_f \mu_2$ and $\mu_2 \succ_f \mu_1$, respectively.

Using Definition 1.61, whose validity is based on Lemma 1.60, we obtain a partial order on the set of stable matchings for Setting III. Using the reduction stated above and facts from Section 1.4.1, this partially ordered set forms an FLD.

1.5 Linear Programming Formulation

The stable matching problem in Setting I admits a linear programming formulation in which the polyhedron defined by the constraints has integer optimal vertices, i.e., the vertices of this polyhedron are stable matchings. This yields an alternative way of computing a stable matching in polynomial time using well-known ways of solving linear programs (LPs). Linear programs for Settings II and III follow using the rural hospital theorem 1.23 and the reduction from Setting III to Setting II given in Section 1.2.3, respectively.

1.5.1 LP for Setting I

A sufficient condition for a worker–firm pair, (w,f) to not form a blocking pair with respect to a matching μ is that if w is matched to firm f' such that $f \succ_w f'$ then f should be matched to a worker w' such that $w' \succ_f w$. The fractional version of this condition appears in the third constraint of LP (1.1); see also Exercise 1.15. The first two constraints ensure that each worker and each firm are fully matched. Observe that the objective function in the LP given below is simply 0:

$$\begin{aligned}
\max \quad & 0 \\
\text{s.t.} \quad & \sum_{w} x_{wf} = 1 \quad \forall w \in W, \\
& \sum_{f} x_{wf} = 1 \quad \forall f \in F, \\
& \sum_{f \succ_w f'} x_{wf'} - \sum_{w' \succ_f w} x_{w'f} \le 0 \quad \forall w \in W,\ \forall f \in F, \\
& x_{wf} \ge 0 \quad \forall w \in W,\ \forall f \in F.
\end{aligned} \tag{1.1}$$

By the first two constraints, every integral feasible solution to LP (1.1) is a perfect matching on $W \cup F$ and, by the third constraint, it has no blocking pairs. It is therefore a stable matching.

Taking α_w, β_f, and γ_{wf} to be the dual variables for the first, second, and third constraints of LP (1.1), respectively, we obtain the dual LP:

$$\begin{aligned}
\min \quad & \sum_{w \in W} \alpha_w + \sum_{f \in F} \beta_f \\
\text{s.t.} \quad & \alpha_w + \beta_f + \sum_{f' \succ_w f} \gamma_{wf'} + \sum_{w \succ_f w'} \gamma_{w'f} \ge 1 \quad \forall w \in W,\ \forall f \in F, \\
& \gamma_{wf} \ge 0 \quad \forall w \in W,\ \forall f \in F.
\end{aligned} \tag{1.2}$$

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

Lemma 1.62. *If x is a feasible solution to LP (1.1) then $\alpha = 0,\ \beta = 0,\ \gamma = x$ is an optimal solution to LP (1.2). Furthermore, if $x_{wf} > 0$ then*

$$\sum_{f \succ_w f'} x_{wf'} = \sum_{w' \succ_f w} x_{w'f}.$$

Proof The feasibility of (α, β, γ) follows from the feasibility of x. The objective function value of this dual solution is 0, i.e., the same as that of the primal. Therefore, this solution is also optimal. For the solution since $\gamma = x$ is an optimal solution, if $x_{wf} > 0$ then $\gamma_{wf} > 0$. Now the desired equality follows by applying the complementary slackness condition to the third constraint of LP (1.1). □

Next, we will work towards proving Theorem 1.65 and Corollary 1.66 below. Let x be a feasible solution to LP (1.1). Corresponding to x, we will define $2n$ unit intervals, I_w and I_f, one corresponding to each worker w and one corresponding to each firm f, as follows. For each worker w, we have $\sum_{f \in F} x_{wf} = 1$. Order the firms according to w's preference list; for simplicity, assume it is $f_1 \succ_w f_2 \succ_w \cdots \succ_w f_n$. Partition I_w into n ordered subintervals such that the ith interval has length x_{wf_i}; if this quantity is zero, the length of the interval is also zero. Next, for each firm f, we have $\sum_w x_{wf} = 1$. Now, order the workers according to f's preference list but in reverse order; assume it is $w_1 \prec_f w_2 \prec_w \cdots \prec_w w_n$, and partition I_f into n ordered subintervals such that the ith interval has length $x_{w_i f}$.

Pick θ with uniform probability in the interval [0, 1] and, for each worker w, determine which subinterval of the interval I_w contains θ. The probability that any of these n subintervals is of zero length is zero, so we may assume that this event does not occur. Define a perfect matching μ_θ as follows: if the subinterval of I_w containing θ corresponds to firm f, then define $\mu_\theta(w) = f$.

Lemma 1.63. *If $\mu_\theta(w) = f$ then the subinterval of I_f containing θ corresponds to worker w.*

Proof Assume that the subinterval containing θ in I_w is $[a, b]$, where $a, b \in [0, 1]$; since it corresponding to f, $b - a = x_{wf}$. Since $x_{wf} > 0$, by the second part of Lemma 1.62 the subinterval corresponding to w in I_f is also $[a, b]$. Clearly, it contains θ. □

Lemma 1.64. *For each $\theta \in [0, 1]$, the perfect matching μ_θ is stable.*

Proof Assume that $\mu_\theta(w) \neq f$. Let $\mu_\theta(w) = f'$ and $\mu_\theta(f) = w'$, where $f \succ_w f'$. We will show that $w' \succ_f w$, thereby proving that (w,f) is not a blocking pair. Extending this conclusion to all worker–firm pairs that are not matched by μ_θ, we obtain that μ_θ is a stable matching.

Since $\mu_\theta(w) = f'$, θ lies in the subinterval corresponding to f' in I_w. In Figure 1.20, this subinterval is marked as $[a, b]$. The number θ also lies in the subinterval corresponding to w' in I_f; this subinterval is marked as $[c, d]$ in Figure 1.20. Let p_f and p_w denote the larger endpoints of the subintervals containing f and w in I_w and I_f, respectively. By the third constraint of LP (1.1), the subinterval $[p_w, 1]$ of I_f is at least as large as the subinterval $[p_f, 1]$ of I_w.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

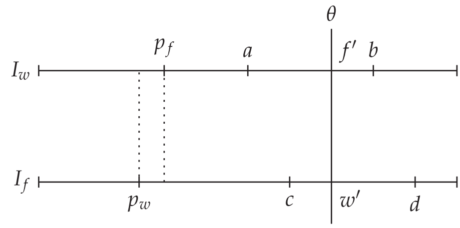


Figure 1.20

Since $f \succ_w f'$, θ lies in $[p_f, 1]$ and therefore it also lies in $[p_w, 1]$. Therefore, $w' \succ_f w$ and the lemma follows. □

Theorem 1.65. *Every feasible solution to LP (1.1) is a convex combination of stable matchings.*

Proof The feasible solution x of LP (1.1) has at most $O(n^2)$ positive variables x_{wf} and therefore there are $O(n^2)$ non-empty subintervals corresponding to these variables in the intervals I_w and I_f. Hence $[0, 1]$ can be partitioned into at most $O(n^2)$ non-empty subintervals, say $I_1, \ldots, I_k$, in such a way that none of these subintervals straddles the subintervals corresponding to the positive variables. For $1 \leq i \leq k$, each $\theta \in I_i$ corresponds to the same stable matching $\mu_i = \mu_\theta$. Let the length of I_i be α_i; clearly $\sum_{i=1}^{k} \alpha_i = 1$. Then we have

$$x = \sum_{i=1}^{k} \alpha_i \mu_i.$$

The theorem follows. □

Corollary 1.66. *The polyhedron defined by the constraints of LP (1.1) has integral optimal vertices; these are stable matchings.*

1.5.2 LPs for Settings II and III

For Setting II, by the rural hospital theorem 1.23, the sets of workers and firms matched in all stable matchings are the same. Let these sets be W' and F', respectively. Clearly, it suffices to work with the primal LP (1.1) restricted to these sets only.

For Setting III, we will use the reduction from Setting III to Setting II given in Section 1.2.3.

1.6 Exercises

Exercise 1.1 In Setting I, let M be a perfect matching which matches each worker $w \in W$ to a firm $f \in F$ such that $f \in R(w)$, i.e., f lies in the realm of possibilities of w. Prove or disprove that M is a stable matching.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

Exercise 1.2 In Setting I, we will say that a perfect matching μ is *Pareto optimal* if there is no perfect matching $\mu' \neq \mu$ under which each agent is weakly better off, i.e., for each agent $a \in W \cup F$, $\mu(a) \succeq_a \mu'(a)$. Prove that every stable matching is Pareto optimal. Is the converse true? Give a proof or a counterexample.

Exercise 1.3 Define a *preferred couple* in Setting I to be a worker–firm pair such that each is the most preferred in the other's preference list.

(a) Prove that if an instance has a preferred couple (w, f) then, in any stable matching, w and f must be matched to each other.
(b) Prove that if in an instance with n workers and n firms there are n disjoint preferred couples then there is a unique stable matching.
(c) Prove that the converse of the previous situation does not hold, i.e., there is an instance which does not have n disjoint preferred couples yet has a unique stable marriage.

Exercise 1.4 Construct an instance of Setting I that has exponentially many stable matchings.

Exercise 1.5 Design a polynomial time algorithm which, given an instance of Setting I, finds an unstable perfect matching, i.e., one having a blocking pair.

Exercise 1.6 In Setting II, let us add a second termination condition to the algorithm, namely terminate when every firm receives at least one proposal. If this condition holds, match every firm to its best proposal, leaving the rest of the workers unmatched. Give a counterexample to show that this matching is not stable.

Exercise 1.7 The following exercises are on the issue of incentive compatibility.

(a) Find an instance $(W, F, \succ)$ in which some worker is matched to the last firm with respect to its preference list in the worker-optimal stable matching. How many such workers can there be in a instance?
(b) For Setting I, find an instance in which a firm can improve its match in the worker-optimal stable matching by misreporting its preference list.
(c) For Setting I, find an instance in which two workers can collude, i.e., both work together, in such a way that one of them does better and the other does no worse than in the worker-optimal stable matching.

Exercise 1.8 Use the fact that the operations of min and max on two-element sets satisfy the distributive property to prove that the operations of meet and join for the lattice of stable matching also satisfy this property.

Exercise 1.9 [2] Prove that lattices arising from instances of stable matching form a *complete set of finite distributive lattices*, i.e., every FDL is isomorphic to some stable matching lattice.

Exercise 1.10 The following exercises are on rotations.

(a) Give an example of a stable matching μ such that $\text{next}(w) \neq \perp$ and yet w is not in a rotation with respect to μ.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

(b) Let ρ be a rotation with respect to the stable matching μ. Obtain ρ' by permuting the sequence of pairs in ρ. Show that if the permutation is a cyclic shift then $\rho'(\mu) = \rho(\mu)$ but otherwise the matching $\rho'(\mu)$ is not stable.

(c) Let ρ be a rotation with respect to the stable matching μ and let $\rho(\mu) = \mu'$. Prove that μ' is a direct successor of μ, i.e., $\mu \rhd \mu'$.

(d) Show that any path in G_μ from $\mu_\top$ to $\mu_\perp$ involves applying each rotation in $\mathcal{R}_\mu$ exactly once. Use this fact to give a polynomial time algorithm for finding all rotations in $\mathcal{R}_\mu$.

(e) Give a polynomial time algorithm for the following problem: given a rotation $\rho \in \mathcal{R}_\mu$, find the join-irreducible matching to which it corresponds.

Exercise 1.11 The following exercises are on Birkhoff's theorem.

(a) Give a polynomial time algorithm for computing the succinct description of lattice $\mathcal{L}_\mu$ promised in Corollary 1.53, i.e., π_μ.

(b) Let S be a lower set of π_μ and let S_ρ be the set of rotations corresponding to the join-irreducibles in S. Let τ be any topological sort of the rotations in S_ρ that is consistent with the partial order π_μ. Show that, if starting with the matching $\mu_\perp$, the inverses of the rotations in S_ρ are applied in the order given by τ then the matching obtained in $\mathcal{L}_\mu$ will be $f_\mu^{-1}(S)$. Use this fact to show that $f_\mu^{-1}(S)$ can be computed in polynomial time for any lower set S of π_μ.

Exercise 1.12 For any positive integer n, let S_n denote the set of divisors of n. Define a partial order $\pi_n = (S_n, \succeq)$ as follows: for $a, b \in S_n$, $a \succeq b$ if $b|a$. Prove that π_n is an FDL with the meet and join of two elements $a, b \in S_n$ being the gcd and lcm of a and b, respectively. Figure 1.21 shows the lattice for $n = 60$. Give a characterization of the join-irreducible matchings of π_{60} and find the projection of π_{60} onto the join-irreducible matchings of this lattice. Do the set of lattices $\{\pi_n \mid n \in \mathbb{Z}_+\}$ form a complete set of FDLs, as defined in Exercise 1.9? Prove or disprove your answer.

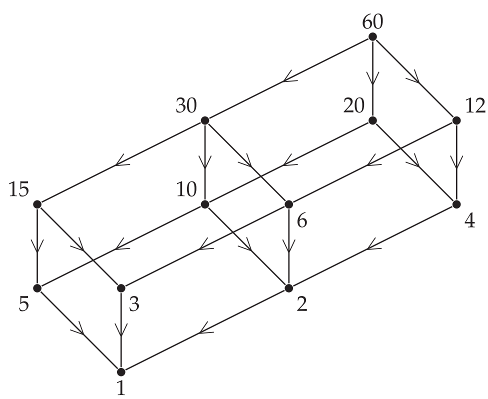


Figure 1.21 The lattice of divisors of 60.

Exercise 1.13 [6] For a stable matching μ in Setting I, define its *value* as follows. Assume $\mu(w) = f$, and that f is the jth firm on w's preference list and w is the kth worker on f's list; if so, define the value of the match (w, f) to be $k + j$. Define the value of μ to be the sum of the values of all matches in μ. Define an *equitable stable matching* to be one that minimizes the value. Give a polynomial time algorithm for finding such a matching.

Hint: Use the algorithms developed in Exercise 1.11. Also note that the problem of finding a minimum-weight lower set of π_μ is solvable in polynomial time, assuming that integer weights (positive, negative, or zero) are assigned to the elements of $\mathcal{J}_\mu$. The weight of a lower set S is defined to be the sum of weights of the elements in S.

Exercise 1.14 Prove Lemma 1.59 and use it to prove Lemma 1.60.

Hint: For the first part of Lemma 1.59, use a blocking pair argument, and, for the second part, first prove that

$$|W_1| + |W_2| = \sum_{f \in F_1} n_1(f) + \sum_{f \in F_2} n_2(f),$$

where $n_1(f) = |\mu_1(f) - \mu_2(f)|$ and $n_2(f) = |\mu_2(f) - \mu_1(f)|$.

Exercise 1.15 The LP (1.1) for the stable matching problem was derived from a sufficient condition, given in Section 1.5.1, which ensures that a worker–firm pair (w,f) does not form a blocking pair with respect to a matching μ. Another sufficient condition is that if f is matched to worker w' in such a way that $w \succ_f w'$ then w is matched to a firm f' such that $f' \succ_w f$. The fractional version of this condition is

$$\forall w \in W,\ \forall f \in F: \sum_{w \succ_f w'} x_{w'f} - \sum_{f' \succ_w f} x_{wf'} \le 0.$$

Show that this condition holds for any feasible solution x to LP (1.1).

Exercise 1.16 [13] Suppose that an instance of stable matching in Setting I has an odd number, k, of stable matchings. For each worker w, order its k matches, with multiplicity, according to its preference list and do the same for each firm f. Match w to the median element in its list. Let us call this the *median matching*. This exercise eventually helps to show that not only is this matching perfect but it is also stable. Moreover, it matches each firm to the median element in its list.

First, let $\mu_1, \ldots, \mu_l$ be any l stable matchings, not necessarily distinct, for an instance of stable matching in Setting I. For each worker–firm pair (w,f) let $n(w,f)$ be the number of these matchings in which w is matched to f and let $x_{wf} = (1/l)n(w,f)$.

Show that x is a feasible solution to LP (1.1), i.e., it is a fractional stable matching. For any k such that $1 \le k \le l$, let $\theta = (k/l) - \epsilon$, where $\epsilon > 0$ is smaller than $1/l$. Consider the stable matching μ_θ as defined by the procedure given in Section 1.5.1 for writing a fractional stable matching as a convex combination of stable matchings. Show that matching μ_θ matches each worker w to the kth firm in the ordered list of the l firms, not necessarily distinct, in which w is matched to under $\mu_1, \ldots, \mu_l$. Furthermore, show that μ_θ matches each firm f to the $(l - k + 1)$th worker in the ordered list of the l workers to which f is matched.

Using this fact, prove the assertions made above about median matching.

1.7 Bibliographic Notes

The seminal paper [5] introduced the stable matching problem and gave the deferred acceptance algorithm. For basic books on this problem and related topics see [4], [12], [8], and [9].

Theorem 1.32, proving that the worker-proposing DA algorithm is DSIC for workers, is due to Dubins and Freedman [3] (see also [10]). This ground-breaking result was instrumental in opening up the DA algorithm to highly consequential applications, such as school choice; for a discussion of the latter, see Chapter 8. Theorem 1.35, showing that there is no DSIC mechanism for firms in Setting III, is due to Roth [10].

John Conway proved that the set of stable matchings of an instance forms a finite distributive lattice; see [8]. The notion of rotation is due to Irving and Leather [6] and Theorem 1.56 is due to Birkhoff [1]. The LP formulation for stable matching was given by Vande Vate [14]; the proof given in Section 1.5.1 is due to Teo and Sethuraman [13].

References

[1] Birkhoff, Garrett. 1937. Rings of sets. *Duke Mathematical Journal*, **3**(3), 443–454.

[2] Blair, Charles. 1984. Every finite distributive lattice is a set of stable matchings. *Journal of Combinatorial Theory, Series A*, **37**(3), 353–356.

[3] Dubins, L. E., and Freedman, D. A. 1981. Machiavelli and the Gale–Shapley algorithm. *American Mathematical Monthly*, **88**(7), 485–494. www.jstor.org/stable/2321753.

[4] Gusfield, Dan, and Irving, Robert W. 1989. *The Stable Marriage Problem: Structure and Algorithm*. The MIT Press.

[5] Gale, David, and Shapley, Lloyd S. 1962. College admissions and the stability of marriage. *American Mathematical Monthly*, **69**(1), 9–15.

[6] Irving, Robert W., and Leather, Paul. 1986. The complexity of counting stable marriages. *SIAM Journal on Computing*, **15**(3), 655–667.

[7] Irving, Robert W., Leather, Paul, and Gusfield, Dan. 1987. An efficient algorithm for the "optimal" stable marriage. *Journal of the ACM*, **34**(3), 532–543.

[8] Knuth, Donald Ervin. 1997. *Stable Marriage and Its Relation to Other Combinatorial Problems: An Introduction to the Mathematical Analysis of Algorithms*. American Mathematical Society.

[9] Manlove, David. 2013. Algorithmics of Matching Under Preferences. World Scientific.

[10] Roth, A. E. 1982. The economics of matching: Stability and incentives. *Mathematics of Operations Research*, **7**(4), 617–628.

[11] Roth, A. E. 1985. The college admissions problem is not equivalent to the marriage problem. *Journal of Economic Theory*, **36**(2), 277–288. www.sciencedirect.com/science/article/pii/0022053185901061.

[12] Roth, Alvin E., and Sotomayor, Marilda. 1992. *Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis*. Econometric Society Monographs.

https://doi.org/10.1017/9781108937535.001 Published online by Cambridge University Press

[13] Teo, Chung-Piaw, and Sethuraman, Jay. 1998. The geometry of fractional stable matchings and its applications. *Mathematics of Operations Research*, **23**(4), 874–891.

[14] Vande Vate, John H. 1989. Linear programming brings marital bliss. *Operations Research Letters*, **8**(3), 147–153.