

# 10

## Matchings in Graphs

### 10.1 A Dancing Problem

At the prom, 300 students took part. They did not all know each other; in fact, every girl knew exactly 50 boys and every boy knew exactly 50 girls (we assume, as before, that acquaintance is mutual).

*We claim that the students can all dance simultaneously so that only pairs who know each other dance with each other.*

Since we are talking about acquaintances, it is natural to describe the situation by a graph (or at least, imagine the graph that describes it). So we draw 300 nodes, each representing a student, and connect two of them if they know each other. Actually, we can make the graph a little simpler: the fact that two boys, or two girls, know each other plays no role whatsoever in this problem: so we don't have to draw those edges that correspond to such acquaintances. We can then arrange the nodes, conveniently, so that the nodes representing boys are on the left, and nodes representing girls are on the right; then every edge will connect a node on the left to a node on the right. We shall denote the set of nodes on the left by  $A$ , the set of nodes on the right by  $B$ .

This way we obtain a special kind of graph, called a *bipartite graph*. Figure 10.1 shows such a graph (of course, depicting a smaller party). The thick edges show one way to pair up people for dancing. Such a set of edges is called a *perfect matching*.

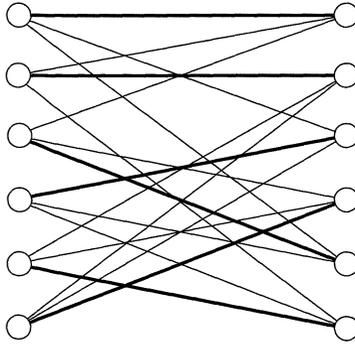


FIGURE 10.1. A bipartite graph with a perfect matching.

To be precise, let's give the definitions of these terms: A graph is *bipartite* if its nodes can be partitioned into two classes, say  $A$  and  $B$ , such that every edge connects a node in  $A$  to a node in  $B$ . A *perfect matching* is a set of edges such that every node is incident with exactly one of these edges.

After this, we can formulate our problem in the language of graph theory as follows: We have a bipartite graph with 300 nodes, in which every node has degree 50. We want to prove that it contains a perfect matching.

As before, it is good idea to generalize the assertion to any number of nodes. Let's be daring and guess that the numbers 300 and 50 play no role whatsoever. The only condition that matters is that all nodes have the same degree (and this is not 0). Thus we set out to prove the following theorem, named after the Hungarian mathematician D. König (who wrote the first book on graph theory).

**Theorem 10.1.1** *If every node of a bipartite graph has the same degree  $d \geq 1$ , then it contains a perfect matching.*

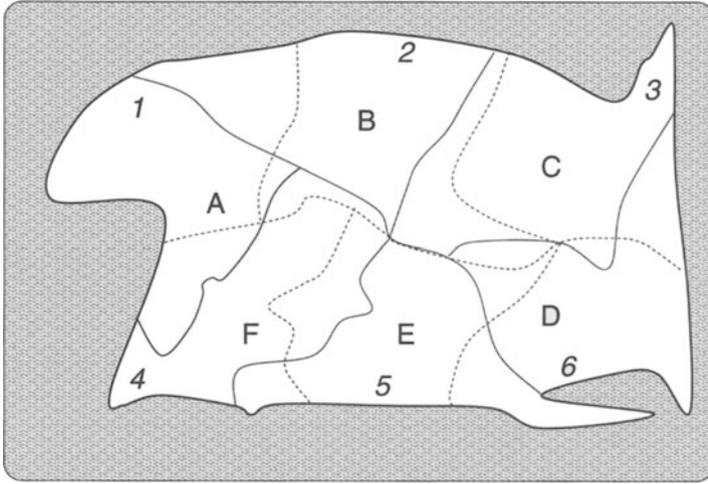
Before proving the theorem, it will be useful to solve some exercises, and then discuss another problem in the next section.

**10.1.1** It is obvious that for a bipartite graph to contain a perfect matching, it is necessary that  $|A| = |B|$ . Show that if every node has the same degree, then this is indeed so.

**10.1.2** Show by examples that the conditions formulated in the theorem cannot be dropped:

- (a) A nonbipartite graph in which every node has the same degree need not contain a perfect matching.
- (b) A bipartite graph in which every node has positive degree (but not all the same) need not contain a perfect matching.

**10.1.3** Prove Theorem 10.1.1 for  $d = 1$  and  $d = 2$ .



A, B... areas of tribes                      ————— border between tribes  
 1, 2.. areas of tortoises                  - - - - - border between tortoises

FIGURE 10.2. Six tribes and six tortoises on an island

## 10.2 Another matching problem

An island is inhabited by six tribes. They are on good terms and split up the island between them, so that each tribe has a hunting territory of 100 square miles. The whole island has an area of 600 square miles.

The tribes decide that they all should choose new totems. They decide that each tribe should pick one of the six species of tortoise that live on the island. Of course, they want to pick different totems, and in such a way that the totem of each tribe should occur somewhere on their territory.

It is given that the territories where the different species of tortoises live don't overlap, and they have the same area, 100 square miles (so it also follows that every part of the island is inhabited by some kind of tortoise). Of course, the way the tortoises divide up the island may be entirely different from the way the tribes do (Figure 10.2)

We want to prove that such a selection of totems is always possible.

To see the significance of the conditions, let's assume that we did not stipulate that the area of each tortoise species is the same. Then some species could occupy more, say, 200 square miles. But then it could happen that two of tribes are living on exactly these 200 square miles, and so their only possible choice for a totem would be one and the same species.

Let's try to illustrate our problem by a graph. We can represent each tribe by a node, and also each species of tortoise by a node. Let us connect a tribe-node to a tortoise-node if the species occurs somewhere on the territory of the tribe (we could also say that the tribe occurs on the territory of the

species, just in case the tortoises want to pick totems too). Drawing the tribe-nodes on the left and the tortoise-nodes on the right makes it clear that we get a bipartite graph (Figure 10.3). And what is it that we want to prove? It is that this graph has a perfect matching!

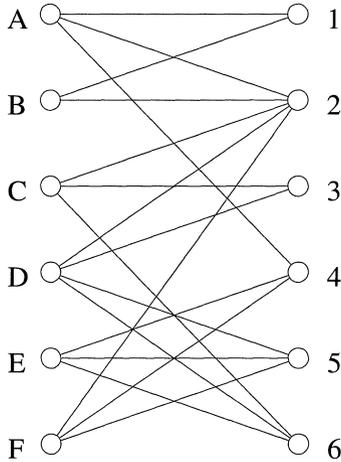


FIGURE 10.3. The graph of tribes and tortoises

So this is very similar to the problem discussed (but not solved!) in the previous section: We want to prove that a certain bipartite graph has a perfect matching. Theorem 10.1.1 says that for this conclusion it suffices to know that every node has the same degree. But this is too strong a condition; it is not at all fulfilled in our example (tribe B has only two tortoises to choose from, while tribe D has four).

So what property of this graph should guarantee that a perfect matching exists? Turning this question around: What would *exclude* a perfect matching?

For example, it would be bad if a tribe could not find any tortoises on its own territory. In the graph, this would correspond to a node with degree 0. Now this is not a danger, since we know that tortoises occur everywhere on the island.

It would also be bad (and this has come up already) if two tribes could only choose one and the same tortoise. But then this tortoise would have an area of at least 200 square miles, which is not the case. A somewhat more subtle sort of trouble would arise if three tribes had only two tortoises on their combined territory. But this, too, is impossible: The two species of tortoises would cover an area of at least 300 square miles, so one of them would have to cover more than 100. More generally, we can see that the combined territory of any  $k$  tribes holds at least  $k$  species of tortoises. In terms of the graph, this means that for any  $k$  nodes on the left, there are

at least  $k$  nodes on the right connected to at least one of them. We'll see in the next section that this is all we need to observe about this graph.

## 10.3 The Main Theorem

Now we state and prove a fundamental theorem about perfect matchings. This will complete the solution of the problem about tribes and tortoises, and (with some additional work) of the problem about dancing at the prom (and some problems further down the road from the prom, as its name shows).

**Theorem 10.3.1 (The Marriage Theorem)** *A bipartite graph has a perfect matching if and only if  $|A| = |B|$  and for any subset of (say)  $k$  nodes of  $A$  there are at least  $k$  nodes of  $B$  that are connected to at least one of them.*

This important theorem has many variations; some of these occur in the exercises. These were discovered by the German mathematician G. Frobenius, by the Hungarian D. König, the American P. Hall, and others.

Before proving this theorem, let us discuss one more question. If we interchange “left” and “right,” perfect matchings remain perfect matchings. But what happens to the condition stated in the theorem? It is easy to see that it remains valid (as it should). To see this, we have to argue that if we pick any set  $S$  of  $k$  nodes in  $B$ , then they are connected to at least  $k$  nodes in  $A$ . Let  $n = |A| = |B|$  and let us color the nodes in  $A$  connected to nodes in  $S$  black, the other nodes white (Figure 10.4). Then the white nodes are connected to at most  $n - k$  nodes (since they are not connected to any node in  $S$ ). Since the condition holds “from left to right,” the number of white nodes is at most  $n - k$ . But then the number of black nodes is at least  $k$ , which proves that the condition also holds “from right to left.”

**Proof.** Now we can turn to the proof of Theorem 10.3.1. We shall have to refer to the condition given in the theorem so often that it will be convenient to call graphs satisfying this conditions “good” (just for the duration of this proof). Thus a bipartite graph is “good” if it has the same number of nodes left and right, and any  $k$  “left” nodes are connected to at least  $k$  “right” nodes.

It is obvious that every graph with a perfect matching is “good,” so what we need to prove is the converse: *Every “good” graph contains a perfect matching.* For a graph on just two nodes, being “good” means that these two nodes are connected. Thus for a graph to have a perfect matching means that it can be partitioned into “good” graphs with 2 nodes. (To partition a graph means that we divide the nodes into classes, and keep an edge between two nodes only if they are in the same class.)

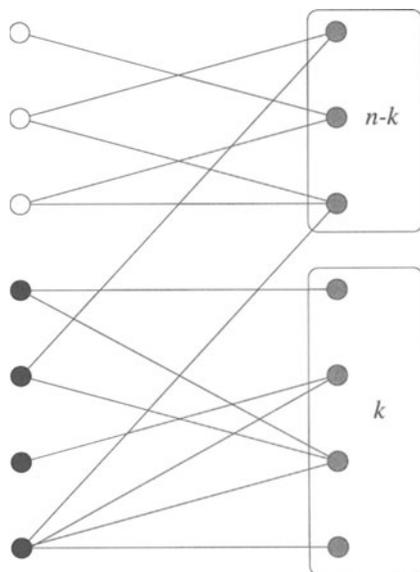


FIGURE 10.4. The good graph is also good from right to left.

Now our plan is to partition our graph into two “good” parts, then partition each of these into two “good” parts, etc., until we get “good” parts with 2 nodes. Then the edges that remain form a perfect matching. To carry out this plan, it suffices to prove that

*if a “good” bipartite graph has more than 2 nodes, then it can be partitioned into two good bipartite graphs.*

Let us try a very simple partition first: Select nodes  $a \in A$  and  $b \in B$  that are connected by an edge; let these two nodes be the first part, and the remaining nodes the other. There is no problem with the first part: it is “good.” But the second part may not be good: It can have some set  $S$  of  $k$  nodes on the left connected to fewer than  $k$  nodes on the right (Figure 10.5). In the original graph, these  $k$  nodes were connected to at least  $k$  nodes in  $B$ ; this can hold only if the  $k$ th such node was the node  $b$ . Let  $T$  denote the set of neighbors of  $S$  in the original graph. What is important to remember is that  $|S| = |T|$ .

Now we try another way of partitioning the graph: We take  $S \cup T$  (together with the edges between them) as one part, and the rest of the nodes as the other. (This rest is not empty: The node  $a$  belongs to it, for example.)

Let’s argue that both these parts are “good.” Take the first graph first. Take any subset of, say,  $j$  nodes in  $S$  (the left-hand side of the first graph). Since the original graph was good, they are connected to at least  $j$  nodes, which are all in  $T$  by the definition of  $T$ .

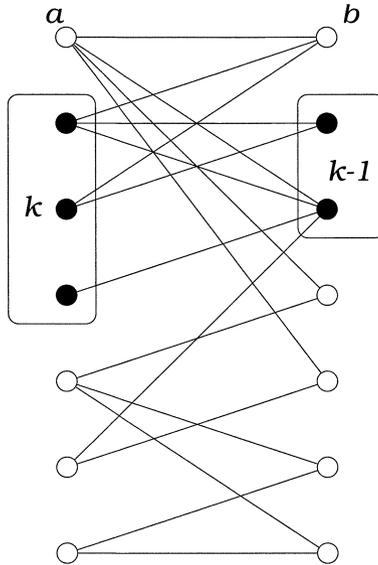


FIGURE 10.5. Goodness lost when two nodes are removed

For the second graph, it follows similarly that it is good if we interchange “left” and “right.” This completes the proof.  $\square$

We still have to prove Theorem 10.1.1. This is now quite easy and is left to the reader as Exercise 10.3.1.

**10.3.1** Prove that if in a bipartite graph every node has the same degree  $d \neq 0$ , then the bipartite graph is “good” (and hence contains a perfect matching; this proves Theorem 10.1.1).

**10.3.2** Suppose that in a bipartite graph, for any subset  $X$  of nodes of  $A$  there are at least  $|X|$  nodes in  $B$  that are connected to one of them (but in contrast to Theorem 10.3.1, we don’t assume that  $|A| = |B|$ ). Prove that there is a set of edges that match every node of  $A$  with a node of  $B$ , where different nodes of  $A$  are matched with different nodes of  $B$  (but some nodes of  $B$  may remain unmatched).

## 10.4 How to Find a Perfect Matching

We have a condition for the existence of a perfect matching in a graph that is *necessary and sufficient*. Does this condition settle this issue once and for all? To be more precise: Suppose that somebody gives us a bipartite graph; what is a good way to *decide* whether it contains a perfect matching? And how do we *find* a perfect matching if there is one?

We may assume that  $|A| = |B|$  (where, as before,  $A$  is the set of nodes on the left and  $B$  is the set of nodes on the right). This is easy to check, and if it fails, then it is obvious that no perfect matching exists, and we have nothing else to do.

One thing we can try is to look at all subsets of the edges, and see whether any of these is a perfect matching. It is easy enough to do so; but there are terribly many subsets to check! Say, in our introductory example, we have 300 nodes, so  $|A| = |B| = 150$ ; every node has degree 50, so the number of edges is  $150 \cdot 50 = 7500$ ; the number of subsets of a set of this size is  $2^{7500} > 10^{2257}$ , a number that is more than astronomical.

We can do a little bit better if instead of checking all subsets of the edges, we look at all possible ways to pair up elements of  $A$  with elements of  $B$ , and check whether any of these pairings matches only nodes that are connected to each other by an edge. Now the number of ways to pair up the nodes is “only”  $150! \approx 10^{263}$ . Still hopeless.

Can we use Theorem 10.3.1? To check that the necessary and sufficient condition for the existence of a perfect matching is satisfied, we have to look at every subset  $S$  of  $A$ , and see whether the number of its neighbors in  $B$  is at least as large as  $S$  itself. Since the set  $A$  has  $2^{150} \approx 10^{45}$  subsets, this takes a much smaller number of cases to check than any of the previous possibilities, but still astronomical!

So Theorem 10.3.1 does not really help too much in deciding whether a given graph has a perfect matching. We have seen that it does help in *proving* that certain properties of a graph imply that the graph has a perfect matching. We’ll come back to this theorem later and discuss its significance. Right now, we have to find some other way to deal with our problem.

Let us introduce one more expression: By a *matching* we mean a set of edges that have no endpoint in common. A perfect matching is the special case when, in addition, the edges cover all the nodes. But a matching can be much smaller: the empty set, or any edge by itself, is a matching.

Let’s try to construct a perfect matching in our graph by starting with the empty set and building up a matching one by one. So we select two nodes that are connected, and mark the edge between them; then we select two other nodes that are connected, and mark the edge between them etc. we can do this until no two unmatched nodes are connected by an edge. The edges we have marked form a matching  $M$ . This is often called the *greedy matching*, since it is constructed greedily, without consideration for the future consequences of our choice. If we are lucky, then the greedy matching is perfect, and we have nothing else to do. But what do we do if  $M$  is not perfect? Can we conclude that the graph has no perfect matching at all? No, we cannot; it may happen that the graph has a perfect matching, but we made some unlucky choices when selecting the edges of  $M$ .

**10.4.1** Show by an example that it may happen that a bipartite graph  $G$  has a perfect matching, but if we are unlucky, the greedy matching  $M$  constructed above is not perfect.

**10.4.2** Prove that if  $G$  has a perfect matching, then every greedy matching matches up at least half of the nodes.

So suppose that we have constructed a matching  $M$  that is not perfect. We have to try to increase its size by “backtracking,” i.e., by deleting some of its edges and replacing them by more edges. But how do we find the edges we want to replace?

The trick is the following. We look for a path  $P$  in  $G$  of the following type:  $P$  starts and ends at nodes  $u$  and  $v$  that are unmatched by  $M$ ; and every second edge of  $P$  belongs to  $M$  (Figure 10.6). Such a path is called an *augmenting path*. It is clear that an augmenting path  $P$  contains an odd number of edges, and in fact, the number of its edges not in  $M$  is one larger than the number of its edges in  $M$ .

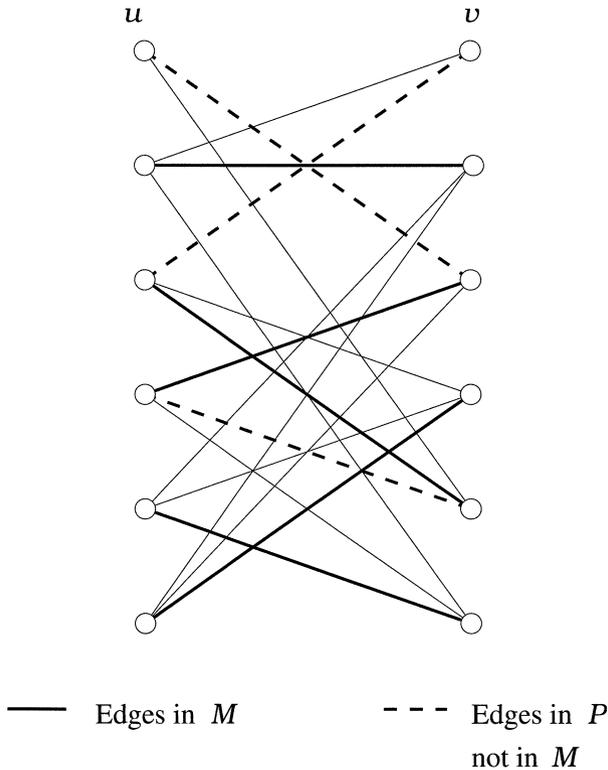


FIGURE 10.6. An augmenting path in a bipartite graph.

If we find an augmenting path  $P$ , we can delete those edges of  $P$  that are in  $M$  and replace them by those edges of  $P$  that are not in  $M$ . It is clear that this results in a matching  $M'$  that is larger than  $M$  by one edge. (The fact that  $M'$  is a matching follows from the observation that the remaining edges of  $M$  cannot contain any node of  $P$ : The two endpoints of  $P$  were supposed to be unmatched, while the interior nodes of  $P$  were matched by edges of  $M$  that we deleted.) So we can repeat this until we get either a perfect matching or a matching  $M$  for which no augmenting path exists.

So we have two questions to answer: how do we find an augmenting path if it exists? And if it does not exist, does this mean that there is no perfect matching at all? It will turn out that an answer to the first question will also imply the (affirmative) answer to the second.

Let  $U$  be the set of unmatched nodes in  $A$  and let  $W$  be the set of unmatched nodes in  $B$ . As we noted, any augmenting path must have an odd number of edges, and hence it must connect a node in  $U$  to a node in  $W$ . Let us try to find such an augmenting path starting from some node in  $U$ . Let's say that a path  $Q$  is *almost augmenting* if it starts at a node in  $U$ , ends at a node in  $A$ , and every second edge of it belongs to  $M$ . An almost augmenting path must have an even number of edges, and must end with an edge of  $M$ .

What we want to do is to find the set of nodes in  $A$  that can be reached on an almost augmenting path. Let's agree that we consider a node in  $U$  to be an almost augmenting path in itself (of length 0); then we know that every node in  $U$  has this property. Starting with  $S = U$ , we build up a set  $S$  gradually. At any stage, the set  $S$  will consist of nodes we already know are reachable by some almost augmenting path. We denote by  $T$  the set of nodes in  $B$  that are matched with nodes in  $S$  (Figure 10.7). Since the nodes of  $U$  have nothing matched with them and they are all in  $S$ , we have

$$|S| = |T| + |U|.$$

We look for an edge that connects a node  $s \in S$  to some node  $r \in B$  that is *not* in  $T$ . Let  $Q$  be an almost augmenting path starting at some node  $u \in U$  and ending at  $s$ . Now there are two cases to consider:

- If  $r$  is unmatched (which means that it belongs to  $W$ ), then by appending the edge  $sr$  to  $Q$  we get an augmenting path  $P$ . So we can increase the size of  $M$  (and forget about  $S$  and  $T$ ).
- If  $r$  is matched with a node  $q \in A$ , then we can append the edges  $sr$  and  $rq$  to  $Q$  to get an almost augmenting path from  $U$  to  $q$ . So we can add  $q$  to  $S$ .

So if we find an edge connecting a node in  $S$  to a node not in  $T$ , we can increase either the size of  $M$  or the set  $S$  (and leave  $M$  as it was). Sooner or later we must encounter a situation where either  $M$  is a perfect matching

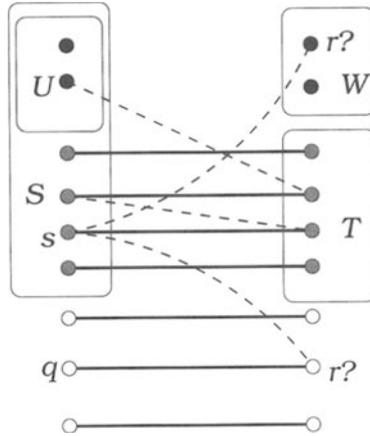


FIGURE 10.7. Reaching nodes by almost augmenting paths. Only edges on these paths, and of  $M$ , are shown.

(and we are done), or  $M$  is not perfect, but no edge connects  $S$  to any node outside  $T$ .

So what are we to do in this case? Nothing! If this occurs, we can conclude that there is no perfect matching at all. In fact, all neighbors of the set  $S$  are in  $T$ , and  $|T| = |S| - |U| < |S|$ . We know that this implies that there is no perfect matching at all in the graph.

Figure 10.8 shows how this algorithm finds a matching in the bipartite graph that is a subgraph of the “grid.”

To sum up, we do the following. At any point in time, we will have a matching  $M$  and a set  $S$  of nodes in  $A$  that we know can be reached on almost augmenting paths. If we find an edge connecting  $S$  to a node not matched with any node in  $S$ , we can either increase the size of  $M$  or the set  $S$ , and repeat. If no such edge exists, then either  $M$  is perfect or no perfect matching exists at all.

**Remark.** In this chapter we restricted our attention to matchings in bipartite graphs. One can, of course, define matchings in general (nonbipartite) graphs. It turns out that both the necessary and sufficient condition given in Theorem 10.3.1 and the algorithm described in this section can be extended to nonbipartite graphs. However, this requires methods that are quite a bit more involved, which lie beyond the scope of this book.

**10.4.3** Follow how the algorithm works on the graph in Figure 10.9.

**10.4.4** Show how the description of the algorithm above contains a new proof of Theorem 10.3.1.

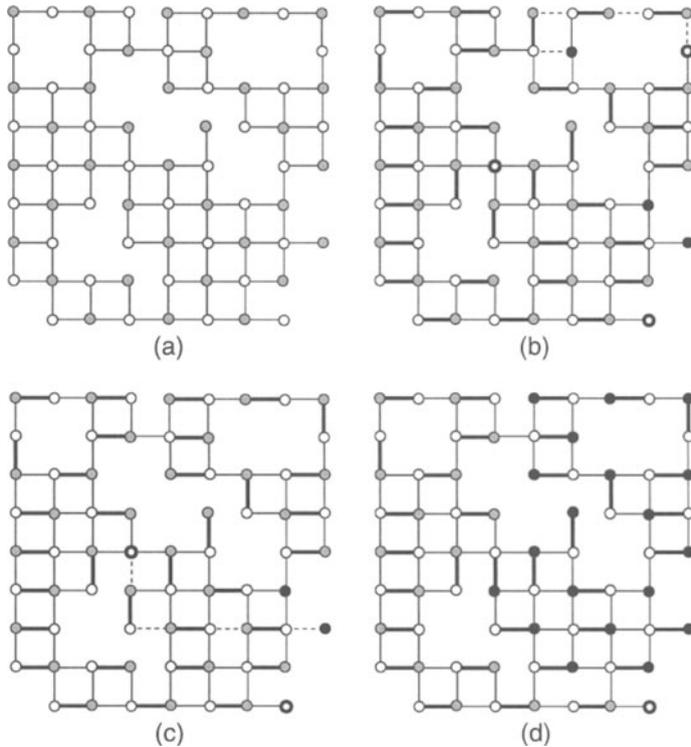


FIGURE 10.8. (a) The graph in which we want to find a perfect matching. (b) Pick a starting matching, and mark the unmatched nodes. There are 3 black and 3 white unmatched nodes. Broken lines indicate an augmenting path. (c) The new matching and the unmatched nodes after augmentation. Broken lines indicate a new augmenting path (much longer this time). (d) The final situation: Nodes that are accessible on almost augmenting paths are marked black. They have fewer neighbors than their number, so the matching is maximum.

## Review Exercises

**10.4.5** Is there a bipartite graph with degrees 3, 3, 3, 3, 3, 3, 3, 3, 3, 5, 6, 6? (These can be distributed in the two classes of nodes arbitrarily.)

**10.4.6** A bipartite graph has 16 nodes of degree 5, and some nodes of degree 8. We know that all degree-8 nodes are on the left hand side. How many degree 8 nodes can the graph have?

**10.4.7** Let  $G$  be a bipartite graph with the same number of nodes on both sides. Suppose that every nonempty subset  $A$  on the left has at least  $|A| + 1$  neighbors on the right. Prove that each edge of  $G$  can be extended to a perfect matching of  $G$ .

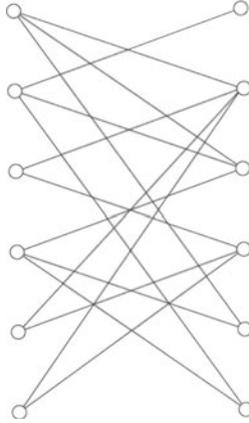


FIGURE 10.9. A graph for trying out the algorithm.

**10.4.8** Now suppose that we have the weaker condition that every nonempty subset  $A$  on the left has at least  $|A| - 1$  neighbors on the right. Prove that  $G$  contains a matching that matches up all but one node on each side.

**10.4.9** Let  $G$  be a bipartite graph with  $m$  nodes on both sides. Prove that if each node has degree larger than  $m/2$ , then it has a perfect matching.

**10.4.10** Does the graph in Figure 10.10 have a perfect matching?

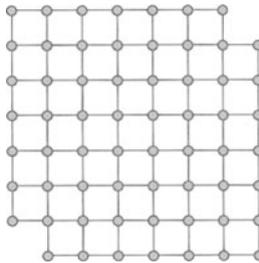


FIGURE 10.10. A truncated chessboard.

**10.4.11** Draw a graph whose nodes are the subsets of  $\{a, b, c\}$ , and for which two nodes are adjacent if and only if they are subsets that differ in exactly one element.

(a) What is the number of edges and nodes in this graph? Can you name this graph?

(b) Is this graph connected? Does it have a perfect matching? Does it have a Hamilton cycle?

**10.4.12** Draw a graph whose nodes are the 2-subsets of  $\{a, b, c, d, e\}$ , and two nodes are adjacent if and only if they are disjoint subsets.

- (a) Show that you get the Petersen graph (Figure 7.13).
- (b) How many perfect matchings does the Petersen graph have?

**10.4.13** (a) How many perfect matchings does a path on  $n$  nodes have? (b) How many matchings (not necessarily perfect) does a path on  $n$  nodes have? [Find a recurrence first.] (c) How many matchings does a cycle on  $n$  nodes have?

**10.4.14** Which 2-regular bipartite graph with  $n$  nodes has the largest number of perfect matchings?

**10.4.15** How many perfect matchings does the “ladder” with  $2n$  nodes (Figure 10.11) have?

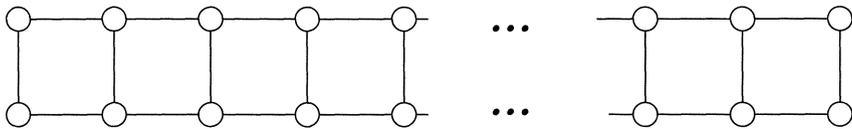


FIGURE 10.11. The ladder graph.