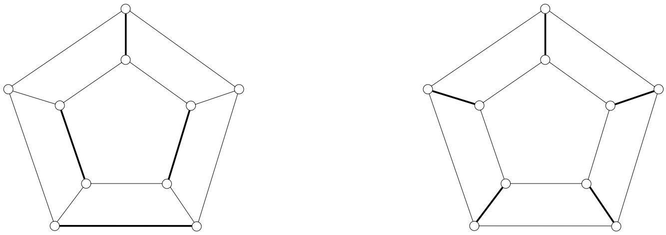
16

Matchings

Contents

16.1 Maximum Matchings

A *matching* in a graph is a set of pairwise nonadjacent links. If M is a matching, the two ends of each edge of M are said to be *matched* under M, and each vertex incident with an edge of M is said to be *covered* by M. A *perfect matching* is one which covers every vertex of the graph, a *maximum matching* one which covers as many vertices as possible. A graph is *matchable* if it has a perfect matching. Not all graphs are matchable. Indeed, no graph of odd order can have a perfect matching, because every matching clearly covers an even number of vertices. Recall that the number of edges in a maximum matching of a graph G is called the *matching number* of G and denoted $\alpha'(G)$. A *maximal matching* is one which cannot be extended to a larger matching. Equivalently, it is one which may be obtained by choosing edges in a greedy fashion until no further edge can be incorporated. Such a matching is not necessarily a maximum matching. Examples of maximal and perfect matchings in the pentagonal prism are indicated in Figures 16.1 and 16.1b, respectively.



(a) (b)

Fig. 16.1. (a) A maximal matching, (b) a perfect matching

The main question we address in this chapter is:

Problem 16.1 THE MAXIMUM MATCHING PROBLEM
GIVEN: *a graph G,*
FIND: *a maximum matching M^* in G.*

There are many questions of practical interest which, when translated into the language of graph theory, amount to finding a maximum matching in a graph. One such is:

Problem 16.2 THE ASSIGNMENT PROBLEM
A certain number of jobs are available to be filled. Given a group of applicants for these jobs, fill as many of them as possible, assigning applicants only to jobs for which they are qualified.

This situation can be represented by means of a bipartite graph $G[X, Y]$ in which X represents the set of applicants, Y the set of jobs, and an edge xy with $x \in X$ and $y \in Y$ signifies that applicant x is qualified for job y. An assignment of applicants to jobs, one person per job, corresponds to a matching in G, and the problem of filling as many vacancies as possible amounts to finding a maximum matching in G.

As we show in Section 16.5, the Assignment Problem can be solved in polynomial time. Indeed, we present there a polynomial-time algorithm for finding a maximum matching in an arbitrary graph. The notions of alternating and augmenting paths with respect to a given matching, defined below, play an essential role in these algorithms.

Augmenting Paths

Let M be a matching in a graph G. An M*-alternating path* or *cycle* in G is a path or cycle whose edges are alternately in M and $E \setminus M$. An M-alternating path might or might not start or end with edges of M (see Figure 16.2).

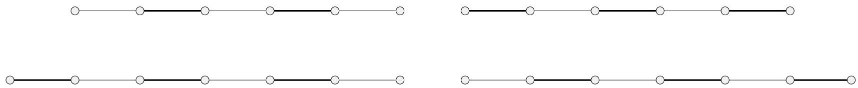
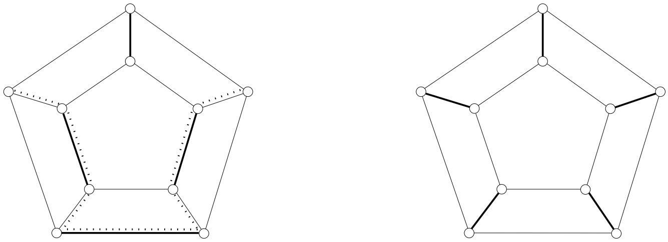


Fig. 16.2. Types of M-alternating paths

If neither its origin nor its terminus is covered by M (as in the top-left path in Figure 16.2) the path is called an M*-augmenting path*. Figure 16.3a shows an M-augmenting path in the pentagonal prism, where M is the matching indicated in Figure 16.1a.



(a) (b)

Fig. 16.3. (a) An M-augmenting path P, (b) the matching $M \triangle E(P)$

Berge's Theorem

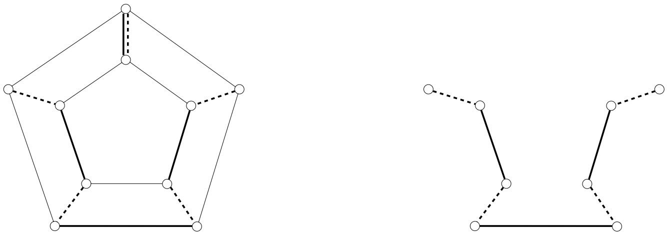
The following theorem, due to Berge (1957), points out the relevance of augmenting paths to the study of maximum matchings.

Theorem 16.3 BERGE'S THEOREM
A matching M in a graph G is a maximum matching if and only if G contains no M-augmenting path.

Proof Let M be a matching in G. Suppose that G contains an M-augmenting path P. Then $M' := M \triangle E(P)$ is a matching in G, and $|M'| = |M| + 1$ (see Figure 16.3). Thus M is not a maximum matching.

Conversely, suppose that M is not a maximum matching, and let M^* be a maximum matching in G, so that $|M^*| > |M|$. Set $H := G[M \triangle M^*]$, as illustrated in Figure 16.4.

Each vertex of H has degree one or two in H, for it can be incident with at most one edge of M and one edge of M^*. Consequently, each component of H is either an even cycle with edges alternately in M and M^*, or else a path with edges alternately in M and M^*.



(a) (b)

Fig. 16.4. (a) Matchings M (heavy) and M^* (broken), and (b) the subgraph $H := G[M \triangle M^*]$

Because $|M^*| > |M|$, the subgraph H contains more edges of M^* than of M, and therefore some path-component P of H must start and end with edges of M^*. The origin and terminus of P, being covered by M^*, are not covered by M. The path P is thus an M-augmenting path in G. □

Exercises

16.1.1

a) Show that the Petersen graph has exactly six perfect matchings.
b) Determine $pm(K_{2n})$ and $pm(K_{n,n})$, where $pm(G)$ denotes the number of perfect matchings in graph G.

16.1.2 Show that it is impossible, using 1×2 rectangles (dominoes), to tile an 8×8 square (chessboard) from which two opposite 1×1 corner squares have been removed.

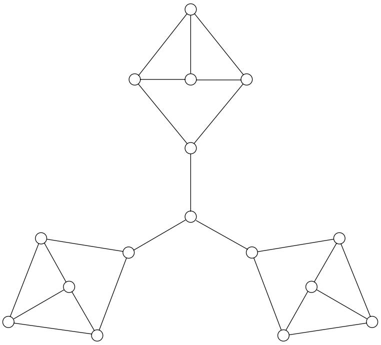


Fig. 16.5. The Sylvester graph: a 3-regular graph with no perfect matching

16.1.3 Show that if G is triangle-free, then $\alpha'(G) = n - \chi(\overline{G})$.

16.1.4 Find a maximal matching M and a perfect matching M^* in the pentagonal prism such that the subgraph induced by $M \triangle M^*$ has two components, one a cycle and the other an M-augmenting path.

$\star$**16.1.5**

a) Let M and M' be maximum matchings of a graph G. Describe the structure of the subgraph $H := G[M \triangle M']$.
b) Let M and M' be perfect matchings of a graph G. Describe the structure of the subgraph $H := G[M \triangle M']$.
c) Deduce from (b) that a tree has at most one perfect matching.

16.1.6 Let M and N be matchings of a graph G, where $|M| > |N|$. Show that there are disjoint matchings M' and N' of G such that $|M'| = |M| - 1, |N'| = |N| + 1$ and $M' \cup N' = M \cup N$.

$\star$**16.1.7**

a) Let M be a perfect matching in a graph G and S a subset of V. Show that $|M \cap \partial(S)| \equiv |S| \pmod 2$.
b) Deduce that if M is a perfect matching of the Petersen graph, and C is the edge set of one of its 5-cycles, then $|M \cap C|$ is even.

16.1.8

a) Let M be a perfect matching in a graph G, all of whose vertices are of odd degree. Show that M includes every cut edge of G.
b) Deduce that the 3-regular graph of Figure 16.5 has no perfect matching.
c) For each $k \geq 2$, find a $(2k+1)$-regular simple graph with no perfect matching.

16.1.9 Let M be a maximal matching in a graph G, and let M^* be a maximum matching in G. Show that $|M| \geq \frac{1}{2}|M^*|$.

16.1.10 Consider a complete graph K on $2n$ vertices embedded in the plane, with n vertices coloured red, n vertices coloured blue, and each edge a straight-line segment. Show that K has a perfect matching whose edges do not cross, with each edge joining a red vertex and a blue vertex.

16.1.11 The game of *Slither* is played as follows. Two players alternately select distinct vertices $v_0, v_1, v_2, \ldots$ of a graph G, where, for $i \geq 0$, v_{i+1} is required to be adjacent to v_i. The last player able to select a vertex wins the game. Show that the first player has a winning strategy if and only if G has no perfect matching.

(W.N. Anderson, Jr.)

16.1.12 Let G be a simple graph with $n \geq 2\delta$. Show that $\alpha' \geq \delta$.

16.1.13 Let G be a nonempty graph which has a unique perfect matching M.

a) Show that G has no M-alternating cycle, and that the first and last edges of every M-alternating path belong to M.
b) Deduce that if $G := G[X, Y]$ is bipartite, then X and Y each contain a vertex of degree one.
c) Give an example of a graph with a unique perfect matching and no vertex of degree one.

16.1.14

a) Let M be a matching in a graph G. Show that there is a maximum matching in G which covers every vertex covered by M.
b) Deduce that every vertex of a connected nontrivial graph is covered by some maximum matching.
c) Let $G[X, Y]$ be a bipartite graph and let $A \subseteq X$ and $B \subseteq Y$. Suppose that G has a matching which covers every vertex in A and also one which covers every vertex in B. Show that G has a matching which covers every vertex in $A \cup B$. (L. Dulmage and N.S. Mendelsohn)

$\star$**16.1.15** Essential Vertex

A vertex v of a graph G is *essential* if v is covered by every maximum matching in G, that is, if $\alpha'(G - v) = \alpha'(G) - 1$.

a) Describe an infinite family of connected graphs which contain no essential vertices.
b) Show that every nonempty bipartite graph has an essential vertex.

(D. de Caen)

16.1.16 A factory has n jobs $1, 2, \ldots, n$, to be processed, each requiring one day of processing time. There are two machines available. One can handle one job at a time and process it in one day, whereas the other can process two jobs simultaneously and complete them both in one day. The jobs are subject to precedence constraints represented by a binary relation $\prec$, where $i \prec j$ signifies that job i

must be completed before job j is started. The objective is to complete all the jobs while minimizing $d_1 + d_2$, where d_i is the number of days during which machine i is in use. Formulate this problem as one of finding a maximum matching in a suitably defined graph. (M. Fujii, T. Kasami, and N. Ninomiya)

16.2 Matchings in Bipartite Graphs

Hall's Theorem

In many applications, one wishes to find a matching in a bipartite graph $G[X, Y]$ which covers every vertex in X. Necessary and sufficient conditions for the existence of such a matching were first given by Hall (1935). Recall that if S is a set of vertices in a graph G, the set of all neighbours of the vertices in S is denoted by $N(S)$.

Theorem 16.4 Hall's Theorem
A bipartite graph $G := G[X, Y]$ has a matching which covers every vertex in X if and only if

$$|N(S)| \geq |S| \quad \textit{for all} \quad S \subseteq X \tag{16.1}$$

Proof Let $G := G[X, Y]$ be a bipartite graph which has a matching M covering every vertex in X. Consider a subset S of X. The vertices in S are matched under M with distinct vertices in $N(S)$. Therefore $|N(S)| \geq |S|$, and (16.1) holds.

Conversely, let $G := G[X, Y]$ be a bipartite graph which has no matching covering every vertex in X. Let M^* be a maximum matching in G and u a vertex in X not covered by M^*. Denote by Z the set of all vertices reachable from u by M^*-alternating paths. Because M^* is a maximum matching, it follows from Theorem 16.3 that u is the only vertex in Z not covered by M^*. Set $R := X \cap Z$ and $B := Y \cap Z$ (see Figure 16.6).

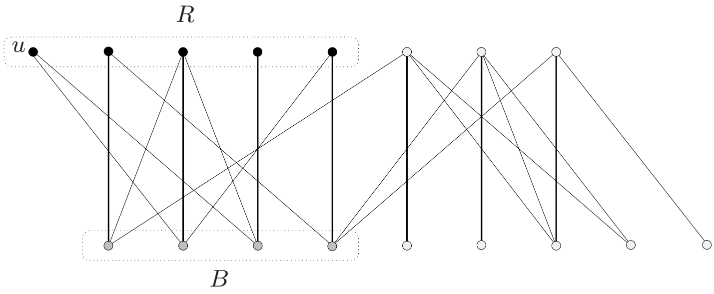


Fig. 16.6. Proof of Hall's Theorem (16.4)

Clearly the vertices of $R \setminus \{u\}$ are matched under M^* with the vertices of B. Therefore $|B| = |R| - 1$ and $N(R) \supseteq B$. In fact $N(R) = B$, because every vertex in $N(R)$ is connected to u by an M^*-alternating path. These two equations imply that

$$|N(R)| = |B| = |R| - 1$$

Thus Hall's condition (16.1) fails for the set $S := R$. $\square$

Theorem 16.4 is also known as the *Marriage Theorem*, because it can be restated more picturesquely as follows: if every group of girls in a village collectively like at least as many boys as there are girls in the group, then each girl can marry a boy she likes.

Hall's Theorem has proved to be a valuable tool both in graph theory and in other areas of mathematics. It has several equivalent formulations, including the following one in terms of set systems.

Let $\mathcal{A} := (A_i : i \in I)$ be a finite family of (not necessarily distinct) subsets of a finite set A. A *system of distinct representatives* (SDR) for the family $\mathcal{A}$ is a set $\{a_i : i \in I\}$ of distinct elements of A such that $a_i \in A_i$ for all $i \in I$. In this language, Hall's Theorem says that $\mathcal{A}$ has a system of distinct representatives if and only if $| \cup_{i \in J} A_i| \geq |J|$ for all subsets J of I. (To see that this is indeed a reformulation of Hall's Theorem, let $G := G[X, Y]$, where $X := I$, $Y := A$, and $N(i) := A_i$ for all $i \in I$.) This was, in fact, the form in which Hall presented his theorem. He used it to answer a question in group theory (see Exercise 16.2.20).

Hall's Theorem provides a criterion for a bipartite graph to have a perfect matching.

Corollary 16.5 *A bipartite graph $G[X, Y]$ has a perfect matching if and only if $|X| = |Y|$ and $|N(S)| \geq |S|$ for all $S \subseteq X$.* $\square$

This criterion is satisfied by all nonempty regular bipartite graphs.

Corollary 16.6 *Every nonempty regular bipartite graph has a perfect matching.*

Proof Let $G[X, Y]$ be a k-regular bipartite graph, where $k \geq 1$. Then $|X| = |Y|$ (Exercise 1.1.9).

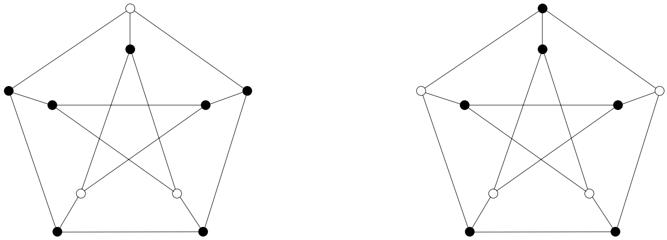
Now let S be a subset of X and let E_1 and E_2 denote the sets of edges of G incident with S and $N(S)$, respectively. By definition of $N(S)$, we have $E_1 \subseteq E_2$. Therefore

$$k|N(S)| = |E_2| \geq |E_1| = k|S|$$

Because $k \geq 1$, it follows that $|N(S)| \geq |S|$ and hence, by Corollary 16.5, that G has a perfect matching. $\square$

Matchings and Coverings

Recall that a *covering* of a graph G is a subset K of V such that every edge of G has at least one end in K. A covering K^* is a *minimum covering* if G has no covering K with $|K| < |K^*|$. The number of vertices in a minimum covering of G



(a) (b)

Fig. 16.7. (a) A minimal covering, (b) a minimum covering

is called the *covering number* of G, and is denoted by $\beta(G)$. A covering is *minimal* if none of its proper subsets is itself a covering. Minimal and minimum coverings of the Petersen graph are indicated (by solid vertices) in Figure 16.7.

If M is a matching of a graph G, and K is a covering of G, then at least one end of each edge of M belongs to K. Because all these ends are distinct, $|M| \leq |K|$. Moreover, if equality holds, then M is a maximum matching and K is a minimum covering (Exercise 16.2.2):

Proposition 16.7 *Let M be a matching and K a covering such that $|M| = |K|$. Then M is a maximum matching and K is a minimum covering.* □

The König–Egerváry Theorem (8.32) tells us that equality always holds when G is bipartite: for all bipartite graphs G,

$$\alpha'(G) = \beta(G)$$

This identity can be derived with ease from the theory of alternating paths. Let $G := G[X, Y]$ be a bipartite graph, let M^* be a maximum matching in G, and let U denote the set of vertices in X not covered by M^*. Denote by Z the set of all vertices in G reachable from some vertex in U by M^*-alternating paths, and set $R := X \cap Z$ and $B := Y \cap Z$. Then $K^* := (X \setminus R) \cup B$ is a covering with $|K^*| = |M^*|$ (Exercise 16.2.7). By Proposition 16.7, K^* is a minimum covering.

Exercises

16.2.1

a) Show that a bipartite graph G has a perfect matching if and only if $|N(S)| \geq |S|$ for all $S \subseteq V$.
b) Give an example to show that this condition does not guarantee the existence of a perfect matching in an arbitrary graph.

16.2.2 Prove Proposition 16.7.

16.2.3 A *line* of a matrix is a row or column of the matrix. Show that the minimum number of lines containing all the nonzero entries of a matrix is equal to the maximum number of nonzero entries, no two of which lie in a common line.

16.2.4 Using Exercise 16.1.15, give an inductive proof of the König–Egerváry Theorem (8.32). (D. de Caen)

$\star$**16.2.5** Let $\mathcal{A} := (A_i : i \in I)$ be a finite family of subsets of a finite set A, and let $f : I \to \mathbb{N}$ be a nonnegative integer-valued function. An *f-SDR* of $\mathcal{A}$ is a family $(S_i : i \in I)$ of disjoint subsets of A such that $S_i \subseteq A_i$ and $|S_i| = f(i)$, $i \in I$. (Thus, when $f(i) = 1$ for all $i \in I$, an f-SDR of $\mathcal{A}$ is simply an SDR of $\mathcal{A}$.)

a) Consider the family $\mathcal{B}$ of subsets of A consisting of $f(i)$ copies of A_i, $i \in I$. Show that $\mathcal{A}$ has an f-SDR if and only if $\mathcal{B}$ has an SDR.
b) Deduce, with the aid of Hall's Theorem (16.4), that $\mathcal{A}$ has an f-SDR if and only if
$$\left|\bigcup_{i\in J} A_i\right| \geq \sum_{i\in J} f(i) \quad \text{for all } J \subseteq I$$

16.2.6

a) Show that every minimal covering of a bipartite graph $G[X, Y]$ is of the form $N(S) \cup (X \setminus S)$ for some subset S of X.
b) Deduce Hall's Theorem (16.4) from the König–Egerváry Theorem (8.32).

$\star$**16.2.7** Let $G := G[X, Y]$ be a bipartite graph, let M^* be a maximum matching in G, and let U be the set of vertices in X not covered by M^*. Denote by Z the set of all vertices in G reachable from some vertex in U by M^*-alternating paths, and set $R := X \cap Z$ and $B := Y \cap Z$. Show that:

a) $K^* := (X \setminus R) \cup B$ is a covering of G,
b) $|K^*| = |M^*|$.

$\star$**16.2.8** The König–Ore Formula

a) Let $G := G[X, Y]$ be a bipartite graph, M a matching in G, and U the set of vertices in X not covered by M. Show that:
   i) for any subset S of X, $|U| \geq |N(S)| - |S|$,
   ii) $|U| = |N(S)| - |S|$ if and only if M is a maximum matching of G.
b) Prove the following generalization of Hall's Theorem (16.4):

   The matching number of a bipartite graph $G := G[X, Y]$ is given by:
$$\alpha' = |X| - \max\{|S| - |N(S)| : S \subseteq X\}$$
   This expression for α' is known as the *König–Ore Formula*.

16.2.9 Deduce from the König–Egerváry Theorem (8.32) that if $G := G[X, Y]$ is a simple bipartite graph, with $|X| = |Y| = n$ and $m > (k-1)n$, then $\alpha' \geq k$.

16.2.10

a) Let G be a graph and let (X, Y) be a partition of V such that $G[X]$ and $G[Y]$ are both k-colourable. If the edge cut $[X, Y]$ has at most $k-1$ edges, show that G also is k-colourable. (P. Kainen)
b) Deduce that every k-critical graph is $(k-1)$-edge-connected. (G.A. Dirac)

$\star$**16.2.11** Recall that an *edge covering* of a graph without isolated vertices is a set of edges incident with all the vertices, and that the number of edges in a minimum edge covering of a graph G is denoted by $\beta'(G)$. Show that $\alpha' + \beta' = n$ for any graph G without isolated vertices. (T. Gallai)

16.2.12 Let $G := G[X, Y]$ be a bipartite graph in which each vertex of X is of odd degree. Suppose that any two vertices of X have an even number of common neighbours. Show that G has a matching covering every vertex of X. (N. Alon)

16.2.13 Let $G := G[X, Y]$ be a bipartite graph such that $d(x) \geq 1$ for all $x \in X$ and $d(x) \geq d(y)$ for all $xy \in E$, where $x \in X$ and $y \in Y$. Show that G has a matching covering every vertex of X. (N. Alon)

16.2.14 Show that a bipartite graph $G[X, Y]$ has an f-factor with $f(x) = 1$ for all $x \in X$ and $f(y) \leq k$ for all $y \in Y$ if and only if $|N(S)| \geq |S|/k$ for all $S \subseteq X$.

16.2.15 A 2-*branching* is a branching in which each vertex other than the root has outdegree at most two. Let T be a tournament, and let v be a vertex of maximum outdegree in T. Set $Y{:=}N^+(v)$ and $X := V \setminus (Y \cup \{v\})$, and denote by $G[X, Y]$ the bipartite graph in which $x \in X$ is adjacent to $y \in Y$ if and only if y dominates x in T. For $S \subseteq X$, denote by $N(S)$ the set of neighbours of S in G.

a) Show that $|N(S)| \geq \frac{1}{2}|S|$, for all $S \subseteq X$.
b) By applying Exercise 16.2.14, deduce that T has a spanning 2-branching of depth at most two with root x. (X. Lu)

16.2.16 Let $\mathcal{C} = \{C_i : 1 \leq i \leq n\}$ be a family of n directed cycles in a digraph D. Show that there exist arcs $a_i \in A(C_i)$, $1 \leq i \leq n$, such that $D[\{a_i : 1 \leq i \leq n\}]$ contains a directed cycle. (A. Frank and L. Lovász)

16.2.17

a) Let G be a graph in which each vertex is of degree either k or $k+1$, where $k \geq 1$. Prove that G has a spanning subgraph H in which:
   i) each vertex is of degree either k or $k+1$,
   ii) the vertices of degree $k+1$ form a stable set.
b) Let H be a graph satisfying conditions (i) and (ii) of (a), where $k \geq 1$. Denote by X the set of vertices in H of degree $k+1$ and by Y the set of vertices in H of degree k. Prove that H has a spanning bipartite subgraph $B(X, Y)$ in which:

i) each vertex of X has degree $k+1$,
ii) each vertex of Y has degree at most k.

c) Let $B(X,Y)$ be a bipartite graph satisfying conditions (i) and (ii) of (b). Prove that there is a matching M in B which covers every vertex of X.
d) Deduce from (a), (b), and (c) that if G is a graph in which each vertex is of degree either k or $k+1$, where $k \geq 1$, then G contains a spanning subgraph in which each vertex is of degree either $k-1$ or k.

(W.T. Tutte; C. Thomassen)

16.2.18 The Birkhoff–von Neumann Theorem
A nonnegative real matrix is *doubly stochastic* if each of its line sums is 1. A *permutation matrix* is a $(0,1)$-matrix which has exactly one 1 in each line. (Thus every permutation matrix is doubly stochastic.) Let $\mathbf{Q}$ be a doubly stochastic matrix. Show that:

a) $\mathbf{Q}$ is a square matrix,
b) $\mathbf{Q}$ can be expressed as a convex linear combination of permutation matrices, that is,

$$\mathbf{Q} = c_1\mathbf{P}_1 + c_2\mathbf{P}_2 + \cdots + c_k\mathbf{P}_k$$

where each $\mathbf{P}_i$ is a permutation matrix, each c_i is a nonnegative real number, and $\sum_{i=1}^{k} c_i = 1$. (G. Birkhoff; J. von Neumann)

16.2.19 Let $\mathcal{A} := (A_i : i \in I)$ and $\mathcal{B} := (B_i : i \in I)$ be two finite families of subsets of a finite set A. Construct a digraph $D(x,y)$ with the property that some SDR of $\mathcal{A}$ is also an SDR of $\mathcal{B}$ if and only if there are $|I|$ internally disjoint directed (x,y)-paths in D.

16.2.20 Let H be a finite group and let K be a subgroup of H. Show that there exist elements $h_1, h_2, \ldots, h_n \in H$ such that $h_1K, h_2K, \ldots, h_nK$ are the left cosets of K and $Kh_1, Kh_2, \ldots, Kh_n$ are the right cosets of K. (P. Hall)

16.2.21 Let $G[X,Y]$ be a bipartite graph, and let S_1 and S_2 be subsets of X. Show that

$$|N(S_1)| + |N(S_2)| \geq |N(S_1 \cup S_2)| + |N(S_1 \cap S_2)|$$

16.2.22 Let $G[X,Y]$ be a bipartite graph in which $|N(S)| \geq |S|$ for all $S \subseteq X$.

a) A subset S of X is said to be *tight* if $|N(S)| = |S|$. Deduce from Exercise 16.2.21 that the union and intersection of tight subsets are tight also.
b) Deduce Hall's Theorem (16.4), that G has a matching covering X, by induction on n, proceeding as follows.
   i) Suppose, first, that there are no nonempty proper tight subsets of X. Let xy be an edge of $G[X,Y]$ with $x \in X$ and $y \in Y$. Show that, for every subset S of $X \setminus \{x\}$, $|N_{G'}(S)| \geq |S|$, where $G' = G - \{x,y\}$. (In this case, by induction, G' has a matching M' that covers $X \setminus \{x\}$, and $M' \cup \{xy\}$ is a matching of G that covers X.)

ii) Suppose, now, that T is a nonempty proper tight subset of X. Let G_1 denote the subgraph of G induced by $T \cup N(T)$ and let $G_2 := G - (T \cup N(T))$. Show that $|N_{G_1}(S)| \geq |S|$, for all $S \subseteq T$ and $|N_{G_2}(S)| \geq |S|$, for all $S \subseteq X \setminus T$. (In this case, by induction, G_1 has a matching M_1 that covers T, and G_2 has a matching M_2 that covers $X \setminus T$, so $M_1 \cup M_2$ is a matching of G that covers X.) (P.R. Halmos and H.E. Vaughn)

16.2.23 A nonempty connected graph is *matching-covered* if every edge belongs to some perfect matching. Let $G := G[X, Y]$ be a connected bipartite graph with a perfect matching. Show that:

a) G is matching-covered if and only if X has no nonempty proper tight subsets,
b) if G is matching covered, then $G - \{x, y\}$ has a perfect matching for all $x \in X$ and all $y \in Y$.

16.2.24 Dulmage–Mendelsohn Decomposition
Let $G[X, Y]$ be a bipartite graph with a perfect matching. Show that there exist a positive integer k and partitions $(X_1, X_2, \ldots, X_k)$ of X and $(Y_1, Y_2, \ldots, Y_k)$ of Y such that, for $1 \leq i \leq k$,

i) the subgraph $G[X_i \cup Y_i]$ of $G[X, Y]$ induced by $X_i \cup Y_i$ is matching-covered,
ii) $N(X_i) \subseteq Y_1 \cup Y_2 \cup \cdots \cup Y_i$. (L. Dulmage and N.S. Mendelsohn)

16.2.25 Let G be a matching-covered bipartite graph.

a) Show that G has an *odd-ear decomposition*, that is, a nested sequence of subgraphs $(G_0, G_1, \ldots, G_k)$ such that $G_0 \cong K_2$, $G_k = G$, and, $G_{i+1} = G_i \cup P_i$, $0 \leq i < k$, where P_i is an ear of G_i of odd length.
b) Show that, in any such decomposition, G_i is matching-covered for all i, $0 \leq i \leq k$.
c) Deduce that G has $m - n + 2$ perfect matchings whose incidence vectors are linearly independent.
d) The *matching space* of G is the vector space generated by the set of incidence vectors of perfect matchings of G. Show that the dimension of this space is $m - n + 2$. (J. Edmonds, L. Lovász, and W.R. Pulleyblank)

(The corresponding results for nonbipartite matching-covered graphs are considerably more difficult; see Carvalho et al. (2002).)

16.2.26

a) Let $G[X, Y]$ be a bipartite graph on $2n$ vertices in which all vertices have degree three except for one vertex in X and one vertex in Y, which have degree two. Show that $pm(G) \geq 2(4/3)^{n-1}$.
b) Deduce that if G is a cubic bipartite graph on $2n$ vertices, then $pm(G) \geq (4/3)^n$. (M. Voorhoeve)

16.2.27

a) Let $G[X, Y]$ be an infinite bipartite graph. Show that the condition $|N(S)| \geq |S|$, for every finite subset S of X, is a necessary condition for G to have a matching covering every vertex of X.
b) Give an example of a countable bipartite graph $G[X, Y]$ for which this condition is not sufficient for the existence of such a matching.

16.3 Matchings in Arbitrary Graphs

In this section, we derive a min–max formula for the number of edges in a maximum matching of an arbitrary graph, analogous to the König–Ore Formula for bipartite graphs (see Exercise 16.2.8). We begin by establishing an upper bound for this number.

Barriers

If M is a matching in a graph G, each odd component of G must clearly include at least one vertex not covered by M. Therefore $|U| \geq o(G)$, where U denotes the set of such vertices and $o(G)$ the number of odd components of G. This inequality can be extended to all induced subgraphs of G as follows.

Let S be a proper subset of V and let M be a matching in G. Consider an odd component H of $G - S$. If every vertex of H is covered by M, at least one vertex of H must be matched with a vertex of S. Because no more than $|S|$ vertices of $G - S$ can be matched with vertices of S, at least $o(G - S) - |S|$ odd components of G must contain vertices not covered by M. This observation yields the following inequality, valid for all proper subsets S of V.

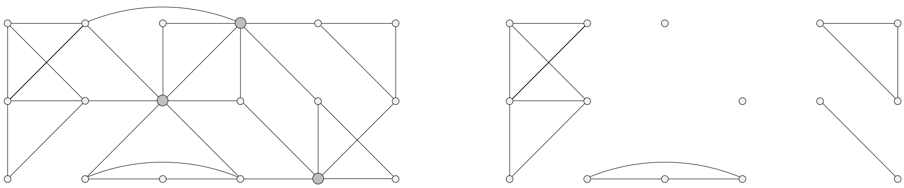
$$|U| \geq o(G - S) - |S| \tag{16.2}$$

From this inequality we may deduce, for example, that the Sylvester graph (of Figure 16.5) has no perfect matching, because three odd components are obtained upon deleting its central cut vertex. Likewise, the indicated set S of three vertices in the graph G of Figure 16.8a shows that any matching M must leave at least $5 - 3 = 2$ uncovered vertices because $G - S$ has five odd components (and one even component), as shown in Figure 16.8b.

Note that if equality should hold in (16.2) for some matching M and some subset $S := B$ of V, that is, if

$$|U| = o(G - B) - |B| \tag{16.3}$$

where $|U| = v(G) - 2|M|$, then the set B would show that the matching M leaves as few uncovered vertices as possible, and hence is a maximum matching (Exercise 16.3.1). Thus, B would serve as a succinct certificate of the optimality of M. Such a set B is called a *barrier* of G. The set of three vertices indicated in



(a) (b)

Fig. 16.8. A set S with $o(G - S) > |S|$

the graph of Figure 16.8 is a barrier, because this graph has a matching covering all but two of its vertices (see Figure 16.15a).

A matchable graph (one with a perfect matching) has both the empty set and all singletons as barriers. The empty set is also a barrier of a graph when some vertex-deleted subgraph is matchable. Graphs which are very nearly matchable, in the sense that every vertex-deleted subgraph is matchable, are said to be *hypomatchable* or *factor-critical*. In particular, trivial graphs are hypomatchable. For future reference, we state as a lemma the observation that all hypomatchable graphs have the empty set as a barrier. (Indeed, the empty set is their only barrier, see Exercise 16.3.8.)

Lemma 16.8 *The empty set is a barrier of every hypomatchable graph.* □

The Tutte–Berge Theorem

In a bipartite graph, a minimum covering constitutes a barrier of the graph (Exercise 16.3.4). More generally, every graph has a barrier. This fact is known as the *Tutte–Berge Theorem*. We present a proof by Gallai (1964a) of this theorem. It proceeds by induction on the number of vertices. By Lemma 16.8, a trivial graph has the empty set as barrier.

Recall that a vertex v of a graph G is *essential* if every maximum matching covers v, and *inessential* otherwise. Thus v is essential if $\alpha'(G - v) = \alpha'(G) - 1$ and inessential if $\alpha'(G - v) = \alpha'(G)$. We leave the proof of the following lemma as an exercise (16.3.5).

Lemma 16.9 *Let v be an essential vertex of a graph G and let B be a barrier of $G - v$. Then $B \cup \{v\}$ is a barrier of G.* □

By Lemma 16.9, in order to show that every graph has a barrier, it suffices to consider graphs with no essential vertices. It turns out that such graphs always have the empty set as a barrier. We establish this fact for connected graphs. Its validity for all graphs can be deduced without difficulty from this special case (Exercise 16.3.6).

Lemma 16.10 *Let G be a connected graph no vertex of which is essential. Then G is hypomatchable.*

Proof Since no vertex of G is essential, G has no perfect matching. It remains to show that every vertex-deleted subgraph has a perfect matching. If this is not so, then each maximum matching leaves at least two vertices uncovered. Thus it suffices to show that for any maximum matching and any two vertices in G, the matching covers at least one of these vertices. We establish this by induction on the distance between these two vertices.

Consider a maximum matching M and two vertices x and y in G. Let xPy be a shortest xy-path in G. Suppose that neither x nor y is covered by M. Because M is maximal, P has length at least two. Let v be an internal vertex of P. Since xPv is shorter than P, the vertex v is covered by M, by induction. On the other hand, because v is inessential, G has a maximum matching M' which does not cover v. Furthermore, because xPv and vPy are both shorter than P, the matching M' covers both x and y, again by induction.

The components of $G[M \triangle M']$ are even paths and cycles whose edges belong alternately to M and M' (Exercise 16.1.5). Each of the vertices x, v, y is covered by exactly one of the two matchings and thus is an end of one of the paths. Because the paths are even, x and y are not ends of the same path. Moreover, the paths starting at x and y cannot both end at v. We may therefore suppose that the path Q that starts at x ends neither at v nor at y. But then the matching $M' \triangle E(Q)$ is a maximum matching which covers neither x nor v, contradicting the induction hypothesis and establishing the lemma. □

One may now deduce (Exercise 16.3.7) the following fundamental theorem and corollary. These results, obtained by Berge (1958), can also be derived from a theorem of Tutte (1947a) on perfect matchings (Theorem 16.13).

Theorem 16.11 THE TUTTE–BERGE THEOREM
Every graph has a barrier. □

Corollary 16.12 THE TUTTE–BERGE FORMULA
For any graph G:

$$\alpha'(G) = \frac{1}{2}\min\{v(G) - (o(G-S) - |S|) : S \subset V\} \qquad \square$$

A refinement of Theorem 16.11 states that every graph G has a barrier B such that each odd component of $G - B$ is hypomatchable and each even component of $G - B$ has a perfect matching. Such a barrier is known as a *Gallai barrier.* In Section 16.5, we present a polynomial-time algorithm which finds not only a maximum matching in a graph, but also a succinct certificate for the optimality of the matching, namely a Gallai barrier.

Exercises

$\star$**16.3.1** Let M be a matching in a graph G, and let B be a set of vertices of G such that $|U| = o(G - B) - |B|$, where U is the set of vertices of G not covered by M. Show that M is a maximum matching of G.

$\star$**16.3.2** Let G be a graph and S a proper subset of V. Show that $o(G-S)-|S| \equiv v(G) \pmod 2$.

$\star$**16.3.3** Show that the union of barriers of the components of a graph is a barrier of the entire graph.

$\star$**16.3.4** Show that, in a bipartite graph, any minimum covering is a barrier of the graph.

$\star$**16.3.5** Give a proof of Lemma 16.9.

$\star$**16.3.6** Deduce from Lemma 16.10 that the empty set is a barrier of every graph without essential vertices.

$\star$**16.3.7**

a) Prove the Tutte–Berge Theorem (Theorem 16.11) by induction on the number of vertices.
b) Deduce the Tutte–Berge Formula (Corollary 16.12) from the Tutte–Berge Theorem.

———— ≀≀ ————

16.3.8

a) Show that:
   i) a graph is hypomatchable if and only if each of its blocks is hypomatchable,
   ii) a graph G is hypomatchable if and only if $o(G - S) \leq |S| - 1$ for every nonempty proper subset S of V.
b) Deduce that a graph is hypomatchable if and only if the empty set is its only barrier.

16.3.9 Let B be a maximal barrier of a graph G. Show that each component of $G - B$ is hypomatchable.

16.3.10 Let G be a graph and let (X, Y) be a partition of V with $X, Y \neq \emptyset$. Show that if both $G\,/\,X$ and $G\,/\,Y$ are hypomatchable, then G is hypomatchable.

16.3.11 Let G be a nonseparable graph which has an odd-ear decomposition starting with an odd cycle (instead of K_2). Show that G is hypomatchable.

16.3.12

a) Let x and y be adjacent inessential vertices of a graph G and let M and N be maximum matchings of $G-x$ and $G-y$, respectively. Show that G has an xy-path of even length whose edges belong alternately to N and M.
b) Deduce that every nontrivial hypomatchable graph G contains an odd cycle C such that $G\,/\,C$ is hypomatchable.
c) Prove the converse of the statement in Exercise 16.3.11: show that every nontrivial nonseparable hypomatchable graph has an odd-ear decomposition starting with an odd cycle (instead of K_2). (L. Lovász)

16.3.13 Let G be a k-chromatic graph containing no stable set of three vertices and no clique of k vertices, where $k \geq 3$. Let $k_1 + k_2$ be a partition of $k+1$ such that $k_1,\ k_2 \geq 2$. By appealing to Exercises 16.1.3 and 16.3.9, show that G has disjoint subgraphs G_1 and G_2 such that $\chi(G_1) = k_1$ and $\chi(G_2) = k_2$.
(L. Lovász and P.D. Seymour)

16.4 Perfect Matchings and Factors

Tutte's Theorem

If a graph G has a perfect matching M, then it follows from (16.2) that $o(G-S) \leq |S|$ for all $S \subseteq V$, because the set U of uncovered vertices is empty. The following fundamental theorem due to Tutte (1947a) shows that the converse is true. It is a special case of the Tutte–Berge Formula (Corollary 16.12).

Theorem 16.13 Tutte's Theorem
A graph G has a perfect matching if and only if

$$o(G-S) \leq |S| \quad \textit{for all} \quad S \subseteq V \tag{16.4}$$

Proof As already noted, (16.4) holds if G has a perfect matching. Conversely, let G be a graph which has no perfect matching. Consider a maximum matching M^* of G, and denote by U the set of vertices in G not covered by M^*. By Theorem 16.11, G has a barrier, that is, a subset B of V such that $o(G-B) - |B| = |U|$. Because M^* is not perfect, $|U|$ is positive. Thus

$$o(G-B) = |B| + |U| \geq |B| + 1$$

and Tutte's condition (16.4) fails for the set $S := B$. □

The first significant result on perfect matchings in graphs was obtained by Petersen (1891) in connection with a problem about factoring homogeneous polynomials into irreducible factors (see Biggs et al. (1986) and Sabidussi (1992)). In this context, perfect matchings correspond to factors of degree one; it is for this reason that they are also referred to as '1-factors'; it is also the origin of the term 'degree'. Petersen was particularly interested in the case of polynomials of degree three; these correspond to 3-regular graphs.

Theorem 16.14 PETERSEN'S THEOREM
Every 3-regular graph without cut edges has a perfect matching.

Proof We derive Petersen's Theorem from Tutte's Theorem (16.13).

Let G be a 3-regular graph without cut edges, and let S be a subset of V. Consider the vertex sets $S_1, S_2, \ldots, S_k$, of the odd components of $G - S$. Because G has no cut edges, $d(S_i) \geq 2$, $1 \leq i \leq k$. But because $|S_i|$ is odd, $d(S_i)$ is odd also (Exercise 2.5.5). Thus, in fact,

$$d(S_i) \geq 3, \quad 1 \leq i \leq k$$

Now the edge cuts $\partial(S_i)$ are pairwise disjoint, and are contained in the edge cut $\partial(S)$, so we have:

$$3k \leq \sum_{i=1}^{k} d(S_i) = d(\cup_{i=1}^{k} S_i) \leq d(S) \leq 3|S|$$

Therefore $o(G - S) = k \leq |S|$, and it follows from Theorem 16.13 that G has a perfect matching. □

The condition in Petersen's Theorem that the graph be free of cut edges cannot be omitted: the Sylvester graph of Figure 16.5, for instance, has no perfect matching. However, a stronger form of the theorem may be deduced from Tutte's Theorem (16.13), namely that each edge of a 3-regular graph without cut edges belongs to some perfect matching (Exercise 16.4.8).

FACTORS

Let G be a graph and let f be a nonnegative integer-valued function on V. An *f-factor* of G is a spanning subgraph F of G such that $d_F(v) = f(v)$ for all $v \in V$. A *k-factor* of G is an f-factor with $f(v) := k$ for all $v \in V$; in particular, a 1-factor is a spanning subgraph whose edge set is a perfect matching and a 2-factor is a spanning subgraph whose components are cycles.

Many interesting graph-theoretical problems can be solved in polynomial time by reducing them to problems about 1-factors. One example is the question of deciding whether a given graph G has an f-factor. Tutte (1954b) showed how this problem can be reduced to the problem of deciding whether a related graph G' has a 1-factor. We now describe this reduction procedure. We may assume that $d(v) \geq f(v)$ for all $v \in V$; otherwise G obviously has no f-factor. For simplicity, we assume that our graph G is loopless.

For each vertex v of G, we first replace v by a set Y_v of $d(v)$ vertices, each of degree one. We then add a set X_v of $d(v) - f(v)$ vertices and form a complete bipartite graph H_v by joining each vertex of X_v to each vertex of Y_v. In effect, the resulting graph H is obtained from G by replacing each vertex v by a complete bipartite graph $H_v[X_v, Y_v]$ and joining each edge incident to v to a separate element of Y_v. Figure 16.9 illustrates this construction in the case of 2-factors. Note that

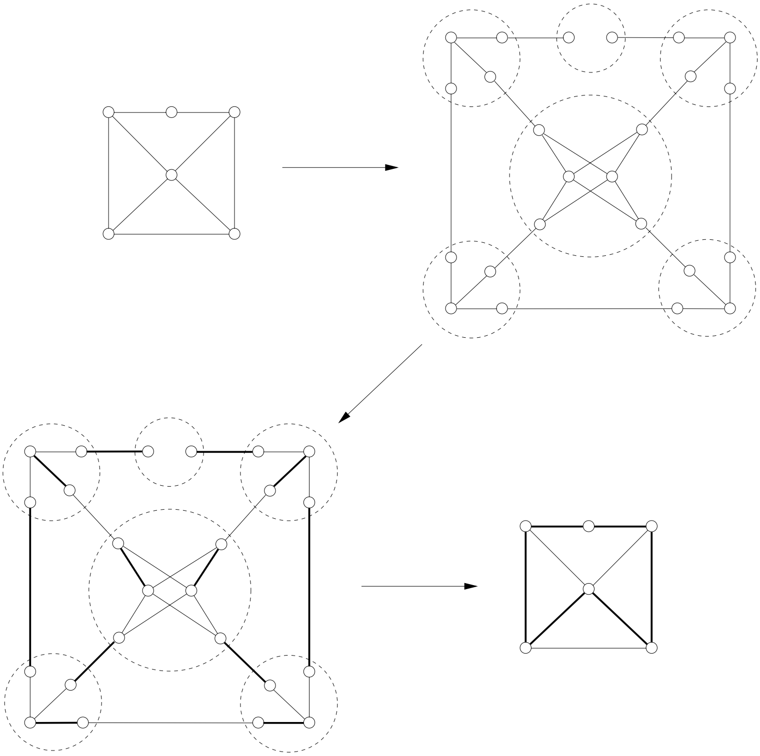


Fig. 16.9. Polynomial reduction of the 2-factor problem to the 1-factor problem

G can be recovered from H simply by shrinking each bipartite subgraph H_v to a single vertex v.

In H, the vertices of X_v are joined only to the vertices of Y_v. Thus if F is a 1-factor of H, the $d(v) - f(v)$ vertices of X_v are matched by F with $d(v) - f(v)$ of the $d(v)$ vertices in Y_v. The remaining $f(v)$ vertices of Y_v are therefore matched by F with $f(v)$ vertices of $V(H) \setminus V(H_v)$. Upon shrinking H to G, the 1-factor F of H is therefore transformed into an f-factor of G. Conversely, any f-factor of G can easily be converted into a 1-factor of H.

This reduction of the f-factor problem to the 1-factor problem is a polynomial reduction (Exercise 16.4.2).

T-Joins

A number of problems in graph theory and combinatorial optimization amount to finding a spanning subgraph H of a graph G (or a spanning subgraph of minimum weight, in the case of weighted graphs) whose degrees have prescribed parities (rather than prescribed values, as in the f-factor problem). Precise statements of such problems require the notion of a T-join.

Let G be a graph and let T be an even subset of V. A spanning subgraph H of G is called a *T-join* if $d_H(v)$ is odd for all $v \in T$ and even for all $v \in V \setminus T$. For example, a 1-factor of G is a V-join; and if P is an xy-path in G, the spanning subgraph of G with edge set $E(P)$ is an $\{x, y\}$-join.

Problem 16.15 THE WEIGHTED T-JOIN PROBLEM
GIVEN: *a weighted graph $G := (G, w)$ and a subset T of V,*
FIND: *a minimum-weight T-join in G (if one exists).*

As remarked above, the Shortest Path Problem may be viewed as a particular case of the Weighted T-Join Problem. Another special case is the *Postman Problem*, described in Exercise 16.4.22, whose solution involves finding a minimum-weight T-join when T is the set of vertices of odd degree in the graph (see Exercises 16.4.21 and 16.4.22).

By means of a construction similar to Tutte's reduction of the 2-factor problem to the 1-factor problem, one may obtain a polynomial reduction of the Weighted T-Join Problem to the following problem (see Exercise 16.4.21).

Problem 16.16 THE MINIMUM-WEIGHT MATCHING PROBLEM
GIVEN: *a weighted complete graph $G := (G, w)$ of even order,*
FIND: *a minimum-weight perfect matching in G.*

This latter problem can be seen to include the maximum matching problem: it suffices to embed the input graph G in a complete graph of even order, and assign weight zero to each edge of G and weight one to each of the remaining edges. Edmonds (1965b) found a polynomial-time algorithm for solving the Minimum-Weight Matching Problem. His algorithm relies on techniques from the theory of linear programming, and on his characterization of the perfect matching polytope (see Exercise 17.4.5).

Exercises

16.4.1 Show that a tree G has a perfect matching if and only if $o(G - v) = 1$ for all $v \in V$. (V. CHUNGPHAISAN)

$\star$**16.4.2**

a) Show that Tutte's reduction of the f-factor problem to the 1-factor problem is a polynomial reduction.
b) Describe how this polynomial reduction of the f-factor problem to the 1-factor problem can be generalized to handle graphs with loops.

16.4.3 Let G be a graph, and let $F := G[X]$ be an induced subgraph of V. Form a graph H from G as follows.

▷ Add edges between nonadjacent pairs of vertices in $V \setminus X$.
▷ If n is odd, add a new vertex and join it to all vertices in $V \setminus X$.

a) Show that there is a matching in G covering all vertices in X if and only if H has a perfect matching.
b) By applying Tutte's Theorem (16.13), deduce that G has a matching covering every vertex in X if and only, for all $S \subseteq V$, the number of odd components of $G - S$ which are subgraphs of $F - S$ is at most $|S|$.
c) Now suppose that F is bipartite. Using Exercise 16.3.9, strengthen the statement in (b) to show that there is a matching covering all vertices in X if and only if, for all $S \subseteq V$, the number of isolated vertices of $G - S$ which belong to $F - S$ is at most $|S|$.

16.4.4 Using Exercise 16.4.3b, derive Hall's Theorem (16.4) from Tutte's Theorem (16.13).

16.4.5 Let G be a graph whose vertices of degree Δ induce a bipartite subgraph. Using Exercise 16.4.3c, show that there is a matching in G covering all vertices of degree Δ. (H. Kierstead)

16.4.6 Derive the Tutte–Berge Formula (Corollary 16.12) from Tutte's Theorem (16.13).

16.4.7 Let G be a graph with a perfect matching and let $x, y \in V$.

a) Show that there is a barrier of G containing both x and y if and only if $G - \{x, y\}$ has no perfect matching.
b) Suppose that x and y are adjacent. Deduce from (a) that there is a perfect matching containing the edge xy if and only if no barrier of G contains both x and y.

16.4.8 Deduce from Tutte's Theorem (16.13) that every edge of a 3-regular graph without cut edges belongs to some perfect matching.

16.4.9

a) For $k \geq 1$, show that every $(k-1)$-edge-connected k-regular graph on an even number of vertices has a perfect matching.
b) For each $k \geq 2$, give an example of a $(k-2)$-edge-connected k-regular graph on an even number of vertices with no perfect matching.

16.4.10 A graph G on at least three vertices is *bicritical* if, for any two vertices u and v of G, the subgraph $G - \{u, v\}$ has a perfect matching.

a) Show that a graph is bicritical if and only if it has no barriers of cardinality greater than one.
b) Deduce that every essentially 4-edge-connected cubic nonbipartite graph is bicritical.

16.4.11

a) Show that every connected claw-free graph on an even number of vertices has a perfect matching. (M. Las Vergnas; D. Sumner)
b) Deduce that every 2-connected claw-free graph on an odd number of vertices is hypomatchable.

16.4.12 Let G be a simple graph with $\delta \geq 2(n-1)$ and containing no induced $K_{1,n}$, where $n \geq 3$. Show that G has a 2-factor. (K. Ota and T. Tokuda)

16.4.13 Show that every 2-connected graph that has one perfect matching has at least two perfect matchings. (A. Kotzig)

16.4.14

a) Show that a simple graph on $2n$ vertices with exactly one perfect matching has at most n^2 edges.
b) For all $n \geq 1$, construct a simple graph on $2n$ vertices with exactly one perfect matching and exactly n^2 edges.

16.4.15 Let $H = (V, \mathcal{F})$ be a hypergraph. A *cycle* in H is a sequence $v_1 F_1 v_2 F_2 \ldots v_k F_k v_{k+1}$, where v_i, $1 \leq i \leq k$, are distinct vertices, F_i, $1 \leq i \leq k$, are distinct edges, $v_{k+1} = v_1$, and $\{v_i, v_{i+1}\} \subseteq F_i$, $1 \leq i \leq k$. The hypergraph H is *balanced* if every odd cycle $v_1 F_1 v_2 F_2 \ldots v_{2k+1} F_{2k+1} v_1$ includes an edge containing at least three vertices of the cycle. A *perfect matching* in H is a set of disjoint edges whose union is V. Show that a balanced hypergraph $(V, \mathcal{F})$ has a perfect matching if and only if there exist disjoint subsets X and Y of V such that $|X| > |Y|$ and $|X \cap F| \leq |Y \cap F|$ for all $F \in \mathcal{F}$

$\star$**16.4.16** A graph G is *k-factorable* if it admits a decomposition into k-factors. Show that:

a) every k-regular bipartite graph is 1-factorable, (D. König)
b) every $2k$-regular graph is 2-factorable. (J. Petersen)

16.4.17

a) Show that the thickness of a $2k$-regular graph is at most k.
b) Find a 4-regular graph of thickness two and a 6-regular graph of thickness three.

16.4.18 Show that every triangulation with m edges contains a spanning bipartite subgraph with $2m/3$ edges. (F. Harary and D. Matula)

16.4.19 Show that every 2-connected 3-regular graph on four or more vertices admits a decomposition into paths of length three.

16.4.20 Let G be a graph. For $v \in V$, let $L(v) \subseteq \{0, 1, \ldots, d(v)\}$ be a list of integers associated with v.

a) Show that G has an f-factor with $f(v) \in L(v)$ for all $v \in V$ if $|L(v)| \geq d^+(v)+1$ for all $v \in V$, where D is an orientation of G.
(A. Frank, L. Lao, and J. Szabó; J.A. Bondy)

b) Deduce that G has an f-factor with $f(v) \in L(v)$ for all $v \in V$ if $|L(v)| > \lceil d(v)/2 \rceil$, for all $v \in V$. (H. Shirazi and J. Verstraëte)

16.4.21

a) Find a reduction of the Minimum-Weight T-Join Problem (16.15) to the Minimum-Weight Matching Problem (16.16).

b) Let G be a graph, let T be the set of vertices of odd degree in G, and let H be a T-join of G. Show that the graph obtained from G by duplicating the edges of H is an even graph.

16.4.22 The Postman Problem
In his job, a postman picks up mail at the post office, delivers it, and then returns to the post office. He must, of course, cover each street at least once. Subject to this condition, he wishes to choose a route entailing as little walking as as possible.

a) Show that the Postman Problem is equivalent to the following graph-theoretic problem.

Problem 16.17 Minimum-Weight Eulerian Spanning Subgraph
Given: *a weighted connected graph $G := (G, w)$ with nonnegative weights,*
Find: *by duplicating edges (along with their weights), an eulerian weighted spanning supergraph H of G whose weight $w(H)$ is as small as possible.*
(An Euler tour in H can then be found by applying Fleury's Algorithm (3.3).)
(M. Guan)

b) In the special case where G has just two vertices of odd degree, explain how the above problem can be solved in polynomial time.

16.4.23 Shortest Even and Odd Paths
Let $G := G(x, y)$ be a graph, and let H be the graph obtained from $G \,\square\, K_2$ by deleting the copies of x and y in one of the two copies of G.

a) Find a bijection between the xy-paths of even length in G and the perfect matchings in H.

b) By assigning weights 0 and 1 to the edges of H in an appropriate way and applying the weighted version of Edmonds' Algorithm, show how to find, in polynomial time, a shortest xy-path of even length in G.

c) By means of a similar construction, show how to find, in polynomial time, a shortest xy-path of odd length in G. (J. Edmonds)

16.4.24 By using minimum-weight matchings, refine the 2-approximation algorithm for the Metric Travelling Salesman Problem presented in Section 8.4, so as to obtain a polynomial-time $\frac{3}{2}$-approximation algorithm for this problem. (N. Christofides)

16.5 Matching Algorithms

This final section of the chapter is devoted to the description of a polynomial-time algorithm for finding a maximum matching in an arbitrary graph. We first consider the easier case of bipartite graphs, and then show how that algorithm can be refined to yield one applicable to all graphs.

Augmenting Path Search

Berge's Theorem (16.3) suggests a natural approach to finding a maximum matching in a graph. We start with some matching M (for example, the empty matching) and search for an M-augmenting path. If such a path P is found, we replace M by $M \triangle E(P)$. We repeat the procedure until no augmenting path with respect to the current matching can be found. This final matching is then a maximum matching.

The challenge here is to carry out an exhaustive search for an M-augmenting path in an efficient manner. This can indeed be achieved. In this section, we describe a polynomial-time algorithm which either finds an M-augmenting path in a bipartite graph, or else supplies a succinct certificate that there is no such path. This algorithm, as well as its extension to arbitrary graphs, is based on the notion of an M-alternating tree.

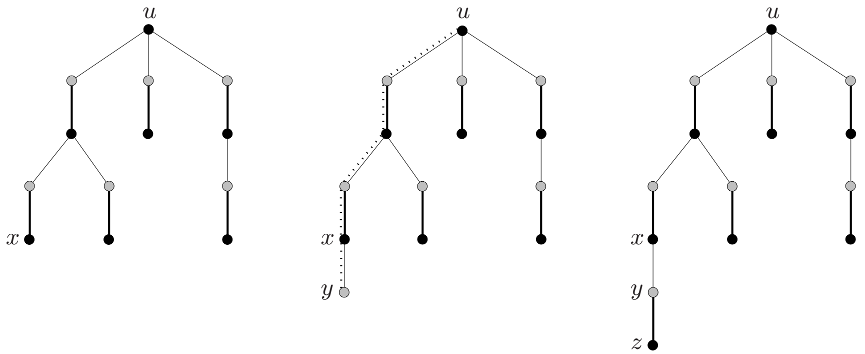
Let G be a graph, M a matching in G, and u a vertex not covered by M. A tree T of G is an *M-alternating u-tree* if $u \in V(T)$ and, for any $v \in V(T)$, the path uTv is an M-alternating path. An M-alternating u-tree T is *M-covered* if the matching $M \cap E(T)$ covers all vertices of T except u (see Figures 16.10a and 16.10c).

There is a simple tree-search algorithm, which we refer to as *Augmenting Path Search*, that finds either an M-augmenting u-path or else a maximal M-covered u-tree (that is, an M-covered u-tree which can be grown no further). We call such a tree an *APS-tree* (rooted at u).

Augmenting Path Search begins with the trivial M-covered tree consisting of just the vertex u. At each stage, it attempts to extend the current M-covered u-tree T to a larger one. We refer to those vertices at even distance from u in T as *red* vertices and those at odd distance as *blue* vertices; these sets will be denoted by $R(T)$ and $B(T)$, respectively (so that $(R(T), B(T))$ is a bipartition of T, with $u \in R(T)$). In the M-covered u-tree T displayed in Figure 16.10a, the red vertices are shown as solid dots and the blue vertices as shaded dots.

Consider an M-covered u-tree T. Being M-covered, T contains no M-augmenting u-path. If there is such a path in G, the edge cut $\partial(T)$ necessarily includes an edge of this path. Accordingly, we attempt to extend T to a larger M-alternating u-tree by adding to it an edge from its associated edge cut $\partial(T)$. This is possible only if there is an edge xy with $x \in R(T)$ and $y \in V(G) \setminus V(T)$. If there is no such edge, then T is an APS-tree rooted at u, and the procedure terminates. If, on the other hand, there is such an edge, two possibilities arise. Either y is not covered by M, in which case we have found our M-augmenting path (Figure 16.10b), or y

is incident to an edge yz of M, and we grow T into a larger M-covered u-tree by adding the two vertices y and z and the two edges xy and yz (Figure 16.10c).



(a) (b) (c)

Fig. 16.10. Augmenting Path Search: growing an M-covered tree

This tree-growing operation is repeated until either an M-augmenting u-path P is found, in which case the matching M is replaced by $M \triangle E(P)$, or the procedure terminates with an APS-tree T. It can be summarized as follows.

Algorithm 16.18 AUGMENTING PATH SEARCH: APS(G, M, u)

INPUT*:* a graph G, a matching M in G, and an uncovered vertex u of G
OUTPUT*:* a matching M with one more edge than the input matching, or an APS-tree T with root $u(T)$, a bipartition $(R(T), B(T))$ of T, and the set $M(T)$ of matching edges in T

set $V(T) := \{u\}$, $E(T) := \emptyset$, $R(T) := \{u\}$
while *there is an edge* xy *with* $x \in R(T)$ *and* $y \in V(G) \setminus V(T)$ ***do***
  replace $V(T)$ *by* $V(T) \cup \{y\}$ *and* $E(T)$ *by* $E(T) \cup \{xy\}$.
  if y *is not covered by* M ***then***
    replace M *by* $M \triangle E(P)$, *where* $P := uTy$
    return M
  else
    replace $V(T)$ *by* $V(T) \cup \{z\}$, $E(T)$ *by* $E(T) \cup \{yz\}$, *and* $R(T)$ *by* $R(T) \cup \{z\}$, *where* $yz \in M$
  end if
end while
set: $T := (V(T), E(T))$, $u(T) := u$, $B(T) := V(T) \setminus R(T)$, *and* $M(T) := M \cap E(T)$
return $(T, u(T), R(T), B(T), M(T))$

In the event that APS outputs an APS-tree T, we note for future reference that:

▷ because T is M-covered, the vertices of $R(T) \setminus u(T)$ are matched with the vertices of $B(T)$, so
$$|B(T)| = |R(T)| - 1 \tag{16.5}$$
and
$$B(T) \subseteq N(R(T)) \tag{16.6}$$
(where $N(R(T))$ denotes the set of neighbours of $R(T)$ in G).

▷ because T is maximal, no vertex of $R(T)$ is adjacent in G to any vertex of $V(G) \setminus V(T)$; that is,
$$N(R(T)) \subseteq R(T) \cup B(T) \tag{16.7}$$

If APS finds an M-augmenting path uPy, well and good. We simply apply APS once more, replacing M by the augmented matching $M \triangle E(P)$ returned by APS. But what if APS returns an APS-tree T? Can we then be sure that G contains no M-augmenting u-path? Unfortunately we cannot, as the example in Figure 16.11b illustrates. However, if it so happens that no two red vertices of T are adjacent in G, that is, if $N(R(T)) \cap R(T) = \emptyset$, then (16.6) and (16.7) imply that
$$N(R(T)) = B(T) \tag{16.8}$$
In this case, we may restrict our search for an M-augmenting path to the subgraph $G - T$. Indeed, the following stronger statement is true (Exercise 16.5.3).

Proposition 16.19 *Let T be an APS-tree returned by $APS(G, M, u)$. Suppose that no two red vertices of T are adjacent in G. Then no M-augmenting path in G can include any vertex of T.* □

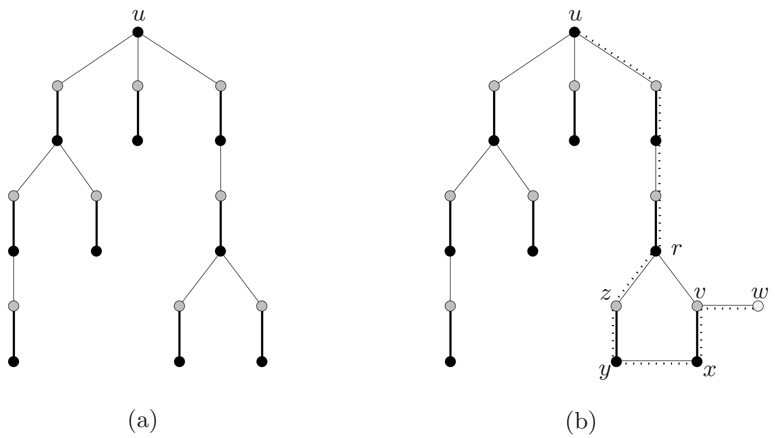


Fig. 16.11. (a) An APS-tree, (b) an M-augmenting u-path

One important instance where condition (16.8) is satisfied is when G is bipartite. In this case, no two red vertices of T can be adjacent in G because they all belong to the same part of the bipartition of G. Thus $\mathrm{APS}(G, M, u)$ finds all the vertices of G that can be reached by M-alternating u-paths. This observation is the basis of the following algorithm for finding a maximum matching in a bipartite graph. It was conceived by the Hungarian mathematician Egerváry (1931), and for this reason is sometimes referred to as the *Hungarian Algorithm*.

Egerváry's Algorithm

Let $G[X, Y]$ be a bipartite graph. In searching for a maximum matching of G, we start with an arbitrary matching M of G (for instance, the empty matching) and apply APS to search for an M-augmenting u-path, where u is an uncovered vertex. (If there are no such vertices, M is a perfect matching.) The output of APS is either an M-augmenting u-path P, or else an APS-tree T rooted at u. In the former case, we replace the matching M by $M \triangle E(P)$ and apply APS once more, starting with an uncovered vertex of G with respect to the new matching M, if there is one. In the latter eventuality, by Proposition 16.19 we may restrict our attention to the subgraph $G - V(T)$ in continuing our search for an M-augmenting path. We simply record the set $M(T) := M \cap E(T)$ (this set will be part of our maximum matching), replace the matching M by the residual matching $M \setminus E(T)$ and the graph G by the subgraph $G - V(T)$, and then apply APS once more, starting with an uncovered vertex of this subgraph, if there is one. We proceed in this way until the subgraph we are left with has no uncovered vertex (so has a perfect matching). The output of this algorithm is as follows.

▷ A set $\mathcal{T}$ of pairwise disjoint APS-trees.
▷ A set $R := \cup\{R(T) : T \in \mathcal{T}\}$ of red vertices.
▷ A set $B := \cup\{B(T) : T \in \mathcal{T}\}$ of blue vertices.
▷ A subgraph $F := G - (R \cup B)$ with a perfect matching $M(F)$.
▷ A matching $M^* := \cup\{M(T) : T \in \mathcal{T}\} \cup M(F)$ of G.
▷ A set $U := \{u(T) : T \in \mathcal{T}\}$ of vertices not covered by M^*.

(When the initial matching M is perfect, $\mathcal{T} = R = B = \emptyset$, $F = G$, $M^* = M$, and $U = \emptyset$.)

Example 16.20 Consider the bipartite graph in Figure 16.12a, with the indicated matching M. Figure 16.12b shows an M-alternating x_1-tree, which is grown until the M-augmenting x_1-path $P := x_1y_2x_2y_1$ is found. As before, the red vertices are indicated by solid dots and the blue vertices by shaded dots. Figure 16.12c shows the augmented matching $M \triangle E(P)$ (the new matching M), and Figure 16.12d an M-alternating x_4-tree which contains no M-augmenting x_4-path and can be grown no further, and thus is an APS-tree T_1 with $R(T_1) = \{x_1, x_3, x_4\}$ and $B(T_1) = \{y_2, y_3\}$. The set of all vertices reachable in G from x_4 by M-alternating paths is therefore $V(T_1) = \{x_1, x_3, x_4, y_2, y_3\}$. This set does not include y_4, the only other vertex not covered by M. Thus we may conclude that $M^* := M$ is a maximum matching. However, for the purpose of illustrating the entire algorithm, we continue, deleting $V(T_1)$ from G and growing an M-alternating y_4-tree in the resulting subgraph (see Figure 16.12e), thereby obtaining the APS-tree T_2 with $R(T_2) = \{y_1, y_4, y_5\}$ and $B(T_2) = \{x_2, x_5\}$, as shown in Figure 16.12f. The procedure ends there, because every vertex of the graph $F := G - V(T_1 \cup T_2)$, which consists of the vertices x_6 and y_6 and the edge x_6y_6, is covered by M. The output of the algorithm is therefore:

$$\mathcal{T} = \{T_1, T_2\}, \quad R = \{x_1, x_3, x_4, y_1, y_4, y_5\}, \quad B = \{y_2, y_3, x_2, x_5\}$$

$$V(F) = \{x_6, y_6\}, \quad E(F) = \{x_6y_6\}$$

$$M^* = \{x_1y_2, x_2y_1, x_3y_3, x_5y_5, x_6y_6\}, \quad U = \{x_4, y_4\}$$

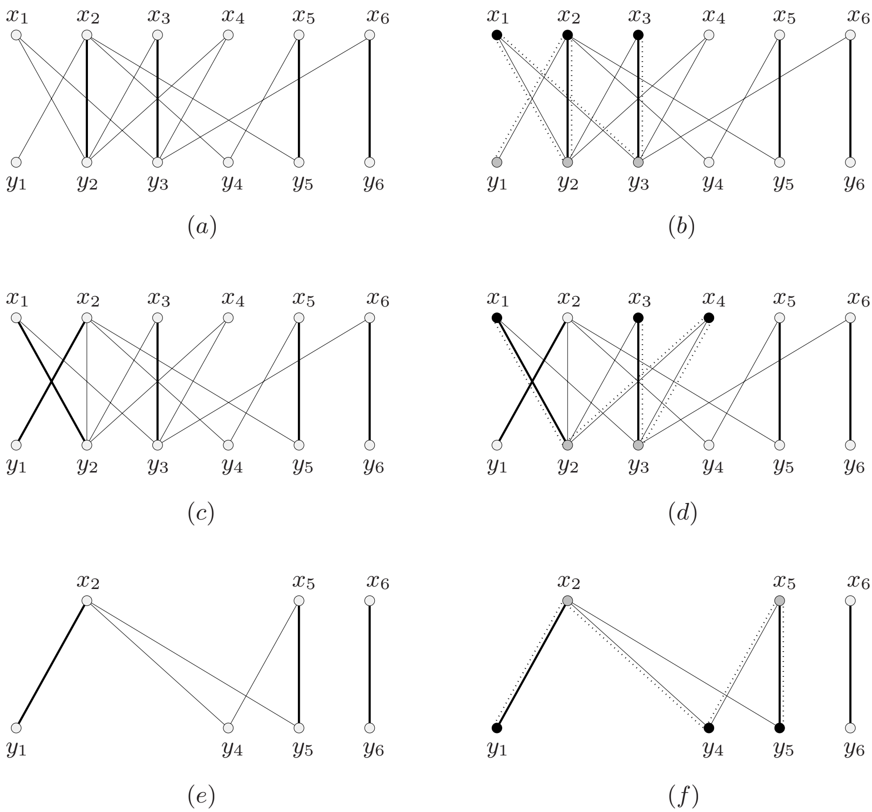


Fig. 16.12. Egerváry's Algorithm: finding a maximum matching in a bipartite graph

We now verify the correctness of Egerváry's Algorithm.

Theorem 16.21 *The matching M^* returned by Egerváry's Algorithm is a maximum matching.*

Proof Let $\mathcal{T}$, R, B, and U be the sets of trees, red vertices, blue vertices, and uncovered vertices returned by Egerváry's Algorithm. Because each tree $T \in \mathcal{T}$ contains exactly one uncovered vertex, namely its root $u(T)$, we have $|U| = |\mathcal{T}|$. Also, by (16.5), $|B(T)| = |R(T)| - 1$ for each tree $T \in \mathcal{T}$. Summing this identity over all $T \in \mathcal{T}$ gives

$$|B| = |R| - |\mathcal{T}|$$

Therefore

$$|U| = |R| - |B| \tag{16.9}$$

Because red vertices are adjacent in G only to blue vertices, in any matching M of G, red vertices can only be matched with blue vertices. There are therefore at least $|R| - |B|$ red vertices not covered by M. Thus, by (16.9), there are at least $|U|$ such vertices, no matter what the matching M. Because there are exactly this number of vertices not covered by M^*, we conclude that M^* is a maximum matching. $\square$

Egerváry's Algorithm returns not only a maximum matching M^* but also a covering K^* of the same size, which is consequently a minimum covering. To see this, let $G[X, Y]$ be a bipartite graph, and let M^* be a maximum matching of G. Consider the sets R and B of red and blue vertices output by Egerváry's Algorithm when applied to G with input matching M^*. Set $F := G - (R \cup B)$.

By (16.6) and (16.7), $N(R) = B$. Thus B covers all edges of G except those of F. Because $X \cap V(F)$ clearly covers $E(F)$, the union $B \cup (X \cap V(F))$ of these two sets is a covering K^* of G. Moreover, there is a bijection between M^* and K^* because each vertex of K^* is covered by M^* and each edge of M^* is incident with just one vertex of K^*. Hence $|M^*| = |K^*|$. It follows from Proposition 16.7 that the covering K^* is a minimum covering.

In view of the relationship between matchings in bipartite graphs and families of internally disjoint directed paths in digraphs (as described in Exercise 8.6.7), Egerváry's Algorithm may be viewed as a special case of the Max-Flow Min-Cut Algorithm presented in Chapter 7.

Blossoms

As the example in Figure 16.11b illustrates, Augmenting Path Search is not guaranteed to find an M-augmenting u-path, even if there is one, should there be two red vertices in the APS-tree that are adjacent in G. However, if we look more closely at this example, we see that the cycle $rvxyzr$ contains two alternating rv-paths, namely, the edge rv and the path $rzyxv$. Because the latter path ends with a matching edge, it may be extended by the edge vw, thereby yielding a uw-alternating path.

In general, suppose that T is an APS-tree of G rooted at u, and that x and y are two red vertices of T which are adjacent in G. The cycle contained in $T + xy$ is then called a *blossom*. A blossom C is necessarily of odd length, because each blue vertex is matched with a red vertex and there is one additional red vertex, which we call the *root* of C and denote by $r := r(C)$ (see Figure 16.13a). Note that $M \cap E(C)$ is a perfect matching of $C - r$. Note, also, that the path uTr is M-alternating, and terminates with a matching edge (unless $r = u$). Moreover, this path is internally disjoint from C.

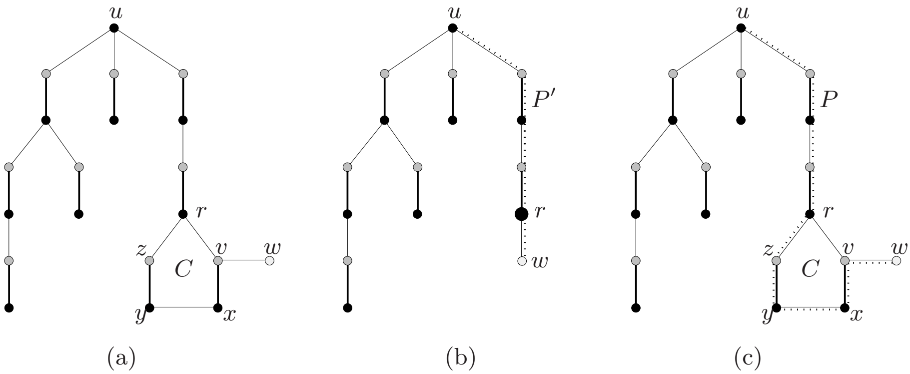


Fig. 16.13. (a) A blossom C, (b) an M-alternating u-path P' in $G' := G\,/\,C$, (c) an M-alternating u-path P in G

The key to finding a maximum matching in an arbitrary graph is to *shrink* blossoms (that is, contract them to single vertices) whenever they are encountered during APS. By shrinking a blossom and continuing to apply APS to the resulting graph, one might be able to reach vertices by M-alternating u-paths which could not have been reached before. For example, if T is an APS-tree with a blossom C, and if there happens to be an edge vw with $v \in V(C)$ and $w \in V(G) \setminus V(T)$, as in Figure 16.13a), then w is reachable from u by an M-alternating path P' in $G' := G\,/\,C$ (see Figure 16.13b, where the shrunken blossom $C = rvxyzr$ is indicated by a large solid dot), and this path P' can be modified to an M-alternating path P in G by inserting the rv-segment of C that ends with a matching edge (Figure 16.13c). In particular, if P' is an M-augmenting path in $G\,/\,C$, then the modified path P is an M-augmenting path of G. We refer to this process of obtaining an M-alternating path of G from an M-alternating path of $G\,/\,C$ as *unshrinking* C.

If C is a blossom with root r, we denote the vertex resulting from shrinking C by r also (and keep a record of the blossom C). The effect of shrinking a blossom C is to replace the graph G by $G\,/\,C$, the tree T by $(T + xy)\,/\,C$, where x and y are the adjacent red vertices of C, and the matching M by $M \setminus E(C)$. When we incorporate this blossom-shrinking operation into APS, we obtain a modified search procedure, APS$^+$.

By way of illustration, consider the graph G and the matching M shown in Figure 16.14a.

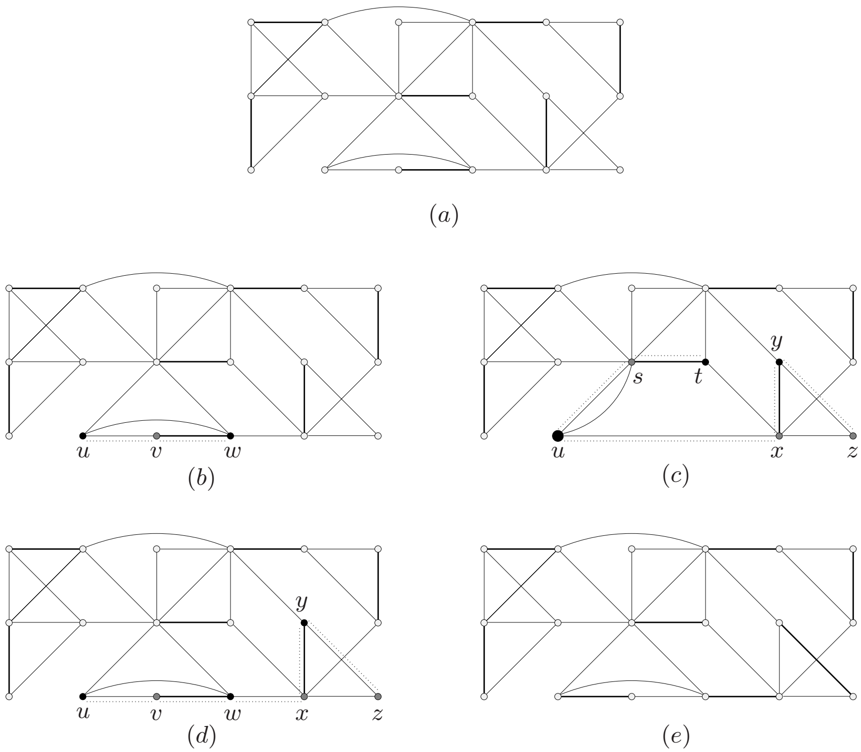


Fig. 16.14. Finding an M-augmenting path

We grow an M-alternating tree T rooted at the uncovered vertex u. A blossom $C = uvwu$ is found (Figure 16.14b) and shrunk to its root u. In Figure 16.14c, the contracted tree (now a single vertex) is grown further, and an M-augmenting path $uxyz$ is found in the contracted graph $G\,/\,C$, giving rise (after unshrinking the blossom C) to the M-augmenting path $uvwxyz$ in G, as shown in Figure 16.14d. The augmented matching is indicated in Figure 16.14e.

Starting with this new matching M and the vertex a not covered by it, the above procedure is now repeated, and evolves as illustrated in Figure 16.15, terminating with the APS-tree depicted in Figure 16.15g. Note that, because the vertex a of Figure 16.15c was obtained by shrinking the blossom $abca$, the blossom $adea$ is, in fact, a 'compound' blossom. We now examine the structure of such compound blossoms.

Flowers

As the above example illustrates, during the execution of APS$^+$, the graph G is repeatedly modified by the operation of shrinking blossoms. Suppose that $(C_0, C_1, \ldots, C_{k-1})$ is the sequence of blossoms shrunk, in that order, during the

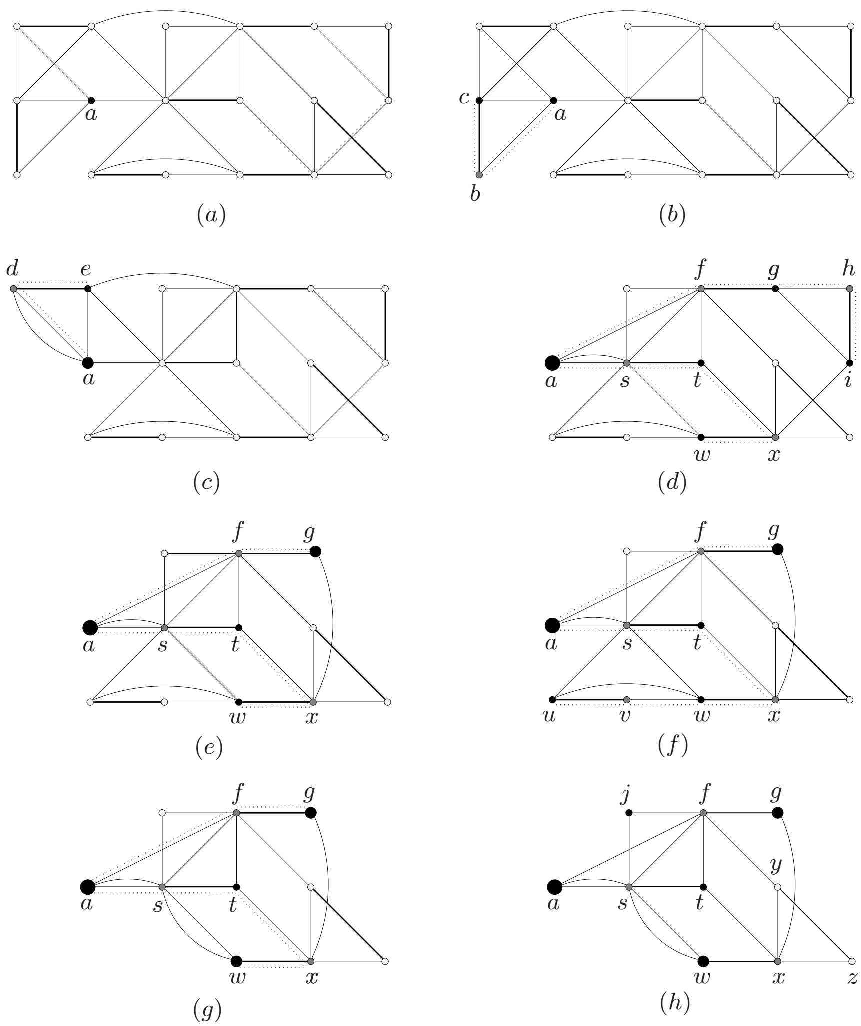


Fig. 16.15. Growing an APS-tree

execution of APS^+. The original graph G is thus progressively modified, yielding a sequence of graphs $(G_0, G_1, \ldots, G_k)$, where $G_0 := G$ and, for $0 \le i \le k-1$, $G_{i+1} := G_i \,/\, C_i$. If APS^+ fails to find an M-augmenting u-path, it terminates with an APS-tree T_k in G_k, no two red vertices of which are adjacent in G_k. Let us suppose that APS^+ returns such a tree.

Because a blossom C_i is always shrunk to its root, a red vertex, the blue vertices in each of the graphs G_i are simply vertices of the original input graph. However, the red vertices might well correspond to nontrivial induced subgraphs of the input graph. The subgraphs of G corresponding to the red vertices of G_i are called the *flowers* of G associated with T_i. For instance, the three flowers of the graph G of Figure 16.14a associated with the APS-tree shown in Figure 16.15g are the subgraphs of G induced by $\{a,b,c,d,e\}$, $\{g,h,i\}$, and $\{u,v,w\}$.

Flowers satisfy two basic properties, described in the following proposition.

Proposition 16.22 *Let F be a flower of G. Then:*

i) F is connected and of odd order,
ii) for any vertex v of F, there is an M-alternating uv-path in G of even length (that is, one terminating in an edge of M).

Proof The proof is by induction on i, where F is a flower associated with T_i. We leave the details to the reader (Exercise 16.5.8). □

We are now ready to prove the validity of Algorithm APS$^+$.

Corollary 16.23 *Let T_k be an APS-tree of G_k no two red vertices of which are adjacent in G_k. Then the red vertices of T_k are adjacent in G_k only to blue vertices of T_k. Equivalently, the flowers of G associated with T_k are adjacent only to the blue vertices of G in T_k.*

Proof It follows from Proposition 16.22(ii) that if G_k has an M-augmenting u-path, then so has G. Thus if G has no M-augmenting u-path, no red vertex of T_k can be adjacent in G_k to any vertex in $V(G_k) \setminus V(T_k)$ that is not covered by M. On the other hand, by the maximality of the APS-tree T_k, no red vertex of T_k is adjacent in G_k to any vertex in $V(G_k) \setminus V(T_k)$ that is covered by M. Because no two red vertices of T_k are adjacent in G_k, we deduce that the red vertices of T_k are adjacent only to the blue vertices of T_k. □

Recall that when G is a bipartite graph, APS(G,M,u) finds all vertices that can be reached by M-alternating u-paths. Algorithm APS$^+(G,M,u)$ achieves the same objective in an arbitrary graph G. This follows from the fact that if APS$^+(G,M,u)$ terminates with an APS-tree T_k, every M-alternating u-path in G that terminates in a blue vertex is of odd length (Exercise 16.5.10).

Edmonds' Algorithm

The idea of combining Augmenting Path Search with blossom-shrinking is due to Edmonds (1965d). It leads to a polynomial-time algorithm for finding a maximum matching in an arbitrary graph, in much the same way as Augmenting Path Search leads to Egerváry's Algorithm for finding a maximum matching in a bipartite graph.

In searching for a maximum matching of a graph G, we start with an arbitrary matching M of G, and apply APS$^+$ to search for an M-augmenting u-path in G, where u is an uncovered vertex. If such a path P is found, APS$^+$ returns the larger matching $M \triangle E(P)$; if not, APS$^+$ returns an APS-tree T rooted at u. In the former case, we apply APS$^+$ starting with an uncovered vertex of G with respect to the new matching M, if there is one. In the latter eventuality, in continuing our search for an M-augmenting path, we restrict our attention to the subgraph $G - V(T)$ (where, by $V(T)$, we mean the set of blue vertices of T together with the set of vertices of G included in the flowers of T). In this case, we record the set $M(T) := M \cap E(T)$ (this set will be part of our maximum matching), replace the matching M by the residual matching $M \setminus E(T)$ and the graph G by the subgraph $G - V(T)$, and then apply APS$^+$ once more, starting with an uncovered vertex of G with respect to this new matching, if there is one. We proceed in this way until the graph F we are left with has no uncovered vertices (and thus has a perfect matching).

For example, after having found the APS-tree in Figure 16.15g, there remains one uncovered vertex, namely j. The APS-tree grown from this vertex is just the trivial APS-tree. The subgraph F consists of the vertices y and z, together with the edge linking them. The red and blue vertices in the two APS-trees are indicated in Figure 16.15h.

The output of Edmonds' Algorithm is as follows.

- ▷ A set $\mathcal{T}$ of pairwise disjoint APS-trees.
- ▷ A set $R := \cup\{R(T) : T \in \mathcal{T}\}$ of red vertices.
- ▷ A set $B := \cup\{B(T) : T \in \mathcal{T}\}$ of blue vertices.
- ▷ A subgraph $F := G - (R \cup B)$ of G with a perfect matching $M(F)$.
- ▷ A matching $M^* := \cup\{M(T) : T \in \mathcal{T}\} \cup M(F)$ of G.
- ▷ A set $U := \{u(T) : T \in \mathcal{T}\}$ of vertices not covered by M^*.

(As in Egerváry's Algorithm, when the initial matching M is perfect, $\mathcal{T} = R = B = \emptyset$, $F = G$, $M^* = M$, and $U = \emptyset$.)

The proof that Edmonds' Algorithm does indeed return a maximum matching closely resembles the proof of Theorem 16.21. We leave it as an exercise (16.5.9).

Theorem 16.24 *The set B returned by Edmonds' Algorithm is a barrier and the matching M^* returned by the algorithm is a maximum matching.* □

To conclude, we note that Edmonds' Algorithm, combined with the polynomial reduction of the f-factor problem to the 1-factor problem described in Section 16.4, yields a polynomial-time algorithm for solving the f-factor problem.

Exercises

16.5.1 Apply Egerváry's Algorithm to find a maximum matching in the bipartite graph of Figure 16.16a.

16.5.2 Show that Egerváry's Algorithm is a polynomial-time algorithm.

⋆**16.5.3** Prove Proposition 16.19.

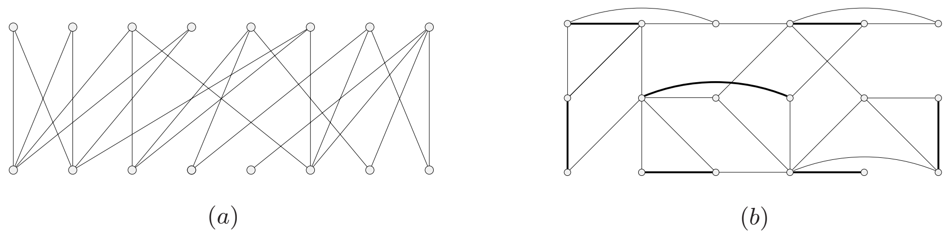


Fig. 16.16. Find maximum matchings in these graphs (Exercises 16.5.1 and 16.5.7)

16.5.4 Describe how the output of Egerváry's Algorithm can be used to find a minimum edge covering of an input bipartite graph without isolated vertices.

16.5.5 Find minimum coverings, maximum stable sets, and minimum edge coverings in the graphs of Figures 16.12 and 16.16a.

16.5.6 For any positive integer k, show that the complete k-partite graph $G := K_{2,2,\ldots,2}$ is k-choosable. (P. Erdős, A.L. Rubin, and H. Taylor)

16.5.7 Apply Edmonds' Algorithm to find a maximum matching in the graph of Figure 16.16b, starting with the matching M indicated. Determine the barrier output by the algorithm.

$\star$**16.5.8** Prove Proposition 16.22.

$\star$**16.5.9**

a) Show that the set B of blue vertices returned by Edmonds' Algorithm constitutes a barrier of G.
b) Give a proof of Theorem 16.24.

$\star$**16.5.10**

a) Let T be an APS-tree returned by $\mathrm{APS}^+(G, M, u)$. Show that every M-alternating u-path in G that terminates in a blue vertex is of odd length.
b) Deduce that $\mathrm{APS}^+(G, M, u)$ finds all vertices of G that can be reached by M-alternating u-paths.

16.5.11 Show that Edmonds' Algorithm is a polynomial-time algorithm.

16.5.12 Deduce from Exercise 16.3.10 that the flowers created during the execution of APS^+ are hypomatchable.

16.5.13 Let B be the barrier obtained by applying Edmonds' Algorithm to an input graph G. Show that:

a) every even component of $G - B$ has a perfect matching,

b) every odd component of $G - B$ is hypomatchable,
c) a vertex v of G is inessential if and only if it belongs to an odd component of $G - B$,
d) B is the set of all essential vertices that have some inessential neighbour.

(Gallai (1964a) was the first to show that every graph has a barrier satisfying the above conditions.)

16.6 Related Reading

Stable Sets in Claw-Free Graphs

Maximum stable sets in line graphs can be determined in polynomial time, by virtue of Edmonds' Algorithm (described in Section 16.5), because a stable set in a line graph $L(G)$ corresponds to a matching in G. More generally, there exist polynomial-time algorithms for finding maximum stable sets in claw-free graphs, a class which includes all line graphs (see Minty (1980), Sbihi (1980), or Lovász and Plummer (1986)).

Transversal Matroids

Let $G := G[X, Y]$ be a bipartite graph. A subset S of X is *matchable* with a subset of T of Y if there is a matching in G which covers $S \cup T$ and no other vertices. A subset of X is *matchable* if it is matchable with some subset of Y. Edmonds and Fulkerson (1965) showed that the matchable subsets of X are the independent sets of a matroid on X; matroids that arise in this manner are called *transversal matroids*. Various results described in Section 16.2 may be seen as properties of transversal matroids. For example, the König–Ore Formula (Exercise 16.2.8) is an expression for the rank of this matroid.

Rado's Theorem

Let $G := G[X, Y]$ be a bipartite graph, and let M be a matroid defined on Y with rank function r. As a far-reaching generalization of Hall's Theorem (16.4), Rado (1942) showed that X is matchable with a subset of Y which is independent in the matroid M if and only if $r(N(S)) \geq |S|$, for all $S \subseteq X$. Many variants and applications of Rado's Theorem can be found in Welsh (1976).

Pfaffians

Let $D := (V, A)$ be a strict digraph, and let $\{x_a : a \in A\}$ be a set of variables associated with the arcs of D. The *Tutte matrix* of D is the $n \times n$ skew-symmetric matrix $\mathbf{T} = (t_{uv})$ defined by:

$$t_{uv} := \begin{cases} 0 & \text{if } u \text{ and } v \text{ are not adjacent in } D, \\ x_a & \text{if } a = (u, v), \\ -x_a & \text{if } a = (v, u). \end{cases}$$

Because $\mathbf{T}$ is skew-symmetric, its determinant is zero when n is odd. But when n is even, say $n = 2k$, the determinant of $\mathbf{T}$ is the square of a certain polynomial, called the Pfaffian of $\mathbf{T}$, which may be defined as follows.

For any perfect matching $M := \{a_1, a_2, \ldots, a_k\}$ of D, where $a_i := (u_i, v_i)$, $1 \leq i \leq k$, let $\pi(M)$ denote the product $t_{u_1v_1}t_{u_2v_2}\ldots t_{u_kv_k}$ and let $\mathrm{sgn}(M)$ denote the sign of the permutation $(u_1v_1u_2v_2\ldots u_kv_k)$. (Observe that $\mathrm{sgn}(M)$ does not depend on the order in which the elements of M are listed.) The *Pfaffian* of $\mathbf{T}$ is the sum of $\mathrm{sgn}(M)\pi(M)$ taken over all perfect matchings M of D.

Now, a polynomial in indeterminates is zero if and only if it is identically zero. Thus the digraph D has a perfect matching if and only if the Pfaffian of $\mathbf{T}$ is nonzero. Because the determinant of $\mathbf{T}$ is the square of its Pfaffian, it follows that D has a perfect matching if and only if $\det \mathbf{T} \neq 0$. Tutte's original proof of Theorem 16.13 was based on an ingenious exploitation of this fact (see Tutte (1998) for a delightful account of how he was led to this discovery). In more recent times, properties of the Tutte matrix have played surprisingly useful roles both in the theory of graphs and in its algorithmic applications; see, for example, Lovász and Plummer (1986), McCuaig (2000), and Robertson et al. (1999).