DYNAMIC MATCHINGS AND QUASI-DYNAMIC FRACTIONAL MATCHINGS, PART II

James B. Orlin*

July 1982

Working Paper #1332-82

*Work supported in part under ECS-8205022, National Science Foundation, entitled "Research Initiation: Dynamic/Periodic Optimization Models".

Abstract

Consider a directed graph G in which every edge has an associated real-valued distance and a real-valued weight. The weight of an undirected circuit of G is the sum of the weights of the edges, whereas the distance of an undirected circuit is the sum of the distances of the forward edges of the circuit minus the sum of the distances of the backward edges. A trivial circuit is a 2-edge circuit in which one edge of G appears twice on the circuit. A quasi-dynamic fractional matching (or Q-matching) is a collection of vertex-disjoint circuits such that each circuit is either trivial or else it is an odd circuit whose distance is non-zero. The Q-matching problem is to find a Q-matching that maximizes the sum of the weights of its circuits.

The Q-matching problem generalizes both the matching problem and the fractional matching problem. Moreover, the dynamic matching problem, which is a matching problem on an infinite dynamic (time-expanded) graph, is linearly transformable to the Q-matching problem, as shown in part I of this paper.

In this paper we solve the Q-matching problem by generalizing Edmonds' blossom algorithm. In fact, all of the major components of the blossom algorithm -- including alternating trees, augmentations, shrinking, and expanding -- are appropriately generalized to yield a running time that is proportional to that for the weighted matching problem. Furthermore, if all edge distances are equal to zero, this new algorithm reduces to the blossom algorithm.

1. Introduction

Consider a directed graph G = (V(G), E(G)) in which each edge $e \in E(G)$ has an associated distance d(e) and an associated weight w(e). The quasi-dynamic fractional matching problem is to find a maximum weight fractional matching subject to the additional proviso that all of the odd circuits in the fractional matching have a non-zero distance. This problem which we shall refer to more briefly as the Q-matching problem was introduced in part I of this paper so as to solve the dynamic matching problem. The reader should refer to part I of this paper for the necessary graph theoretic terminology.

The Q-matching problem generalizes both the ordinary matching problem and the fractional matching problem. The ordinary matching problem is the special case of the Q-matching problem in which all edge distances are zero. The fractional matching problem is equivalent to the special case of the Q-matching problem in which all edge distances are one.

The Q-matching problem is a special case of the weighted F-packing problem as proposed by Cornuejols et. al. [1982] in which the class F consists of edges of G and some family of hypomatchable graphs. Cornuejols et. al. showed how to solve in polynomial time a large subclass of weighted F-packing problems, and our algorithm for unweighted Q-matchings may be viewed as a (speeded-up) implementation of their approach. (However, the two algorithms were developed independently.)

2. Graph Theoretical Preliminaries

In part I of this paper we defined certain basic concepts such as paths, circuits, distance of a path, neutral and aneutral circuits, etc. Below we introduce some other terminology that we use later to describe the Q-matching algorithm. Many of the definitions are common in the matching literature and can be found for example in Edmonds [1965a]. Moreover, in this section on terminology, many of the definitions are taken directly from Cunningham and Marsh [1978].

Suppose $P = v_0, e_1, v_1, \ldots, e_k, v_k$ is a path in the directed graph G. We say that e_j is an odd (resp., even) edge of P if j is odd (resp., even). We let $E^o(P)$ denote the odd edges of P and $E^e(P)$ denote the even edges of P. A vertex, v_j, is odd (resp., even) if j is odd (resp., even).

A Q-matching is an ordered pair (M, Q), where M is a set of vertex disjoint edges, and Q is a set of vertex disjoint odd aneutral circuits, and $V(M) \cap V(Q) = \phi$. The cardinality of the Q-matching (M,Q) is $|V(M)| + |V(Q)|$.

The symmetric difference of the two subgraphs H and J of the digraph G is the subgraph K (denoted $H \oplus J$) such that $V(K) = V(H) \cup V(J)$ and $E(K) = (E(H) - E(J)) \cup (E(J) - E(H))$. An (M,Q)-alternating path is a path in which the edges are alternately in E(M) and E(G) - E(M) - E(Q) (and thus no edges are in E(Q)). A vertex v is called (M,Q)-exposed if $v \notin V(M) \cup V(Q)$.

A tree is a connected circuitless subgraph T (we do not assume that V(T) = V(G)). An (M,Q)-alternating tree is a rooted tree in which the root is exposed and such that the path from the root to any other vertex of V(T) is an (M,Q)-alternating path. A vertex of a rooted tree is odd (resp., even) if the vertex is an odd (resp., even) vertex on the path in T from the root. We denote the set of odd (resp., even) vertices of T as $V^o(T)$, (resp., $V^e(T)$). (This definition agrees with that of Cunningham and Marsh [1978], and differs slightly from Edmonds [1965a].) An alternating tree is illustrated in Figures 3.1 and 3.2.

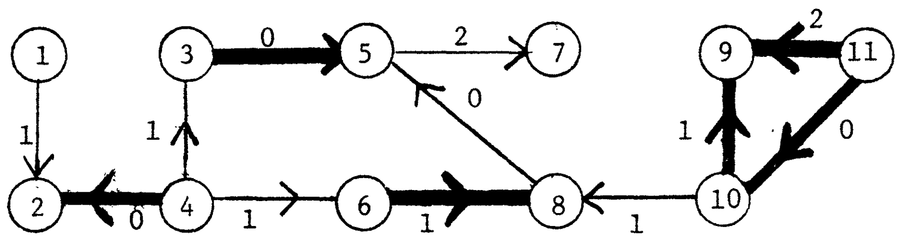


Figure 3.1 A directed graph with a Q-matching in boldface.

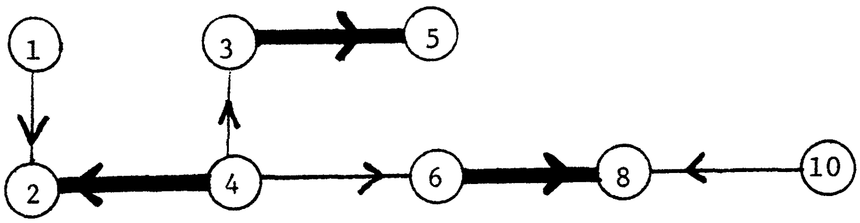


Figure 3.2 An (M,Q)-alternating tree with root vertex 1. The even and odd vertices are $V^e(T) = \{1,4,5,8\}$ and $V^o(T) = \{2,3,6,10\}$.

An Overview

In the sequel we will modify Edmonds' blossom alborithm so that it can solve the Q-matching problem. In particular, we will modify the concepts of (1) augmenting paths, (2) shrinking and expanding subgraphs, (3) the linear programming formulation, i.e. the construction of the polytope describing the convex hull of all Q-matchings, and (4) the primal-dual algorithm.

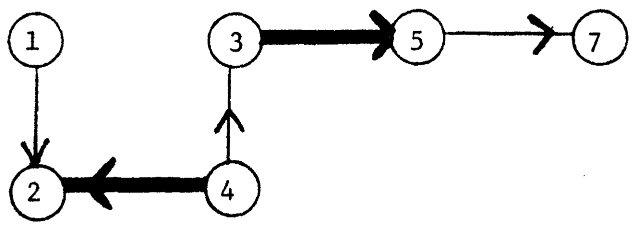
Just as the Q-matching problem properly generalizes the matching problem, so does much of the polyhedral and combinatorial theory for matchings extend to Q-matchings. In particular, we extend Edmonds' [1965 generalization of Tutte's [1947] theorem on the existence of matchings. We also extend Edmonds' [1965b] characterization of the convex hull of the set of matchings. Of course, we cannot include in this paper all possible extensions from matchings to Q-matchings. For example, we have omitted extensions of the elegant results of Pulleyblank [1973] and Pulleyblank and Edmonds [1974] concerning facets of the matching polyhedra. We have also restricted attention to only the primal-dual algorithm for the weighted Q-matching problem, and thus we omit extensions of other very elegant results including the primal algorithm developed by Cunningham and Marsh [1978].

The unweighted version of the Q-matching problem is properly viewed as a special case of some recent (independently developed) results by Cornuejols et. al. [1982]. By appropriately modifying and extending earlier results by Cornuejols and Pulleyblank [1980a and b] and [1982] concerning triangle-free two matchings, Cornuejols et. al. showed how to solve in polynomial time a large subclass of F-packing problems. (An F-packing problem is the problem of covering the vertices of a graph with a collection of vertex disjoint edges and circuits, where the circuits must satisfy a given property F.) Moreover, the unweighted Q-matching algorithm presented below may be viewed as a specialization of their techniques for solving F-packings. Also, the characterization of graphs possessing a perfect Q-matching (which is a special case of Q-matching duality theorem below) is a specialization of the characterization of graphs possessing perfect F-packings as given in Cornuejols et. al.

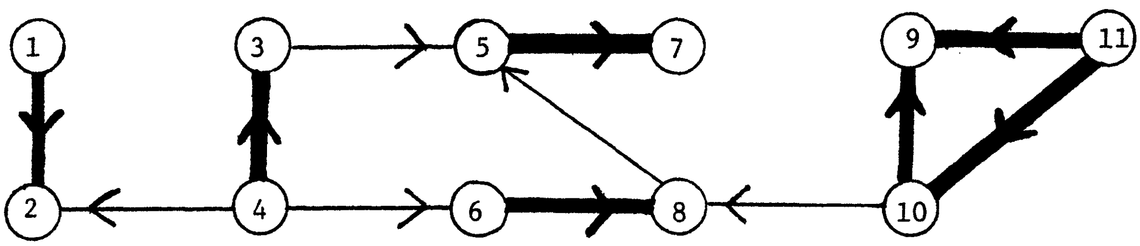
Our algorithms differ from those of Cornuejols et. al. in two important respects. First, our algorithm uses data structures that exploit the edge distances so as to speed up the algorithm significantly. Secondly, and more importantly, we can solve the weighted Q-matching problem in polynomial time, whereas no polynomial algorithm is known for the weighted F-packing problem. Thus in some significant sense, the Q-matching problem seems to be much easier to solve than the F-packing problem.

3. (M,Q)-Augmenting Paths

Below we define three types of paths that lead to augmentations in Q-matchings. A type-1 (M,Q)-augmenting path is an (M,Q)-alternating path P initiating and terminating at distinct exposed vertices. To augment along a type-1 (M,Q)-augmenting path P is to replace (M,Q) by (M',Q) with $M' = M \oplus P$. A type-1 augmentation is portrayed in Figure 3.3 and 3.4. All of the Figures below have the Q-matching in boldface lines. The vertex numbers refer to edge distances. We omit the edge weights and also the edge distances when they are not needed.



<u>Figure 3.3.</u> A type-1 (M,Q)-augmenting path for the directed graph of Figure 3.1.



<u>Figure 3.4</u> The Q-matching after augmentation along the path in Figure 3.3.

A type-2 (M,Q)-augmenting path is an (M,Q)-alternating path P whose initial vertex is an exposed vertex, and whose terminal vertex is a vertex of V(Q). Suppose P is a type-2 path with terminus v, and assume that v is the initial and terminal vertex of circuit $C \in Q$. To augment along P is to replace (M,Q) by the Q-matching (M',Q') where $M' = (M \oplus P) \cup E^e(C)$, and $Q' = Q - \{C\}$. A type-2 augmentation is portrayed in Figures 3.5 and 3.6.

An (M,Q) blossom is an ordered pair (P,C) where P is an (M,Q)-alternating path whose terminal edge is in E(M) and C is an odd (M,Q)-alternating circuit whose initial vertex is the terminus of P. An (M,Q)-augmenting blossom (or a type-3 (M,Q)-augmenting path) is an (M,Q)-blossom (P,C) such that P is an even path whose initial vertex is exposed and such that C is aneutral. To augment along (P,C) is to replace (M,Q) by (M',Q') where $M' = (M \oplus P) - C$ and $Q' = Q \cup \{C\}$. An augmentation along a blossom is portrayed in Figures 3.7 and 3.8.

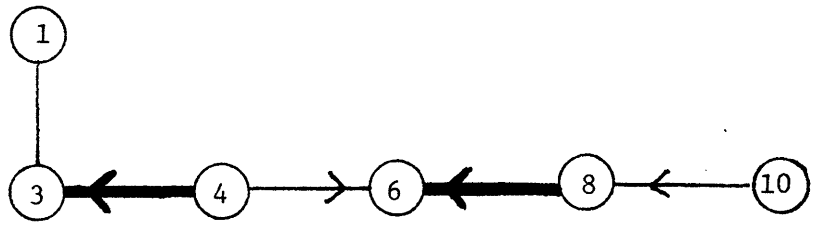


Figure 3.5. A type-2 (M,Q)-augmenting path for the directed graph of Figure 3.1.

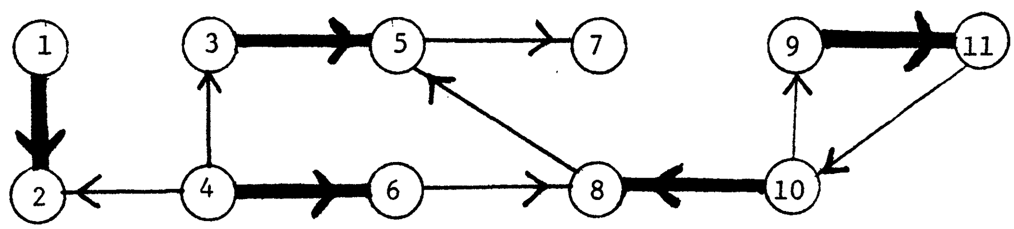


Figure 3.6. The Q-matching after augmentation along the path in Figure 3.5.

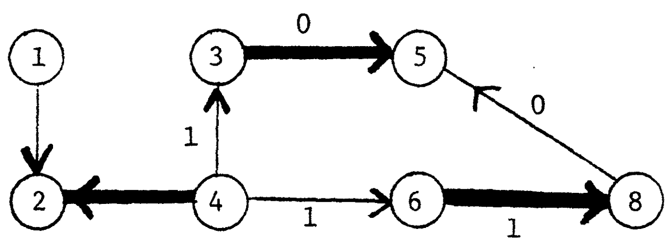


Figure 3.7. An (M,Q)-augmenting blossom for the directed graph of Figure 3.1. (The circuit of the blossom has length ± 1 and is thus aneutral).

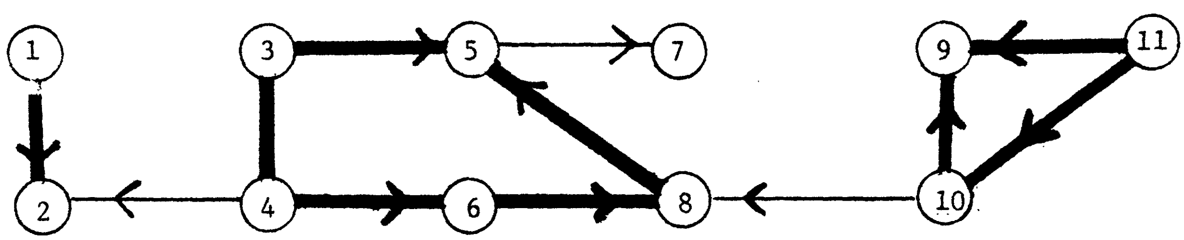


Figure 3.8. The Q-matching after augmentation along the blossom in Figure 3.7

We will refer to any of the three types of augmenting paths described above as (M,Q)-augmenting paths, although the third type is not a simple path. We note one interesting distinction between augmenting paths for Q-matchings and augmenting paths for matchings.

<u>Observation</u>. If there is no (M,Q)-augmenting path, then it is not necessarily true that (M,Q) is a maximum cardinality Q-matching.

The above observation is illustrated in Figure 3.9. There we define a Q-matching of cardinality 4, even though the circuit with vertices 1, 2, 3, 4 and 5 has cardinality 5. This result contrasts with Berge's [1957] theorem stating the existence of an augmenting path for any matching problem in which a given matching does not have maximum cardinality.

Nevertheless the Q-matching algorithm determines maximum cardinality Q-matchings via augmentations. In order to locate augmenting paths, the algorithm performs a sequence of contractions and expansions of the directed graph. It is during the contraction procedure (described in the next section) that augmenting paths may be created.

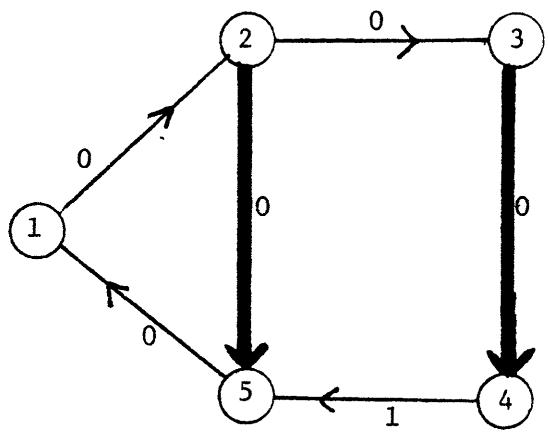


Figure 3.9. A Q-matching that is not maximum cardinality and relative to which there is no augmenting path.

4. Shrinking Subgraphs

Our Q-matching algorithm parallels Edmonds blossom algorithm in its reliance on shrinking subgraphs. Below we define some more graph theoretic terminology related to these shrinkings. These definitions, however, differ slightly from the common usage as given for example in Cunningham and Marsh [1978]. First, the Q-matching algorithm only shrinks neutral subgraphs, i.e., those containing no aneutral circuits. Moreover, the distances on "shrunk edges" are changed appropriately. Secondly, we assume that each subgraph to be shrunk is both rooted and connected.

Let H be a rooted, connected, neutral subgraph of the directed graph G, and let v' be the root of H. The <u>root function</u> in G induced by H is the function $r_H(\cdot)$ defined as follows:

$$r_H(v) = \begin{cases} v' & \text{if } v \in V(H) \\ v & \text{if } v \notin V(H) \end{cases}$$

The <u>distance function</u> in G induced by H is the function $d_H(v)$ defined as follows:

$$d_H(v) = \begin{cases} \text{the distance in H from } v' \text{ to } v & \text{if } v \in V(H) \\ 0 & \text{if } v \notin V(H) \end{cases}$$

The distance function is uniquely defined since H is connected and neutral. The distance function $d_H(v)$ is equivalently defined to be the distance from $r_H(v)$ to v using only edges of E(H).

Let H be a rooted, connected, neutral subgraph of the directed graph G and let v' be the root of H. The graph G × H obtained from G by shrinking H is defined as follows: $V(G \times H) = V(G) - V(H) \cup \{v'\}$. The edge set $E(G \times H)$ is constructed as follows: for each edge $e = (u,v) \in E(G) - E(H)$ there is an associated edge $e' \in E(G \times H)$ denoted $e' = e \times H$ and such that:

(1) $e' = (r_H(u), r_H(v))$

and (2) $d(e') = d(e) + d_H(u) - d_H(v)$.

For each subset $H' \subseteq G$, we let $H' \times H$ denote the subgraph of $G \times H$ induced by H'. In particular $E(H' \times H)$ is the set of all edges in $E(G \times H)$ induced by edges in $E(H') - E(H)$.

The shrinking of a subgraph is illustrated in Figures 4.1 and 4.2. Figure 4.1 portrays a directed graph in which the neutral connected rooted subgraph is in boldface, and vertex 6 is the root. The shrunk graph is portrayed in Figure 4.2. The distance and root functions are described in Table 4.1, and the changes in the edges are portrayed in Table 4.2.

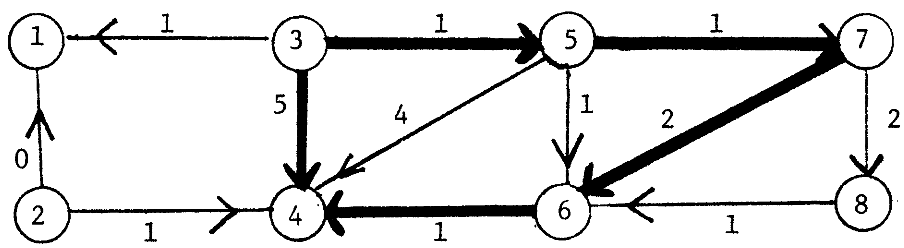


Figure 4.1. A directed graph in which the boldface subgraph is a neutral subgraph with root vertex 6. However, the induced, subgraph of the vertices on the boldface contains the aneutral circuit through vertices 5, 6 and 7.

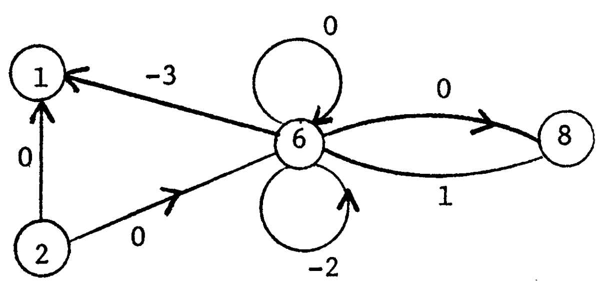


Figure 4.2. The directed graph obtained by shrinking the rooted neutral subgraph of Figure 4.2. The loop with distance-2 corresponds to edge (5,6)of Figure 4.1.

Vertex	1	2	3	4	5	6	7	8
$r_H(.)$	1	2	6	6	6	6	6	8
$d_H(.)$	0	0	-4	1	-3	0	-2	0

Table 4.1

Edge of G	(2,1)	(2,4)	(3,1)	(5,4)	(5,6)	(7,8)	(8,6)
Distance in G	0	1	1	4	1	2	1
Edge of G x H	(2,1)	(2,6)	(6,1)	(6,6)	(6,6)	(6,8)	(8,6)
Distance in G x H	0	0	-3	0	-2	0	1

Table 4.2

The definition of edge distances is designed so that the following lemma and its corollary are true.

Lemma 1. Let G be a directed graph, and suppose that H is a connected neutral rooted subgraph of G. Let $d_H(.)$ and $r_H(.)$ be the distance and root functions induced by H. Finally suppose that P is a path in G and let u and v be the initial and terminal vertices of P. Then $P \times H$ is a path in $G \times H$ from $r_H(u)$ to $r_H(v)$. Moreover, the distance of the path $P \times H$ is $d(P) + d_H(u) - d_H(v)$.

Proof. The lemma is easily verified if P consists of a single edge. Suppose instead that $P = v_0,e_1,v_1,\ldots,e_k,v_k$ and $k \geq 2$. Assume inducductively that the lemma is valid for all paths of at most k-1 edges. Let $P_1 = v_0,e_1,v_1$ and let $P_2 = v_1,\ldots,e_k,v_k$. Then $P_1 \times H$ is a path from $r_H(v_0)$ to $r_H(v_1)$ with distance $d(P_1) + d_H(v_0) - d_H(v_1)$, and by the inductive hypothesis $P_2 \times H$ is a path from $r_H(v_1)$ to $r_H(v_k)$ with distance $d(P_2) + d_H(v_1) - d_H(v_k)$. Concatening $P_1 \times H$ and $P_2 \times H$ gives the path $P \times H$ from $r_H(v_0)$ to $r_H(v_k)$ with distance $d(P_1) + d(P_2) + d_H(v_0) - d_H(v_k)$, completing the proof.

Corollary 1. Let H, G, $d_H(\cdot)$, and $r_H(\cdot)$ satisfy the conditions of Lemma 1, and suppose that C is any circuit of G. If $C \times H$ is neutral, then circuit C is neutral, Moreover, if $C \times H$ is an aneutral circuit, then C is also aneutral.

Proof. We first note that $C \times H$ is a closed walk and is thus the union of edge-disjoint circuits. Since $d(C) = d(C \times H)$, it follows that if all circuits of $C \times H$ are neutral, then $d(C \times H) = d(C) = 0$.

... ...d rooted subgraphs of G with maximal members $H_1,\ldots,H_k$, the graph $G \times \mathcal{F}$ is defined to be $(\ldots((G \times H_1) \times H_2) \times\ldots\times H_k)$. It is easily verified that the graph $G \times \mathcal{F}$ does not depend on the order in which these sets are shrunk. The roots of the maximal members of $\mathcal{F}$ are called pseudo-verticies of $G \times \mathcal{F}$. The other verties of $V(G \times \mathcal{F})$ are the real verticies of $G \times \mathcal{F}$.

For a nested family $\mathcal{F}$ and for each $H \in \mathcal{F}$, let $\mathcal{F}[H] = \{H' \in \mathcal{F}: \ E(H') \subset E(H)\}$. Thus $\mathcal{F}[H]$ is the collection of those subsets of $\mathcal{F}$ that are strictly contained in H.

The nested families $\mathcal{F}$ that we are interested in all have the foll...ing property:

For each $H \in \mathcal{F}$, $H \times \mathcal{F}[H]$ is an odd neutral circuit. (5.1)

A nested family $\mathcal{F}$ satisfying (5.1) is called a shrinking family. The definition here of shrinking families is a direct extension of the usual definition for undirected graphs (see Cunningham and Marsh [1978] for example). The only significant modification of the definition stems from our concerns with edge distances.

The motivation underlying the construction of shrinking families is given in Theorem 1 as proved in the next subsection. The result parallels and generalizes the following result proved by Cunningham and Marsh. (We note that a shrinking family for an undirected graph is the same as a shrinking family for a directed graph in which all edge distances are 0.)

Theorem (Cunningham and Marsh). Let $\mathcal{F}$ be a shrinking family of G and let M be a perfect matching (i.e., M has no exposed vertices) of $G \times \mathcal{F}$. Then there is a perfect matching M_1 of G such that $M = M_1 \times \mathcal{F}$.

5. Expanding Pseudo-Vertices

The Q-matching algorithm that we develop below involves a sequence of shrinkings of odd neutral circuits resulting in a shrinking family. The advantage of this shrinking family $\mathfrak{F}$ may be stated briefly as follows: an (M,Q)-augmenting path in $G \times \mathfrak{F}$ leads to an augmentation in G. This result is stated more formally in Theorem 1.

Theorem 1. Let $\mathfrak{F}$ be a shrinking family of G and let (M,Q) be a Q-matching of $G \times \mathfrak{F}$. Then there is a Q-matching (M*,Q*) of G such that $(M,Q) = (M^*, Q^*) \times \mathfrak{F}$ and the number of (M,Q)-exposed vertices and pseudo-vertices in $G \times \mathfrak{F}$ is equal to the number of (M*,Q*)-exposed vertices in G.

Proof. The result is trivially true if $\mathfrak{F} = \phi$. Suppose $\mathfrak{F} = \phi$, and assume inductively that the theorem is valid for all shrinking families with fewer elements than $\mathfrak{F}$.

Let H be a minimal element of $\mathfrak{F}$. Let $G' = G \times H$, and let $\mathfrak{F}' = \mathfrak{F} - \{H\}$. Because $G' \times \mathfrak{F}' = G \times \mathfrak{F}$ and by our inductive hypothesis, there is a Q-matching (M',Q') in G' such that $(M,Q) = (M',Q') \times \mathfrak{F}'$ and the number of (M,Q)-exposed vertices and pseudo-vertices in $G \times \mathfrak{F}$ is equal to the number of (M',Q')-exposed vertices and pseudo-vertices in G'.

Because H is minimal, it follows that H is spanned by an odd neutral circuit C. Let v' denote the pseudo-vertex of G' corresponding to the root of H. In order to obtain the Q-matching (M*,Q*) with the properties stated in the theorem, we consider separately the cases in which v' is (M',Q')-exposed or $v' \varepsilon V(M')$ or $v' \varepsilon V(Q')$.

Suppose first that v' is (M',Q')-exposed. Then let $M = M' \cup E^e(H)$ and let $Q^* = Q'$.

Suppose next that v' is an endpoint of edge $e' \varepsilon E(M')$. Let e be the edge of E(G) such that $e' = e \times H$, and let u be the endpoint of e that is a vert. of $V(H)$. Finally suppose that C is expressed as a path from vertex u to vertex u. Then let $M^* = (M' + e-e') \cup E^e(C)$ and let $Q^* = Q'$. (This expansion is portrayed in Figures 5.1 and 5.2. It is an expansion that is also used in the blossom algorithm for the matching problem.)

Lastly, we assume that v' is a vertex of some aneutral circuit $C' \varepsilon Q$, and assume that C' is expressed as a path from vertex v' to vertex v'. Let P_1 be the (unique) path in G with $|E(C')|$ edges such that $C' = P_1 \times H$. (It is easy to prove inductively that such a path must exist.) Let P_2 and P_3 be the two paths in H from the terminal vertex of P_1 to the initial vertex of P_1, such that $E(P_2) + E(P_3) = E(H)$ and chosen so that (P_1,P_2) is an odd circuit in G. Then let $M^* = M' \cup E^e(P_3)$, and let

$Q^* = (Q' - C') \cup \{P_1,P_2\}$ because C' is aneutral and by Corollary ., (P_1,P_2) is an aneutral circuit of G. Thus (M^*,Q^*) is a Q-matching of G. This expansion is portrayed in Figures 5.3-5.6.

In all three cases we have constructed a Q-matching (M^*,Q^*) such that $(M^*,Q^*) \times H = (M',Q')$ and the number of (M^*,Q^*)-exposed vertices in G is equal to the number of (M',Q')-exposed vertices and pseudo-vertices in G', completing the proof.

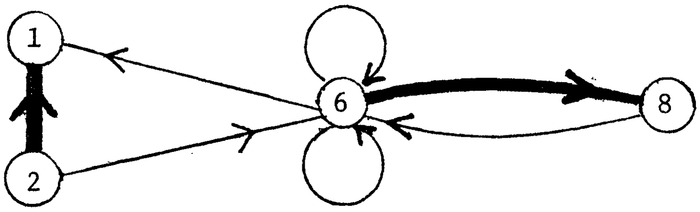


Figure 5.1. A Q-matching for the shrunken graph portrayed in Figure 4.2.

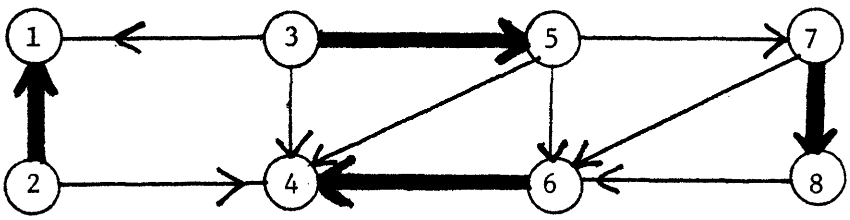


Figure 5.2. The Q-matching for the graph portrayed in Figure 4.1 obtained by expanding the Q-matching of Figure 5.1. Note that edge (7,8) had been shrunk to edge (6,8) of Figure 5.1.

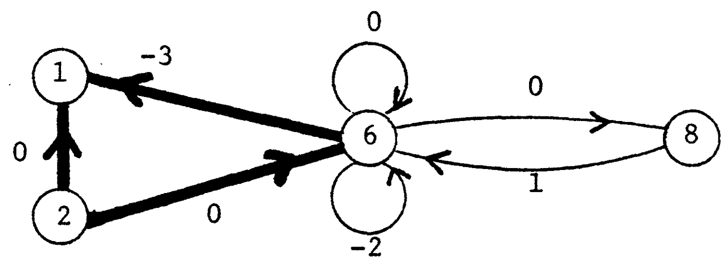


Figure 5.3. A Q-matching for the shrunken graph portrayed in Figure 4.2. Here the pseudo-vertex is in the Q-matched circuit 6,1,2,6.

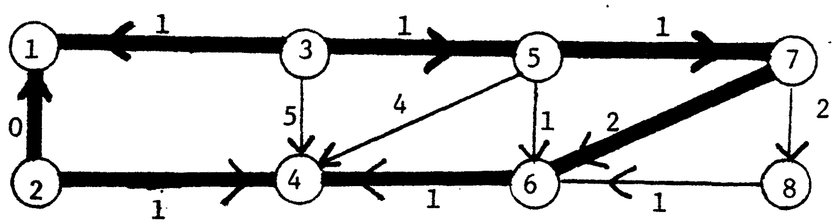


Figure 5.4. The Q-matching obtained by expanding the Q-matching of Figure 5.3. The Q-matched circuit C of Figure 5.3 corresponds to the path $P_1 = 3,1,2,4$. The paths P_2,P_3 of the proof of Theorem 1 are $P_2 = 4,6,7,5,3$ and $P_3 = 4,3$, chosen so that P_1,P_2 is an odd aneutral circuit.

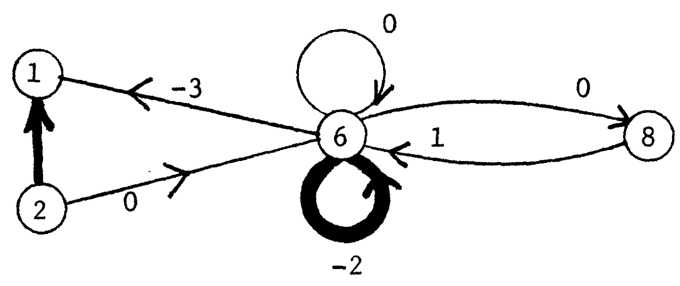


Figure 5.5. A Q-matching for the shrunken graph portrayed in Figure 4.2.

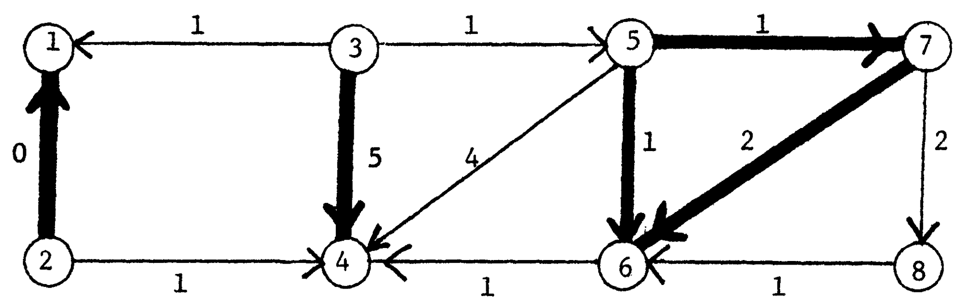


Figure 5.6. The Q-matching obtained by expanding the Q-matching of Figure 5.5. (Note that the Q-matched circuit here has the same distance as the Q-matched circuit of Figure 5.5.)

Henceforth, by expanding a pseudo-vertex we mean recovering the odd neutral circuit from which it was obtained while simultaneously altering the Q-matching as in Theorem 1 so as to leave unaltered the number of exposed vertices and pseudo-vertices.

6. The Cardinality Q-Matching Algorithm

The following description of the maximum cardinality Q-matching algorithm closely parallels the description given by Lawler [1976].

STEP 0. Let G be a directed graph. Let $M = Q = \phi$. Let $\mathfrak{F} = \phi$.

STEP 1. (Labeling and Scanning).

(1.0) Let the alternating rooted tree be empty. All vertices are "unlabeled" and all edges "unscanned".

(1.1) Find an unlabelled exposed vertex; label it even and start rooting a tree at this vertex. If no exposed vertex exists, go to Step 4.

(1.2) Find an unscanned even vertex v to T incident to at least one unscanned edge. If no such vertex exists, go to Step (1.1). Else continue.

(1.3) Find an unscanned edge e incident to the vertex v. If no such edge exists then label v "scanned" and return to 1.2. Else let v' be the other endpoint of edge e. Label e "scanned" and continue.

(1.3a) If v' is an even vertex of T then go to Step 3.

(1.3b) If v' is an odd vertex of T then return to 1.3.

(1.3c) If v' is exposed, then go to Step 2.

(1.3d) If $v' \in V(Q)$, then go to Step 2.

(1.3e) If $v' \in V(M)$ and v' is unlabeled, then continue.

(1.4) Add edge e to the alternating tree. Find the unique edge e' of E(M) that is incident to v' and add edge e' to the tree T. Label e' "scanned", label v' odd and label the other endpoint of e' even. Then return to Step 1.3.

STEP 2. (Type 1 and Type 2 augmentations).
Find the type 1 or type 2 augmenting path in $T \cup \{e\}$. Augment along this path. Recursively expand all pseudo-vertices and return to Step 1.0.

STEP 3. (Shrinking or Type 3-augmentations).
Determine the odd circuit C in $T \cup \{e\}$. If C is neutral, then replace G, (M,Q), and T by $G \times C$, $(M,Q) \times C$, $T \times C$ and return to Step 1.2. If C is aneutral, then perform the type-3 augmentation; recursively expand all pseudo-vertices and return to Step 1.2.

STEP 4. (Hungarian Forest)
Each even vertex and pseudo-vertex is joined only to odd vertices. Expand all pseudo-vertices. The resulting Q-matching has maximum cardinality.

We note that the above algorithm determines a Q-matching in a polynomial number of steps. Below we prove that the Q-matching obtained by the algorithm does indeed have maximum cardinality. Simultaneously, we prove a duality result that extends Edmonds' "odd cover theorem" [1965a].

A set consisting of a single vertex v is said to <u>cover</u> $\delta(v)$, and the <u>capacity</u> of $\{v\}$ is 2. A subset $S \subseteq V(G)$ with $|S| \geq 2$ is said to <u>cover</u> the edges of G[S]. If $|S| \geq 3$ and is odd and if G[S] is neutral, then the capacity of S is $|S|-1$. Otherwise the <u>capacity</u> of S is $|S|$. A <u>set cover</u> is a family of subsets of V(G) such that each edge $e \in E(G)$ is covered by at least one member of the family.

Q-Matching Duality Theorem. The maximum cardinality of a Q-matching of G is equal to the minimum capacity of a set cover of G.

Proof. We see that the maximum cardinality of a Q-matching is at most the minimum capacity of a set cover as follows. Suppose that (M,Q) is a Q-matching. Let the cardinality of each edge $e \in E(M)$ be two and the cardinality of each edge $e \in E(Q)$ be one. Then the sum of the cardinality of the edges is the cardinality of the Q-matching. Moreover, the sum of the cardinalities of the edges is at most the capacity of any set cover.

Conversely, let $\mathcal{F}$ denote the shrinking family obtained prior to the last step, let T be the "hungarian" tree in $G \times \mathcal{F}$ obtained prior to the last step, and let (M,Q) be the Q-matching prior to expansion. Since each pseudo-vertex of $G \times \mathcal{F}$ is even (as is the case with the usual cardinality matching algorithms), it follows that all odd vertices of $G \times \mathcal{F}$ are real. Finally, we construct a set cover S as follows:

(1) For each odd vertex v of $G \times \mathcal{F}$, there is a singleton set $\{v\}$ in S.

(2) For each pseudo-vertex of $G \times \mathcal{F}$ corresponding to a maximal subset $H \in \mathcal{F}$, there is the set V(H) in S.

(3) There is a subset consisting of all vertices in $V(M) \cup V(Q)$ that are left unlabeled by the algorithms. (In fact, the set always contains all vertices of the matched circuits of Q.)

We first see that the collection S is a set cover as follows. Suppose edge e = (u,v) is not covered by S. If e is not an edge of $G \times \mathcal{F}$, then e is covered by a set defined in (2). If u or v is an odd vertex of T, then e is covered. If neither u nor v is labeled, then e is covered by the set defined in (3). Thus u or v is an even vertex of T(or both). If u (resp., v) is even, then v (resp., u) is labeled by the algorithm. Hence both u and v are even and members of different maximal subsets of $\mathcal{F}$. But then scanning edge e would have led in Step 3 either to another shrinking or another expansion, contradicting that edge e is not covered.

Finally, we see that the cardinality of the Q-matching obtained by the algorithm by expanding (M,Q) is equal to the capacity of S by observing that the capacity of each subset $T \in S$ is equal to the sum of the cardinalities of the edges of (M,Q) that it covers. (Each set T constructed from a pseudo-vertex is odd and G[T] is neutral. Thus its capacity is $|T|-1$, and it covers $(|T|-1)/2$ edges of M).

We note that the optimality of the Q-matching algorithm is implicit in the proof of the Q-matching duality theorem.

7. Maximum Weight Q-Matchings

Below we give a polynomial algorithm for finding a maximum weight Q-matching. The algorithm is a direct extension of Edmonds' [1965b] primal-dual algorithm for the maximum weight matching problem. However, the presentation of the algorithm parallels that of Lawler [1976].

In order to simplify the notation, we let $G = (V(G), E(G))$ be a directed graph in which $V(G) = \{1,\ldots,n\}$, $E(G) = \{e_1,\ldots,e_m\}$, the distance of edge e_i is d_i, and $w_i = 2w(e_i)$. (We need the factor of 2 in order to be consistent in our definition of the weight of a Q-matching.)

Let $\mathbb{N}$ be the set of all odd neutral subgraphs of G. Let $x = (x_j)$ be an assignment of the values in {0, 1/2, 1} to the edges in E(G) so as to

$$\text{Maximize} \quad \sum_{j=1}^{m} w_j x_j \qquad (6.1a)$$

$$\text{Subject to} \quad \sum_{e_j \in \delta(V)} x_j \leq 1 \qquad \text{for all vertices } v \in V(G) \qquad (6.1b)$$

$$\sum_{e_j \in E(H)} x_j \leq 1/2(|V(H)|-1) \qquad \text{for all } H \in \mathbb{N} \qquad (6.1c)$$

$$x_j \in \{0, 1/2, 1\} \qquad \text{for } j = 1,\ldots,m. \qquad (6.1d)$$

The Q-matching (M,Q) induces a feasible solution $x = (x_j)$ for (6.1) given by: $x_i = 1$ for $e_i \in E(M)$, and $x_i = 1/2$ for $e_i \in E(Q)$. We note that a Q-matching (M,Q) has maximum weight, if and only if the induced solution x is optimal for (6.1). Moreover, the weight of (M,Q) is wx. In fact, we prove a much stronger result below. The problem obtained from (6.1) by relaxing constraint (6.1d) and adding the constraint "$x \geq 0$" is called the continuous relaxation of (6.1).

Theorem 2. A Q-matching (M,Q) has maximum weight if and only if the induced solution x is optimal for the continuous relaxation of (6.1).

The proof of Theorem 2 follows the presentation of the maximum weight Q-matching algorithm.

Theorem 2 suggests the following variant of the weighted matching algorithm, and both may be viewed as a special case of the primal-dual linear programming algorithm presented, for example, by Papadimitriou and Steiglitz (1982). First, form the dual to linear program (6.1) by associating dual prices λ and σ with constraints (6.1b) and (6.1c) as below.

$$\text{Minimize} \quad \sum_{v \in V(G)} \lambda_v + \sum_{H \in \mathcal{N}} (1/2)\sigma_H(|V_H|-1)$$

$$\text{Subject to} \quad \lambda_u + \lambda_v + \sum_{H: e_j \in E(H)} \sigma_H \geq w_j \qquad \text{for all } e_j = (u,v) \in E(G) \qquad (6.2)$$

$$\text{and} \quad \lambda, \sigma \geq 0$$

As is well known, a necessary and sufficient condition for feasible solution x for the continuous relaxation of (6.1) to be optimal is that there exists feasible dual vectors λ, σ for (6.2) such that for all u,v and H the following complementary slackness conditions (6.3) are satisfied.

If $e_j = (u,v)$ and $x_j > 0$ then

$$\lambda_u + \lambda_v + \sum_{H:e_j \varepsilon E(H)} \sigma_H = w_j \qquad (6.3a)$$

$$\text{If } \lambda_v \geq 0 \text{ then } \sum_{e_j \varepsilon \delta(v)} x_j = 1 \quad \text{for all } v \varepsilon V(G) \qquad (6.3b)$$

$$\text{If } \sigma_H > 0 \text{ then } \sum_{e_j \varepsilon E(H)} x_j = (1/2)(|V(H)|-1) \qquad (6.3c)$$

Below we let $\overline{w}_j$ denote the reduced weight of edge $e_j = (u,v)$, i.e., $\overline{w}_j = w_j - \lambda_u - \lambda_v - (\sum \sigma_H : e_j \varepsilon E(H))$. An edge e_j is feasible with respect to (λ,σ) if $\overline{w}_j = 0$. The feasible subgraph with respect to (λ,σ) is the subgraph H such that E(H) is the set of all feasible edges.

Suppose $\mathcal{F}$ is a shrinking family and T is a rooted alternating tree in $G \times \mathcal{F}$. We say that $H \varepsilon \mathcal{F}$ is even (resp., odd) if H is a maximal element of $\mathcal{F}$ that is shrunk to an even (resp., odd) pseudo-vertex of T. We say that a vertex $v \varepsilon V(G)$ is even (resp., odd) if either v is a real even (resp., odd) vertex of T or else v is a vertex of some even (resp., odd) subgraph $H \varepsilon \mathcal{F}$.

The Maximum Weight Q-Matching Algorithm

STEP 0. Let $\lambda_v = 1/2 \max_j \{w_j\}$ for each $v \varepsilon V(G)$.
Let $\sigma = 0$, and let $M = Q = \phi$. Let $\mathcal{F} = \phi$, and let G' be the feasible subgraph of G with respect to (λ,σ).

STEP 1. Carry out Step 1 of G' as in the cardinality Q-Matching Algorithm on subgraph G'. If G' has no exposed vertices, then go to Step 3.

(1.1) If an (M,Q)-augmenting path is determined, then perform the augmentation, expand all pseudo-vertices corresponding to subgraphs H with $\sigma_H = 0$. Then return to Step 1 with the augmented Q-matching.

(1.2) If a neutral blossom is found, then shrink it as in the cardinality Q-matching algorithm. Repeat Step 1 until a hungarian forest is obtained. Then proceed to Step 2.

STEP 2. (Change of Dual Variables)

Let $\Delta_1 = \min \{\bar{w}_i : e_i$ is incident to both an even vertex and an unlabeled vertex$\}$.

Let $\Delta_2 = \min \{1/2\ \bar{w}_i : e_i$ is incident to two even vertices not in the same shrunken pseudo-vertex$\}$.

Let $\Delta_3 = \min \{\sigma_H : H \text{ is odd}\}$.

Let $\Delta_4 = \min \{\lambda_v : v \text{ is even}\}$.

Let $\Delta = \min (\Delta_1, \Delta_2, \Delta_3, \Delta_4)$.

(2.1) Increase (resp., decrease) λ_v by Δ if v is odd (resp., even).

(2.2) Decrease (resp., increase) σ_H by 2Δ if H is odd (resp., even).

(2.3) If $\Delta = \Delta_3$ expand all pseudo-vertices for which the corresponding dual variable is 0.

If $\Delta = \Delta_4$, to to Step 3. Else let G' be the resulting feasible digraph and return to Step 1.

STEP 3. Expand all pseudo-vertices, choosing a maximum weight Q-matching (M,Q) consisting of feasible edges. The corresponding dual vectors λ, σ are optimal for (6.2).

<u>Theorem 3</u>. The maximum weight Q-matching algorithm determines a solution which is optimal for the continuous relaxation of (6.1), and the algorithm runs in polynomial time.

<u>Proof</u>. We first prove inductively that the dual variables λ and σ are feasible for linear program (5.2). Note that the initial choice of λ and σ is feasible. Assume that $\bar{w}_j \leq 0$ for $j = 1, \ldots, m$ at a given iteration

for a given dual solution λ, σ. Let λ', σ', and $\overline{w}'$ be the dual variables and reduced weights after the next change of variables in Step 2, and we claim that $\overline{w}'_j \le 0$ for j = 1, ..., m. Let e_j = (u,v) and we divide the analysis into the following cases: (1) both u and v are in the same shrunken pseudo-vertex, (2) both u and v are even but are not in the same shrunken pseudo-vertex, (3) both u and v are odd but not in the same shrunken pseudo-vertex, (4) u is even and v is odd, (5) u is even and v is unlabelled, (6) u is odd and v is unlabeled, and (7) both u and v are unlabelled. (The other cases can be obtained by interchanging u and v in (4), (5) and (6).) It is easily verified that $\overline{w}'_j = \overline{w}_j$ if e_j is as in cases (1), (4), and (7). In case (2), $\overline{w}'_j = \overline{w}_j + \Delta$. In case (3), $\overline{w}'_j = \overline{w}_j - 2\Delta$. In case (5), $\overline{w}'_j = \overline{w}_j + \Delta$. In case (6), $\overline{w}'_j = w_j - \Delta$. By our choice of Δ in Step 2, we are guaranteed that $\overline{w}'_j \le 0$ in each case.

We next note by case (1) above, that all edges in neutral blossoms ramain feasible after contraction, and by cases (4) and (7), all edges in $M \cup Q$ remain feasible after the change of dual variables. We also note that (6.3c) is satisfied at each stage because we do not expand pseudo-vertices unless the associated dual variable is zero.

We see that the algorithm runs in polynomial time as follows. First, the number of augmentations is bounded by $|V|$, and the number of contractions between two successive augmentations is bounded by $|V|/3$. Finally, if $\Delta = \Delta_1, \Delta_2$, or Δ_3 in Step 2, then the new feasible edge will result in either (1) the labeling of an unlabeled vertex, or (2) the contraction of a neutral blossom, or (3) an augmentation. Thus the number of iterations of Step 2 between successive augmentations is bounded by $|V|$. Thus the algorithm is polynomial.

Finally, we claim that the algorithm terminates with a maximum weight Q-matching. To see this, we note that the algorithm either terminates because there are no exposed vertices, or else $\Delta = \Delta_4$ in Step 2. In the first case, we have primal feasibility, dual feasibility and complementary slackness, and thus the Q-matching is optimal for the linear relaxation of (6.1). In the second case, we note after the change in dual prices that $\lambda_v = 0$ for some exposed vertex v. In fact, $\lambda_w = 0$ for any exposed vertex w. To see this, note that at the initial step each λ-variable is set to $w_{max}/2$ and at each change of dual variables each exposed vertex is decreased by a common amount Δ. Because $\lambda_w = 0$ for all exposed vertices w, the complementary slackness condition (6.3b) is satisfied at termination. □

8. Extensions and New Directions

Speed-up Techniques

The primal-dual algorithm for the Q-matching problem presented in Section 7 is polynomial, but little attempt was made to minimize the run-time. In fact, the algorithm can be made to run in $O(|V(G)|^3)$ steps by using the data structures and implementation described by Lawler [1976] for the weighted matching problem. However, the reader should be forewarned that the details of such an implementation are complex and require a great deal of care.

It is an open question as to whether the cardinality Q-matching algorithm may be solved in $o(|V(G)|^3)$ steps. In particular, it would be interesting to know whether the $O(|V(G)|^{5/2})$ time algorithm developed by Even and Kariv [1975] can be extended so as to determine Q-matchings in a proportional amount of time. We conjecture that such an extension does not exist for the following reason. The Even-Kariv result depended on theory developed by Hopcroft and Karp [1973] concerning "maximal sets of augmenting paths of at most k edges". The Karp and Ullman results have no direct extension to Q-matchings. In fact, we have already seen that the Q-matching procedure does not depend on the existence of augmentations in the original graph.

Extensions to b-Matchings

A b-matching is the generalization of matchings in which we require the number of edges incident to vertex i to be at most b_i for each $i \in V(G)$. The b-matching problem has been investigated and solved by Edmonds et al. [1969] and Pulleyblank [1973] and [1977]. We conjecture that if the dynamic matching problem of Part I of this paper is generalized to dynamic b-matchings, then it can be solved in polynomial time by an appropriate generalization of b-matchings to Quasi-dynamic fractional b-matchings (Q-b-matchings).

Acknowledgements

I wish to thank Professor Arthur F. Veinott, Jr. and John VandeVate for their helpful suggestions.

Referenc

Berge, C. 1957. Two Theorems in Graph Theory. Proceedings of the National Academy of Science 43, 842-844.

Cornuejols, G., D. Hartwigger and W. Pulleyblank. 1982. Packing Subgraphs in a Graph. CORE Discussion Paper No. 8208, Center for Operations Research and Econometrics, Louvain-la-Neuve, Belgium.

Cornuejols, G. and W. Pulleyblank. 1980a. A Matching Problem with Side Conditions. Discrete Mathematics 29, 135-159.

Cornuejols, G. and W. Pulleyblank. 1980b. Perfect Triangle-Free 2-matchings, Mathematical Programming Study 13, 1-7.

Cornuejols, G. and W. Pulleyblank. 1981. Critical Graphs, Matchings and Tours or a Hierarchy of Relaxations for the Travelling Salesman Problem. Technical Report No. 81198-OR. Institut fur Okonometrie and Operations Research, Reinische Friedrich-Wilhelms-Universitat, Bonn, Germany.

Cunningham, W. and A. Marsh III. 1979. A Primal Algorithm for Optimum Matching. Mathematical Programming Study 8, 50-72.

Edmonds, J. 1965a. Paths, Trees and Flowers. Canadian Journal of Mathematics 17, 449-467.

Edmonds, J. 1965b. Maximum Matchings and a Polyhedron with (0,1) Vertices. Journal of Research of the National Bureau of Standards 6913, 125-130.

Edmonds, J., E. Johnson and S. Lockhart. 1969. Blossom I, A Code for Matching. Unpublished Report, IBM Research Center, Yorktown Heights, New York.

Even, S. and Q. Kariv. 1975. An $0(n^{5/2})$ Algorithm for Maximum Matching in General Graphs. In 16th Annual Symposium on the Found. of Comp. Science, IEEE, New York, 100-112.

Garey, M. and D. Johnson. 1979. Computers and Intractability: A Guide to the Theory of NP-Completeness. San Francisco: W. H. Freeman and Co.

Hopcroft, J. and R. Karp. 1973. A $n^{5/2}$ Algorithm for Maximum Matching in Bipartite Graphs. SIAM Journal of Computing 2, 225-231.

Lawler, E. 1976. Combinatorial Optimization: Networks and Matroids. Holt, Rinehart and Winston, New York.

Orlin, J. 1982. Dynamic Matchings and Quasi-Dynamic Fractional Matchings I.

Papadimitriou, C. and K. Steiglitz. 1982. Combinatorial Optimization: Algorithms and Complexity. Englewood Cliffs, New Jersey: Prentice-Hall.

Pulleyblank, W. 1973. Facets of Matching Polyhedra. Ph.D. Thesis. University of Waterloo, Waterloo, Ontario.

Pulleyblank, W. 1977. Dual Integrality in b-Matching Problems. CORE Discussion Paper, Louvain, Belgium

Pulleyblank, W. and J. Edmonds. 1974. Facets of 1-Matching Polyhedra. In C. Berge and Ray-Chaudhuri, eds., Hypergraph Seminar, Lecture Notes in Mathematics, No. 411, Springer, Berlin.

Tutte, W. 1947. The Factorization of Linear Graphs. Journal of the London Mathematical Society 22, 107-111.