A Note on Matchings Constructed during Edmonds' Weighted Perfect Matching Algorithm

Matthias Walter and Volker Kaibel
Otto-von-Guericke University Magdeburg

April 17, 2015

Abstract

We reprove that all the matchings constructed during Edmonds' weighted perfect matching algorithm are optimal among those of the same cardinality (provided that certain mild restrictions are obeyed on the choices the algorithm makes). We conclude that in order to solve a weighted matching problem it is not needed to solve a weighted perfect matching problem in an auxiliary graph of doubled size. This result was known before, e.g., posed as an exercise in see Lawler's book from 1976, but is not present in several modern books on combinatorial optimization.

1 Introduction

In this note we consider the minimum weight (perfect) matching problem. An instance consists of an undirected simple graph $G = (V, E)$ with node set V and edge set E as well as weights $w \in \mathbb{R}^E$. A *matching* $M \subseteq E$ is a set of edges no two of which share a node. A matching M is *perfect* if every node is covered, i.e., $|M| = \frac{1}{2}|V|$. The *matching number* $\nu(G)$ is the maximum cardinality of a matching in G. The objective is to find a (perfect) matching $\widehat{M}$ of minimum weight $w(\widehat{M}) = \sum\limits_{e \in M} w_e$. We call a matching *cardinality-optimal* if it has minimum weight among all matchings of the same cardinality.

The set of edges incident to a node $v \in V$ will be denoted by $\delta(v)$ and for some node set $U \subseteq V$ the set of edges inside U is $E[U]$. The set $\mathcal{U} = \{U \subseteq V : |U| \text{ odd}\}$ consists of all node subsets of odd cardinality. The characteristic vector $\chi(M) \in \{0,1\}^E$ of a subset $M \subseteq E$ is the vector with $\chi(M)_e = 1$ if and only if $e \in M$. For a vector $x \in \mathbb{R}^E$ and some set $F \subseteq E$ we write $x(F)$ for $\sum_{e \in F} x_e$. We assume that the reader is familiar with the matching algorithm, basic matching theory, and with linear programming concepts. For further material, see [8].

In 1965, Edmonds devised a combinatorial algorithm for cardinality-maximum matching [4]. In the same year he carried out the analysis of the matching polyhedron yielding a version of the algorithm that can solve weighted matching problems as well (see [3]). Lawler poses an exercise in his book from 1976 (see Problem 8.2 in [7]): Prove that the intermediate matchings constructed in the algorithm are cardinality-optimal. His version of Edmonds' algorithm is based on the so-called blossom inequalities which makes this problem easy to solve.

In contrast to this, more modern standard books on combinatorial optimization (e.g., [9], [6]) are based on the cut inequalities (which are only valid for the perfect-matching polytope), and for which the technical effort for a proof based on linear programming is higher.

We use the the version of Edmonds' algorithm presented in Chapter 26 of Schrijver's book [9] and only briefly recall the algorithm and important properties in Section 2. We then discuss the matching polytope, adjacency properties, and linear programming formulations. In Section 4 we present the proof of the mentioned result (Theorem 7) stating that the intermediate matchings constructed in the algorithm are cardinality-optimal. We provide two proofs, one by constructing a dual-feasible solution, and another one by extending the current graph. In both cases we prove optimality using complementary slackness. We close by discussing the imposed conditions as well as applications of our main result.

Before going into the details of the algorithm, observe that we can assume the edge weights w to be nonnegative, since otherwise we can add a large constant C to every weight. Then the weight of every matching of cardinality k is increased by the same value $k \cdot C$, leaving the ordering of these matchings intact.

2 Edmonds' Algorithm

In order to setup the notation we present the algorithm according to Schrijver's book [9] skipping the analysis.

The input is an undirected graph $G = (V, E)$ with nonnegative edge weights $w \in \mathbb{R}_+^E$. The algorithm is primal-dual where the primal state is given by a current matching M in G (with primal variable $x = \chi(M)$). The dual state is given by a laminar collection $\Omega \subseteq \mathcal{U}$ (i.e., for each $U, U' \in \Omega$ we either have $U \cap U' = \varnothing$ or $U \subseteq U'$ or $U' \subseteq U$) with associated dual variables $\pi : \mathcal{U} \to \mathbb{R}$ having support only in Ω, i.e., $\pi(U) = 0$ for all $U \notin \Omega$. We can state the constraints for π as follows:

$$\pi(U) \geq 0 \qquad \forall U \text{ with } |U| \geq 3 \tag{1}$$

$$\sum_{U : e \in \delta(U)} \pi(U) \leq w_e \qquad \forall e = \{u, v\} \in E \tag{2}$$

All singleton sets $\{v\}$ will be contained in the collection Ω throughout the algorithm and hence the inclusion-wise maximal sets in Ω, denoted by $\Omega_{\max}$, form a partition of V. The main auxiliary graph is G' which is obtained by only considering edges $\{u, v\} \in E$ which satisfy (2) with equality and shrinking all node sets $U \in \Omega_{\max}$ to single nodes.

The algorithm initializes $M := \varnothing$ and $\pi := \mathbb{O}_{\mathcal{U}}$. Throughout the algorithm π will always be feasible and complementary slackness will always be preserved. As soon as the matching M becomes feasible (i.e., a perfect matching) it is also optimal.

The algorithm attempts to do primal updates or, if these are not possible, dual updates and repeats until a perfect matching is found or it asserts that no perfect matching exists.

For the primal update, the set $X \subset \Omega_{\max}$ of unmatched nodes in G' is considered. We try to find an M-alternating walk in G' from a node in X to another node in X. In case this walk is a path, M gets augmented, otherwise a blossom is found and its associated odd set U is then shrunken and added to Ω.

In case no such walk is found, the disjoint node sets $\mathcal{S}, \mathcal{T} \subseteq \Omega_{\max}$ of G' are computed which contain the nodes to which G' has an odd (resp. even) length M-alternating path starting from any node in X. Then the π-values of those nodes U in $\mathcal{S}$ (resp. $\mathcal{T}$) are decreased (resp. increased) by the largest α such that the constraints (1) and (2) are still satisfied. If α can be chosen arbitrarily large, then the algorithm asserts that G has no perfect matching. At the end of the phase all $U \in \Omega_{\max}$ with $|U| \geq 3$ for which $\pi(U)$ became 0 are deshrunken and are removed from Ω.

The next proposition states that the case of arbitrarily large α occurs as late as possible.

Proposition 1. *When the algorithm asserts that G has no perfect matching, then the current matching has maximum cardinality, that is, $|\widetilde{M}| = \nu(G)$.*

Proof. Suppose we are in the situation that α can be chosen arbitrarily large.

First observe that the nodes in $\mathcal{S}$ must be singletons, i.e., are not shrunken, since their π-values are decreased by α and for non-singletons Inequality (1) would restrict the decrease. Furthermore, they are matched to nodes in $\mathcal{T}$ by definition of $\mathcal{S}$ and $\mathcal{T}$.

Second, nodes in $\mathcal{T}$ only have neighbors (in G') in $\mathcal{S}$ since their π-values are increased by α, but Inequality (2) for the edge in question does not restrict α.

Now assume, for the sake of contradiction, that $|\widetilde{M}| < \nu(G)$ holds and hence, there exists a $\widetilde{M}$-augmenting path P in G connecting some $\widetilde{M}$-exposed nodes $s, t \in X$.

From this it follows that every $\widetilde{M}$-edge of P is either an edge in G' connecting a node from $\mathcal{S}$ with a node from $\mathcal{T}$, or is an edge inside a blossom (whose representative node is in $\mathcal{T}$). But this already contradicts the fact that P is $\widetilde{M}$-augmenting since there is no edge between two nodes from $\mathcal{T}$. $\square$

This version of the algorithm clearly satisfies the following condition which is important as we will see later.

Condition 2. *The dual values are initialized by $\pi(U) := 0$ for U with $|U| \geq 3$ and $\pi(\{v\}) := \beta$ for all $U = \{v\}$ with $v \in V$ for some fixed $\beta \in \mathbb{R}$.*

Furthermore, in a dual update step, the dual values $\pi(U)$ are increased by the same amount for all $U \in \mathcal{T}$ for which the values are increased.

Let for every node $v \in V$ the accumulated dual value be denoted by $\pi^*(v) := \sum_{U \in \mathcal{U}, v \in U} \pi(U)$ and let $\pi^*_{\max} := \max_{v \in V} \pi^*(v)$ be their maximum. Let at any stage $\widetilde{M}$ be the matching that one obtains from the current matching M by iteratively deshrinking the sets in $\Omega_{\max}$.

Proposition 3. *At any stage of the algorithm, the matching $\widetilde{M}$ and the dual values π satisfy these properties:*

(i) If $\pi(U) > 0$ for some set $U \in \mathcal{U}$ with $|U| \geq 3$, then $U \in \Omega$ satisfies

$$\left|\widetilde{M} \cap E[U]\right| = \frac{1}{2}(|U| - 1) \ . \tag{3}$$

(ii) Shrinking and deshrinking do not modify the current matching $\widetilde{M}$. Furthermore, augmenting M in G' corresponds to an augmentation of $\widetilde{M}$ along a single path.

(iii) Matched nodes will always remain matched.

(iv) The dual values $\pi(U)$ for unmatched *nodes U of G' are always increased in a dual update.*

The next corollary follows readily from Proposition 3 (iii) and (iv).

Corollary 4. *Given Condition 2, at any stage $\pi^*(v) = \pi^*_{\max}$ holds for every unmatched node $v \in X$.*

3 The Matching Polytope

Let $P(G) = \operatorname{conv}\{\chi(M) : M \text{ matching in } G\}$ be the *matching polytope* of a graph G. Here $\chi(M) \in \{0,1\}^E$ denotes the characteristic vector which has $\chi(M)_e = 1$ for $e \in E$ if and only if $e \in M$. Edmonds [3] gave the following outer description of the polytope $P(G)$:

$$x_e \geq 0 \qquad \forall e \in E \tag{4}$$

$$x(\delta(v)) \leq 1 \qquad \forall v \in V \tag{5}$$

$$x(E[U]) \leq \frac{|U|-1}{2} \qquad \forall U \in \mathcal{U} \tag{6}$$

Chvátal [1] characterized adjacency in $P(G)$:

Proposition 5. *The vertices corresponding to two matchings M and M' are adjacent in $P(G)$ if and only if $M \Delta M'$ is connected.*

Corollary 6. *The sequence of matchings constructed during the algorithm corresponds to the vertex sequence of a path in the matching polytope. The vertices correspond to matchings of strictly increasing cardinality.*

Proof. This follows directly from Proposition 3 (i) and (ii). □

Another direct consequence is that $|M| - |M'| \in \{-1, 0, +1\}$ holds for adjacent vertices $\chi(M)$, $\chi(M')$. Hence, for every $k \in \mathbb{Z}_+$ we have that the convex hull P_k of all matchings of cardinality k is equal to P intersected with all $x \in \mathbb{R}^E$ satisfying

$$x(E) = k \ , \tag{7}$$

since such a hyperplane does not intersect the relative interior of any edge of $P(G)$. We now state the corresponding linear program

$$\text{(P)} \qquad \min \langle w, x \rangle \ \text{ s.t. } (4), (5), (6), \text{ and } (7)$$

and its dual:

$$
\text{(D)}\quad
\begin{array}{llll}
\min & \sum_{v\in V} y_v + \sum_{U\in\mathcal{U}} \frac{|U|-1}{2} z_U + k\gamma & & \\
\text{s.t.} & y_v + y_w + \sum_{\substack{U\in\mathcal{U}\\ e\in E[U]}} z_U + \gamma \le w_e & \forall e=\{v,w\}\in E & (8)\\
& y_v \le 0 & \forall v \in V & (9)\\
& z_U \le 0 & \forall U \in \mathcal{U} & (10)
\end{array}
$$

Here y, z, γ correspond to Inequalities (5), (6), and Equation (7), respectively.

For the case of perfect matchings, i.e., $k = \frac{1}{2}|V|$, there is an equivalent formulation of P_k using blossom inequalities in *cut form* requiring (5), (7) and

$$
x(\delta(U)) \ge 1 \qquad \forall U \in \mathcal{U} \tag{11}
$$

instead of (6).

4 Intermediate Matchings

We consider the state of the algorithm in any stage. Let $k := |\widetilde{M}|$ and consider the two linear programs for finding cardinality-optimal matchings. Clearly, $x := \chi(\widetilde{M})$ is a feasible solution for (P) since it is a matching of the right cardinality. The setup constructed so far prepares us for proving our main result.

Theorem 7. *Given Condition 2, every matching $\widetilde{M}$ constructed during Edmonds' weighted matching algorithm is cardinality-optimal.*

Our proof strategy is to construct a dual feasible solution y, z, γ and then proof complementary slackness. Clearly, dual values for Inequalities (6) can be calculated from those of Inequalities (11) by following the transformation in the proof showing their equivalence. This leads to the following formulas:

$$
\begin{array}{rcll}
\gamma & := & 2\pi^*_{\max} & \\
y_v & := & \pi^*(v) - \pi^*_{\max} & \forall v \in V \\
z_U & := & \begin{cases} -2\pi(U) & \forall U \in \Omega \text{ with } |U| \ge 3 \\ 0 & \forall U \in \mathcal{U} \text{ with } |U| = 1 \text{ or } U \notin \Omega \end{cases} &
\end{array} \tag{12}
$$

Here, again, $\pi^*(v) := \sum_{U\in\Omega, v\in U} \pi(U)$ and $\pi^*_{\max} := \max_{v\in V} \pi^*(v)$. Using this transformation we can easily proof Theorem 7.

Proof of Theorem 7. Let $\pi(U)$ for $U \in \Omega$ be the dual values at the current stage of the algorithm, and let $k := |\widetilde{M}|$ be the cardinality of the current matching. We construct a solution to the dual LP by applying the transformation (12).

We first prove that (y, z, γ) is feasible for (D). Clearly $y \le \mathbb{O}_V$ and $z \le \mathbb{O}_{\mathcal{U}}$. Furthermore, observe for every edge $e = \{u, v\}$ the relation

$$
y_u + y_v + \sum_{\substack{U\in\mathcal{U}\\ e\in E[U]}} z_U + \gamma = \sum_{\substack{U\in\mathcal{U}\\ u\in U}} \pi(U) + \sum_{\substack{U\in\mathcal{U}\\ v\in U}} \pi(U) - 2\sum_{\substack{U\in\mathcal{U}\\ e\in E[U]}} \pi(U) = \sum_{\substack{U\in\mathcal{U}\\ e\in\delta(U)}} \pi(U) \le w_e \tag{13}
$$

is satisfied, proving feasibility. It also proves that Inequality (8) is satisfied with equality for edges e with $x_e > 0$ since in this case the algorithm ensures that also Inequality (2) is satisfied with equality.

Second, assume $z_U < 0$, that is, $\pi(U) > 0$ for some $U \in \mathcal{U}$ with $|U| \geq 3$. Then, by Proposition 3 (i), the corresponding Inequality (6) is tight. Third, if $x(\delta(v)) < 1$ holds, i.e., v is not matched by $\widetilde{M}$, then by Corollary 4 we have $y_v = \pi^*(v) - \pi^*_{\max} = 0$. This concludes the proof. □

Alternative Proof

Jens Vygen suggested the following alternative proof idea (personal communication). Let $v_1, \dots, v_k$ be all unmatched nodes at the current stage. We construct an auxiliary graph $G' = (V', E')$ with $V' = V \;\dot\cup\; \{u_1, \dots, u_k\}$ and $E' = E \;\dot\cup\; \{\{v, u_i\} : v \in V, i \in [k]\}$ (see Figure 1). We define weights $w'_e = w_e$ for $e \in E$ and $w'_e = 0$ for $e \in E' \setminus E$. Then every matching M in G can be extended to a perfect matching M' in G' of the same weight.

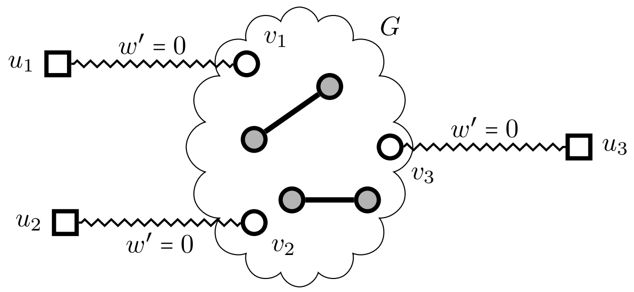

Figure 1: Alternative proof of Theorem 7 via auxiliary graph.

The extended version $\widetilde{M}'$ of the current matching $\widetilde{M}$ (from the algorithm) is a minimum perfect matching using the following dual values:

$$\begin{array}{rcll} \pi'(U) & := & \pi(U) & \forall U \in \Omega \\ \pi'(\{u_i\}) & := & -\pi^*_{\max} & \forall i \in [k] \end{array}$$

Then π' is dual feasible since π was dual feasible in G and since $\pi^*(v_i) = \pi^*_{\max}$ for all $i \in [k]$ we have $\pi^*(v_i) + \pi^*(u_i) = 0 = w'_e$ for all edges $e = \{v_i, u_i\}$, $i \in [k]$. Furthermore, these edges are tight and hence $\widetilde{M}'$ consists of only tight edges. Clearly all U have some leaving edges since $\widetilde{M}'$ is perfect, that is, the complementary slackness conditions are satisfied, too.

5 Discussion

From our main theorem it readily follows that one can run the algorithm on any graph and obtain a cardinality-optimal matching of every possible cardinality. From those we can then select the optimum matching and hence solve the minimum weight matching problem using the minimum weight perfect matching algorithm. This is different from the usual construction auf an auxiliary graph $\widetilde{G}$ of twice the size in order to solve a matching problem on G by solving a perfect matching problem on $\widetilde{G}$.

Unfortunately, Condition 2 seems to be unattractive for practical usage. Cook and Rohe [2] invented the idea not to modify the dual values by the same amount on all nodes in question in order to allow greater changes in different parts of the graph. Since then, for state of the art implementations (e.g. *Blossom V*, [5]) our imposed condition does not hold.

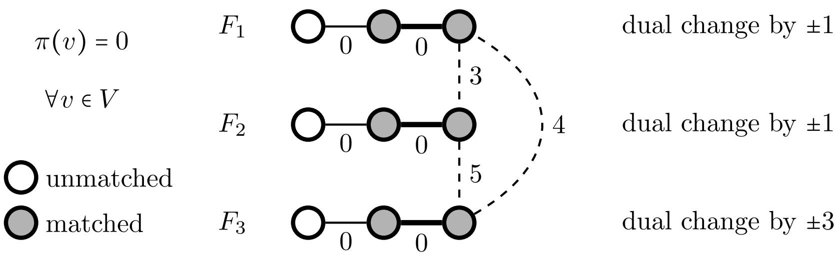

Figure 2: An example showing necessity of Condition 2.

In the following example (see Figure 2) we have three alternating forests of which all edges have weight 0. All dual variables are 0 as well. According to the above mentioned "variable dual changes" approach we can, for whatever reason, modify the values of the forests F_1, F_2, F_3 by 1, 1, and 3, respectively. In the next step only the edge of weight 4 has become tight, leading to a suboptimal augmentation.

Ackknowledgements. We are thankful to Jens Vygen for comments on an earlier version of the manuscript, among those the alternative proof of Theorem 7 given above.

References

[1] V. Chvátal. On certain polytopes associated with graphs. *Journal of Combinatorial Theory, Series B*, 18(2):138 – 154, 1975.

[2] W. Cook and A. Rohe. Computing minimum-weight perfect matchings. *INFORMS J. on Comp.*, 11:138–148, 1998.

[3] J. Edmonds. Maximum matching and a polyhedron with 0, 1-vertices. *Journal of Research of the National Bureau of Standards B*, 69:125–130, 1965.

[4] J. Edmonds. Paths, trees, and flowers. *Canad. J. Math.*, 17:449–467, 1965.

[5] V. Kolmogorov. Blossom V: a new implementation of a minimum cost perfect matching algorithm. *Math. Program. Ser. C*, 1:43–67, 2009.

[6] B. Korte and J. Vygen. *Combinatorial Optimization*, volume 21. Springer, fifth edition, 2012.

[7] E. Lawler. *Combinatorial Optimization: Networks and Matroids.* Saunders College Publishing, Fort Worth, 1976.

[8] A. Schrijver. *Theory of linear and integer programming.* John Wiley & Sons, Inc., New York, NY, USA, 1986.

[9] A. Schrijver. *Combinatorial Optimization - Polyhedra and Efficiency.* Springer, 2003.