Lecture notes for Mar 22, 2023
Maximum weighted matching in general graphs

Chun-Hung Liu

March 22, 2023

We first reduce the problem for finding maximum weighted matching to the problem for finding minimum weighted perfect matching.

Proposition 1 *If there exists an algorithm for finding the minimum weighted perfect matching of any input simple graph* (G', w') *in time* $f(|V(G')|)$*, then there exists an algorithm for finding the maximum weighted matching in an input graph* (G, w) *in time* $f(2|V(G)|) + O(|V(G)| + |E(G)|)$*.*

Proof. Let G be a graph. Let $w : E(G) \to \mathbb{R}$ be a function. Note that maximum weighted matching does not contain an edge with negative weight (since, if so, then we can remove it from the matching to get another matching with bigger weight). And removing edges with zero weight does not change the weight of a matching. So we may assume that $w > 0$. Moreover, for every pair of adjacent vertices u and v, if there are multiple edges between u and v, then we can only keep an edge with the maximum weight and deleting other edges between u and v without changing the maximum weight of a matching. So we may assume G is simple.

Create two disjoint copies (G_1, w_1) and (G_2, w_2) of (G, w). Let H be the graph obtained from $G_1 \cup G_2$ by adding a perfect matching $\{v'v'' : v' \in V(G_1), v'' \in V(G_2), v'$ and v'' are the copies of the same vertex v of $G\}$. For every $e \in E(H)$, let $w_H(e) = -w_1(e)$ if $e \in E(G_1)$, let $w_H(e) = -w_2(e)$ if $e \in E(G_2)$, and let $w_H(e) = 0$ for every $e \in E(H) - (E(G_1) \cup E(G_2))$.

Let M be a matching of G with $\sum_{e \in M} w(e)$ maximum. Let $k = \sum_{e \in M} w(e)$. Let M_H be the matching of H by collecting the edges in G_1 and G_2 corresponding to M and the edges between the vertices of G_1 and G_2 corresponding to non-M-saturated vertices of G. So M_H is a perfect matching in H, and $\sum_{e\in E(M_H)} w_H(e) = -2\sum_{e\in E(G)} w(e) = -2k$.

Let M^* be a perfect matching of H with minimum weight. Let $k_H = \sum_{e\in M^*} w(e)$. So $k_H \leq -2k$ by the minimality of M^*. On the other hand, $M^* \cap E(G_1)$ and $M^* \cap E(G_2)$ are matachings in G, so their weight are at most k. Since the edges in $E(H) - (E(G_1) \cup E(G_2))$ has zero weight, $\sum_{e\in M^*} w_H(e) = -\sum_{e\in M^*\cap E(G_1)} w(e) - \sum_{e\in M^*\cap E(G_2)} w(e) \geq -2k$. So $k_H = \sum_{e\in M^*} w_H(e) = -2k$.

Note that k_H can be found in time $f(|V(H)|) = f(2|V(G)|)$, and H can be constructed in time $O(|V(G)| + |E(G)|)$. So k can be found in time $f(2|V(G)|) + O(|V(G)| + |E(G)|)$. Moreover, given a matching in H with weight k_H, we can find a matching in G with weight $-k/2$ in time $O(|V(G)|+|E(G)|)$ by simply taking the edges contained in G_1. ■

The main idea to find a minimum weighted perfect matching is to combine Edmonds' blossom algorithm for finding unweighted maximum matching and the Hungarian method for finding maximum weighted matching in bipartite graphs.

We can formulate the minimum weighted perfect matching problem as follows.

Proposition 2 *Let G be a graph. Then a subset M of $E(G)$ is a perfect matching if and only if the corresponding vector $x \in \{0,1\}^{|E(G)|}$ satisfies*

- *for every $v \in V(G)$, $\sum_{e\in\delta(v)} x_e = 1$, and*
- *for every $O \subseteq V(G)$ with $|O|$ odd, $\sum_{e\in\delta(O)} x_e \geq 1$.*

Therefore, given $w : E(G) \to \mathbb{R}$, the perfect matchings with minimum weight are exactly the optimal solutions of the problem $\min_x w^T x$ subject to

- *for every $v \in V(G)$, $\sum_{e\in\delta(v)} x_e = 1$,*
- *for every $O \subseteq V(G)$ with $|O|$ odd, $\sum_{e\in\delta(A)} x_e \geq 1$,*
- *for every $e \in E(G)$, $x_e \in \{0,1\}$.*

Proof. The first part of this proposition is obvious. The second part follows from the first part. ■

Let $\mathcal{O} = \{O \subseteq V(G) : |O| \text{ is odd}\}$.

The dual of the LP-relaxation of the maximization problem in Proposition 2 is $\max_y 1^T y$, where y is indexed by $\mathcal{O}$ such that

- for every $e \in E(G)$, $\sum_{O \in \mathcal{O}, \delta(O) \ni e} y_O \leq w(e)$, and
- for every $O \in \mathcal{O}$ with $|O| \geq 3$, $y_O \geq 0$.

The key idea to find the minimum weighted matching is to find a vector $x \in \{0,1\}^{|E(G)|}$ and a feasible solution y for the dual maximization problem such that

- for every $e \in E(G)$, if $x_e > 0$, then $\sum_{O \in \mathcal{O}, \delta(O) \ni e} y_O = w(e)$, and
- for every $O \in \mathcal{O}$, if $y_O > 0$, then $\sum_{e \in \delta(O)} x_e \leq 1$.

Note that we do not require x to be a feasible solution of the primal minimization problem. But if we know such an x corresponds to a perfect matching of G, then x is a feasible solution of the primal problem, and x and y is a pair of solution satisfying the complementary slackness, so x is an optimal solution for the primal problem and we obtain a minimum weighted perfect matching of (G, w).

A remark is that $|\mathcal{O}|$ is exponential in $|V(G)|$, so we are not affordable to record y_O for every $O \in \mathcal{O}$. We will only specify y_O for those $O \in \mathcal{O}$ with $y_O > 0$, and it is good enough for obtaining the complementary slackness mentioned above. And we will make sure that $\{O \in \mathcal{O} : y_O > 0\}$ is a laminar family and hence has size at most $2|V(G)|$. (A collection $\mathcal{F}$ of subsets of a set U is *laminar* if for any two members A, B of $\mathcal{F}$, either $A \cap B = \emptyset$, or $A \subseteq B$, or $B \subseteq A$. It is not hard to show that if $\mathcal{F}$ is a laminar set of subsets of U, then $|\mathcal{F}| \leq 2|U|$.)

Recall that in the Hungarian method, we consider the graph G_f, which is the subgraph of G consisting of the edges whose corresponding inequalities in the dual problem are tight, and look for a perfect matching in G_f. We consider similar things here as well, but we also contract some subsets of $V(G)$. Given a feasible solution y of the dual problem,

- an edge e of G is *tight (with respect to y)* if $\sum_{O \in \mathcal{O}, \delta(O) \ni e} y_O = w(e)$, and
- the *tight graph (with respect to y)*, denoted by G_y, is the graph obtained from G by deleting all non-tight edges and identifying each (inclusion-)maximal set in $\{O \in \mathcal{O} : y_O > 0\}$ into a vertex.

Note that $\{O \in \mathcal{O} : y_O > 0\}$ is laminar, so its inclusion-maximal sets are pairwise disjoint, and hence the identification mentioned above is well-defined.

Moreover, during the algorithm, we will make sure that for every $O \in \mathcal{O}$ with $y_O > 0$, the graph obtained from $G[O]$ by deleting all edges e with $\sum_{O\in\mathcal{O},\delta(O)\ni e} y_O < w(e)$ is factor-critical. That is, for every $v \in O$, $G[O] - v$ has a perfect matching M_v such that every edge e in the matching M_v satisfies $\sum_{O\in\mathcal{O},\delta(O)\ni e} y_O = w(e)$. It implies that if we have a matching M in G_y, then we can extend M to a matching in G such that every edge e in the matching M satisfies $\sum_{O\in\mathcal{O},\delta(O)\ni e} y_O = w(e)$, and for every maximal O with $y_O > 0$ whose corresponding vertex in G_y is saturated by M, every vertex in O is saturated by M.

================================

Algorithm for minimum weighted perfect matching in general graphs
Input: A simple graph G and a function $w : E(G) \to \mathbb{R}$.
Output: A perfect matching M in G such that $\sum_{e\in M} w(e)$ is minimum.
Procedure:

Step 1: Set $M = \emptyset$. For every $O \in \mathcal{O}$ with $|O| = 1$, set $y_O = \frac{1}{2}\min_{e\in\delta(O)} w(e)$.

(We will make sure that $\{O \in \mathcal{O} : y_O > 0\}$ is laminar during the entire process, and for every $O \in \mathcal{O}$ with $y_O > 0$, the graph obtained from $G[O]$ by only keeping tight edges is factor-critical.)

Step 2: Construct the tight graph G_y. Set M_y^0 to be the edges in M between different vertices of G_y.

(We will make sure that every edge in M is tight, so M_y^0 is a matching in G_y.)

Step 3: Use Edmonds' Blossom Algorithm starting from M_y^0 to find a maximum (unweighted) matching M_y in G_y and the corresponding Gallai-Edmonds decomposition (X_y, Y_y, W_y).

Step 4: Extend M_y to a matching M in G such that

- every edge in M is tight,
- for every $O \in \mathcal{O}$ with $y_O > 0$ corresponding to a vertex of G_y saturated by M_y, every vertex in O is saturated by M, and
- for every $O \in \mathcal{O}$ with $y_O > 0$, $|M \cap \delta(O)| \le 1$.

Step 5: If M_y is a perfect matching in G_y, then output M and stop. Otherwise, do the following:

- Let $\epsilon_1 = \min\{y_O : O \in \mathcal{O}, y_O > 0, |O| \geq 3$, the vertex of G_y corresponding to O is a vertex in $X_y\}$.
- Let $\epsilon_2 = \min\{w(e) - \sum_{O \in \mathcal{O}, y_O > 0} y_O : e \in E(G)$ is between a vertex of G contained in a member of $\mathcal{O}$ corresponding to a vertex in Y_y and a vertex of G contained in a member of $\mathcal{O}$ corresponding to a vertex in $W_y\}$.
- Let $\epsilon_3 = \frac{1}{2} \min\{w(e) - \sum_{O \in \mathcal{O}, y_O > 0} y_O : e \in E(G)$ is between a vertex of G contained in a member of $\mathcal{O}$ corresponding to a vertex in Y_y and a vertex of G contained in another member of $\mathcal{O}$ corresponding to a vertex in $Y_y\}$.
- Let $\epsilon = \min\{\epsilon_1, \epsilon_2, \epsilon_3\}$.
- For every O corresponding to a vertex of G_y belonging to X_y, redefine y_O to be $y_O - \epsilon$.
- For every component C of $G_y[Y_y]$, recall that $|V(C)|$ is odd by the Gallai-Edmonds decomposition, so the union of set $O \in \mathcal{O}$ corresponding to a vertex of G_y belonging to C is a subset of $V(G)$ with odd size, and we denote this set by O_C.
- For every component C of $G_y[Y_y]$, if y_{O_C} is undefined, then define $y_{O_C} = \epsilon$, otherwise, redefine y_{O_C} to be $y_{O_C} + \epsilon$.
- Do Step 2.

==============================

Lemma 3 *During the entire process, the following properties are preserved:*

1. $\{O \in \mathcal{O} : y_O > 0\}$ *is laminar.*

2. y *is a feasible solution of the dual maximization problem.*

3. *If an edge* e *of* G *is tight before an update of* y *and turns non-tight because of this update, then there exist* $O_1 \in \mathcal{O}$ *corresponding to a vertex of* G_y *belonging to* X_y *(before update) and* $O_2 \in \mathcal{O}$ *corresponding to a vertex of* G_y *belonging to* $X_y \cup W_y$ *(before update) such that* e *is between* O_1 *and* O_2*.*

4. *For every $O \in \mathcal{O}$ with $y_O > 0$, and for every $v \in O$, there exists a perfect matching $M_{O,v}$ of $G[O] - v$ using tight edges only, and for every $O' \in \mathcal{O}$ with $O' \subseteq O$, $|M_{O,v} \cap \delta(O')| \leq 1$.*

5. *For every $O \in \mathcal{O}$ with $y_O > 0$, the graph obtained from $G[O]$ by only keeping tight edges is factor-critical.*

6. *The extension from M_y to M in Step 4 is always possible.*

Proof. We first show property 1. It holds at the end of Step 1. During the process, the set $\{O \in \mathcal{O} : y_O > 0\}$ loses members or getting new members only during Step 5. Note that removing members from this set keeps it laminar. And at each Step 5, the new sets added into the collection are pairwise disjoint because each of them corresponds to a component of $G_y[Y_y]$, and each of those new sets added into the collection has the property that if it intersects some existing set, then it contains it. So property 1 is preserved.

Property 2 is preserved by the choice of ϵ.

Property 3 is clear from the algorithm.

Now we show property 4 is preserved. Property 4 clearly holds before the first time that Step 5 is executed. Note that whenever we run Step 5 to update y, we merge each component of $G[Y_y]$ into a vertex and possibly split some vertices in X_y into more than one vertices, and there is no other change for the vertex-set of G_y. Gallai-Edmonds decomposition ensures that each of the new vertex of G_y corresponds to a member O of $\mathcal{O}$ satisfying $G[O]$ is factor-critical with using tight edges only. And every removed tight edge because of the update of y is between maximal elements of $\mathcal{O}$ disjoint from Y_y by Property 3. So Property 4 is preserved at the end of this round of Step 5 as long as it is preserved at the beginning of this round of Step 5. As Steps 1-4 does not change $\mathcal{O}$, property 4 is preserved for the entire process.

Then properties 5 and 6 follow from property 4. ■

Lemma 4 (Edmonds) *If the algorithm stops, then the output M is a minimum weighted perfect matching in (G, w).*

Proof. By Step 4, since M_y is a perfect matching of G_y, M is a perfect matching of G. Let x be the 0-1 vector corresponding to M. Since every edge in M is tight, we know that for every $e \in E(G)$, if $x_e > 0$, then $\sum_{O \in \mathcal{O}, \delta(O) \ni e} y_O = w(e)$. And M has the property that for every $O \in \mathcal{O}$ with $y_O > 0$, $|M \cap \delta(O)| \leq 1$, so $\sum_{e \in \delta(O)} x_e \leq 1$. Since M is a perfect matching, for every $O \in \mathcal{O}$ with $y_O > 0$, $\sum_{e \in \delta(O)} x_e = 1$. Hence x is a feasible solution for the primal problem and y is a feasible solution of the dual problem such that x and z satisfy the complementary slackness. So x is an optimal solution of the primal problem. In other words, M is a minimum weighted perfect matching in (G, w). ■

So the only remaining concern is whether the algorithm stops (in finite time). It can be proved, but the proof is more complicated so we do not include it here.

Theorem 5 (Gabow) *The above algorithm can be implemented so that it runs in time $O(|V(G)|^3)$. Therefore, a minimum weighted perfect matching of a weighted simple graph (G, w) can be found in time $O(|V(G)|^3)$.*

Corollary 6 *Given a weighted graph (G, w), a matching in (G, w) with maximum weight can be found in time $O(|V(G)|^3 + |E(G)|)$.*

Proof. It immediately follows from Proposition 1 and Theorem 5. ■