Lecture notes for Mar 1, 2023
Edmonds' Blossom Algorithm for matching

Chun-Hung Liu

March 1, 2023

Now we try to find a maximum matching in a general graph. We use a similar strategy for finding M-augmenting paths like for bipartite graphs. But now the analysis is more complicated. One reason is that we would see edges between outer vertices to form odd cycles. However, such an odd cycle has certain nice properties, so we can still make it work by revising the algorithm. We will give more details in this section.

Let M be a matching in G. Let $v \in V(G)$. An *M-blossom based at v* is a subset B of $V(G)$ with $v \in B$ such that

- either v is not M-saturated, or v is matched by M to a vertex in $V(G) - B$,
- for every $u \in B - \{v\}$, there exist M-alternating paths P_1 and P_2 in $G[B]$ from u to v such that the edge of P_1 incident with u is in M, and the edge of P_2 incident with u is in $E(G) - M$.

Note that $\{v\}$ is an M-blossom based at v.

Like the maximum flow problem and the algorithm for solving it, there is a weak duality for matchings, and the algorithm will find a matching and a witness showing that the weak duality is tight. To do so, we first show an easy upper bound for the maximum size of the matching.

For a graph G and a subset X of $V(G)$, we define $\text{oddcomp}_G(X)$ to be the number of components of $G - X$ with an odd number of vertices.

Proposition 1 *Let G be a graph. Then*

$$\max_M |M| \leq \min_{X \subseteq V(G)} \frac{|V(G)| + |X| - \text{oddcomp}_G(X)}{2},$$

where the maximum is over all matchings M in G.

Proof. Let M be an arbitrarily matching in G. Let X be an arbitrary subset of $V(G)$. For each odd component C of $G - X$, at least one vertex in C either is unsaturated by M or is matched to some vertex in X. So at least $\text{oddcomp}_G(X) - |X|$ vertices in G are not saturated by M. Hence the number of saturated vertices is at most $|V(G)| - (\text{oddcomp}_G(X) - |X|)$. That is, $2|M| \leq |V(G)| + |X| - \text{oddcomp}_G(X)$. ■

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

Edmonds' Blossom Algorithm
Input: A simple graph G.
Output: A matching M and a subset X of $V(G)$ with $2|M| = |V(G)| + |X| - \text{oddcomp}_G(X)$. (So M must be a maximum matching by Proposition 1.)
Procedure:

Step 0: Set $M = \emptyset$.

Step 1: Set $\mathcal{F} = \{(\{v\}, v) : v \in V(G)\}$. Set T be the empty graph. Set S to be the empty sequence. For every $v \in V(G)$, set $b(v) = *$, $a(v) = *$, $R(v) = *$, $s(v) = *$, and unmark v. Unmark all edges of G.

Step 2: Do the following:

Step 2-1: If $S = \emptyset$ and every vertex of G is either marked or M-saturated, then set $X = \{v \in V(G) : s(v) = \text{inner}\}$, output M and X, and terminate the algorithm.

Step 2-2: Otherwise, if $S = \emptyset$, then pick an unmarked and non-M-saturated vertex v of G, add v into S and T, mark v, set $R(v) = v$, $b(v) = v$, and $s(v) = \text{outer}$, and then repeat Step 2.

Step 2-3: Otherwise, let v be the first entry of S, remove v from S, and do the following:

Step 2-3-1: Pick an unmarked edge uv incident with v, do the following based on the following cases:
(Before we mention those cases, we first define some notations. For each $x \in V(T)$ with $s(v)$=outer,

- let P_x^+ be the path $P_x^+ = v_1v_2...v_{|V(P_x^+)|}$, where $v_1 = x$, $v_{|V(P_x^+)|} = b(x)$, and if i is odd, then $v_{i+1} = a(v_i)$, and if i is even, then v_{i+1} is the vertex matched to v_i by M,
- let P_x^- be the path $P_x^- = v_1v_2...v_{|V(P_x^-)|}$, where $v_1 = x$, $v_{|V(P_x^-)|} = b(x)$, and if i is odd, then v_{i+1} is the vertex matched to v_i by M, and if i is even, then $v_{i+1} = a(v_i)$,
- let P_x be the path from x to $R(x)$ defined by repeatedly concatenating paths as follows: starting with the path P_x^-, and if the end (say x') of the current P_x other than x is not $R(x)$, then concatenate $x'p_{x'}P^{\circ}_{p_{x'}}$, where $p_{x'}$ is the parent of x', and $\circ = +$ if $x'p_{x'} \in M$, and $\circ = -$ if $x'p_{x'} \notin M$.

We just define P_x^+, P_x^- and P_x here. We do not construct these paths until we need them to define other paths mentioned below.)

- **When u is unmarked and u is not M-saturated (Augmenting case):**
  Set $M = M\Delta E(P_v + uv)$. Repeat Step 1.
- **When u is unmarked and u is M-saturated (Growing case):**
  Add uv into T and mark uv. Set $R(u) = R(v)$, $s(u)$ =inner, and $a(u) = v$. Let w be the vertex matched to u by M. Add uw into T, mark uw, set $R(w) = R(u)$, $b(w) = w$, $s(w)$ =outer, and add w to be the last entry of S. Repeat Step 2-3-1.
- **When u is marked, $R(u) \neq R(v)$, and $s(u)$=outer (Augmenting case):**
  Construct the path P_u, P_v, and let $P = (P_u \cup P_v) + uv$. Set M to be $M\Delta E(P)$. Repeat Step 1.
- **When u is marked, $R(u) = R(v)$, and $s(u)$=outer (Shrinking case):**
  Let w be the common ancestor of $b(u)$ and $b(v)$ furthest from $R(u) = R(v)$.
  For each $x \in \{u, v\}$ and each vertex y in the subpath of P_x between x and $b(w)$ with $b(y) \neq b(w)$ such that the distance in P_x between x and y is even, redefine $a(y)$ to be the neighbor of y in the subpath of P_x between x and

y.
For each $x \in \{u, v\}$, if $x \neq b(w)$, then define $a(x)$ to be the vertex in $\{u, v\} - \{x\}$.
For each $x \in \{u, v\}$, let P'_x be the path in T from $b(w)$ to x. Let $\mathcal{F}' = \{(Z, z) \in \mathcal{F} : |Z| = 1, Z \cap V(P'_u \cup P'_v) \neq \emptyset\} \cup \{(Z, z) \in \mathcal{F} : |Z| \geq 2, (Z - \{z\}) \cap V(P'_u \cup P'_v) \neq \emptyset\}$. Let $B = \bigcup_{(Z,z) \in \mathcal{F}'} Z$. Redefine $\mathcal{F}$ to be $(\mathcal{F} - \mathcal{F}') \cup \{(B, b(w))\}$. For every $x \in B$, redefine $b(x) = b(w)$, and if $s(x) =$inner, then redefine $s(x) =$outer, put x into the last entry of S, redefine $a(x)$ to be its parent, and unmark all edges of $E(G) - E(T)$ incident with x. Mark uv, and then repeat Step 2-3-1.

- **Other cases:** Mark uv and then repeat Step 2-3-1.

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

Lemma 2 *During the entire process, the following properties are preserved.*

1. *T is a forest and $V(T)$ equals the set of marked vertices.*

2. *M is a matching of G.*

3. *For every $v \in S$, $s(v) =$outer, and if v is M-saturated, then the vertex matched to v by M is in $V(T)$.*

4. *For every marked edge, its both ends are in T.*

5. *For every edge e in M, either $e \in E(T)$, or its both ends are not in $V(T)$.*

6. *$\bigcup_{(Z,z) \in \mathcal{F}} Z = V(G)$, and if (Z_1, z_1) and (Z_2, z_2) are distinct members of $\mathcal{F}$ with $Z_1 \cap Z_2 \neq \emptyset$, then $|Z_1| \geq 2 \leq |Z_2|$, $|Z_1 \cap Z_2| = 1$ and $Z_1 \cap Z_2 \subseteq \{z_1, z_2\}$.*

7. *For every member (Z, z) of $\mathcal{F}$, if $|Z| \geq 2$, then $Z \subseteq V(T)$, $T[Z]$ is a connected subgraph in T rooted at z, and $s(v) =$outer for every $v \in Z$.*

8. *For every $v \in V(T)$, the following hold.*

   (a) *$R(v)$ is the root of the component of T containing v.*

   (b) *If $R(v) \neq v$, then v is M-saturated.*

(c) *If $b(v) = v \neq R(v)$, then the edge between v and its parent is in M.*

(d) *If $a(v) \neq *$, then $va(v) \in E(G) - M$, and $s(a(v))$ =outer.*

(e) *If $s(v)$ is inner, then it has a parent in T, the edge between v and its parent is not in M, it has a unique child in T, and the edge between v and the child is an edge in M.*

(f) *If $v = b(u)$ for some $u \in V(G)$, then $s(v)$ =outer.*

(g) *If $b(v) \neq *$, then there exists a unique path W of the form $v_1v_2...v_{|V(W)|}$ satisfying $v_1 = v$, $v_{|V(W)|} = b(v)$, and $v_{i+1} = a(v_i)$ for every odd i, and v_{i+1} is the vertex matched to v_i by M for every even i; moreover, W is an M-alternating path in G from v to $b(v)$, $V(W)$ is contained in Z for some $(Z, b(v)) \in \mathcal{F}$, and all vertices of W are outer.*

(h) *If $b(v) \neq *$, then there exists a unique path W of the form $v_1v_2...v_{|V(W)|}$ satisfying $v_1 = v$, $v_{|V(W)|} = b(v)$, and $v_{i+1} = a(v_i)$ for every even i, and v_{i+1} is the vertex matched to v_i by M for every odd i; moreover, W is an M-alternating path in G from v to $b(v)$, and $V(W)$ is contained in Z for some $(Z, b(v)) \in \mathcal{F}$, and all vertices of W are outer.*

(i) *If $b(v) \neq v$, then the path P_v defined in the algorithm is an M-alternating path from v to $R(v)$ such that the edge of P_v incident with v is in M and the edge of P_v incident with $R(v)$ is not in M.*

9. *For every member (Z, z) of $\mathcal{F}$, Z is a blossom based at z.*

Proof. Clearly all properties hold at the beginning of the algorithm. Assume all properties are preserved at some point at the algorithm. We shall prove that they remain preserved when the algorithm keeps going. Properties 1, 3, 4, 5, 6, 7 and 8(a)-8(e) are clearly preserved. Since Properties 2 and 8(g) are preserved at this point, we know Property 2 is also preserved.

Now we show that Property 8(f) holds. It holds if $s(v)$ was outer when v was added into T. So we may assume that $s(v)$ was inner when v was added into T. Assume that we are at the moment that we change $s(v)$ to be outer. So we are at the Shrinking case. Let w be the vertex mentioned in that case, and denote the vertices u, v in the shrinking case in the algorithm by α, β, respectively. Note that $b(w) = w = v$, for otherwise $b(v)$ was assigned to a proper ancestor of v and cannot become v in the future. Since $s(v)$ was inner at that moment, the edge in M incident with v is the tree edge between v and its unique child c. By the definition of w, c is not a common ancestor of $b(\alpha)$ and $b(\beta)$. So $w = b(\alpha)$ or $w = b(\beta)$. But $s(b(\alpha))$ =outer= $s(b(\beta))$ by 8(e). Hence $w \notin \{b(\alpha), b(\beta)\}$, a contradiction. This proves Property 8(f).

Now we prove that 8(g) and 8(h) are preserved. We may assume $v \neq b(v)$, for otherwise we are done. And we may assume that we are executing the shrinking case, for otherwise, the existence of W does not change. Let α, β, w be the vertices u, v, w mentioned in the shrinking case, respectively.

We first assume that $b(v) \neq *$ before executing this shrinking case. So we have the desired path W at the moment, call W_0. If W_0 does not intersect $P_\alpha \cup P_\beta$, then $a(u)$ does not change for every $u \in V(W_0)$, so W_0 remains the desired path after executing this shrinking case. (Note that the existence of P_α and P_β follow from the assumption that 8(i) is preserved at this point.) So we may assume that there exists a vertex in $x \in V(W_0) \cap V(P_\alpha \cup P_\beta)$ such that the subpath W_0' of W_0 between v and x is internally disjoint from $P_\alpha \cup P_\beta$. Say $x \in V(P_\alpha)$ by symmetry. Note that P_α and P_β passes through $b(w)$ as Properties 6 and 7 are preserved. If $x = v$, then the concatenation of the subpath of P_α from x to α, the edge $\alpha\beta$, and the subpath of P_β between β and $b(w)$ is the desired path W. So we may assume $x \neq v$. By 8(b), the edge of W_0' incident with x is not in M, so the concatenation of W_0' and the subpath of P_α between x and $b(w)$ is the desired path W.

So we may assume that $b(v) = *$ before executing this shrinking case. That is, $s(v)$ was inner and turns outer in this shrinking case. So $a(v)$ is defined to be the parent p of v. Note that $s(p)$ is not inner during the algorithm. So the desired path W for 8(h) starting from p to $b(w)$ exists, called it W_p^-. By the construction of W_p^-, all vertices of W_p^- are either outer before this shrinking case or in $P_\alpha \cup P_\beta$. Note that every vertex in $P_\alpha \cup P_\beta$ is also outer before this shrinking case. So $v \notin V(W_p^-)$. Hence $vp + W_p^-$ is a desired path for 8(g) from v to $b(w)$. Similarly, let u be the vertex matched with v, then $s(u)$ =outer before this shrinking case. So the desired path W for 8(g) starting from u to $b(w)$ exists, called it W_u^+. By the construction of W_u^+, all its vertices are either outer before this shrinking case or in $P_\alpha \cup P_\beta$. Note that every vertex in $P_\alpha \cup P_\beta$ is also outer before this shrinking case. So $v \notin V(W_u^+)$. Hence $vu + W_u^+$ is a desired path for 8(h) from v to $b(w)$. This proves that property 8(g) and 8(h) are preserved.

Now we show 8(i). Note that 8(g) and 8(h) imply that the paths P_x^+ and P_x^- mentioned in the algorithm is well-defined. So the path P_x is well-defined path from v to $R(v)$ such that the edge incident with v is in M. Since $R(v)$ is not M-saturated, the edge of P_x incident with v is not in M. So it suffices to show that P_x is M-alternating. And it follows from 8(c), 8(g) and 8(h).

Finally we show property 9. If $z = R(z)$, then z is not M-saturated. If $z \neq R(z)$, then by 8(g), z is matched to its parent by M, but its parent is not in Z by Property 7. So Z is an M-blossom based at z by Properties 8(g) and 8(h). This proves the lemma. ∎