CS 598CSC: Combinatorial Optimization	**Lecture date: Feb 9, 2010**
Instructor: Chandra Chekuri	**Scribe: Matthew Yancey**

1 Matchings in Non-Bipartite Graphs

We discuss matching in general undirected graphs. Given a graph G, $\nu(G)$ denotes the size of the largest matching in G. We follow [1] (Chapter 24).

1.1 Tutte-Berge Formula for $\nu(G)$

Tutte (1947) proved the following basic result on perfect matchings.

Theorem 1 (Tutte) *A graph $G = (V, E)$ has a prefect matching iff $G - U$ has at most $|U|$ odd components for each $U \subseteq V$.*

Berge (1958) generalized Tutte's theorem to obtain a min-max formula for $\nu(G)$ which is now called the Tutte-Berge formula.

Theorem 2 (Tutte-Berge Formula) *For any graph $G = (V, E)$,*

$$\nu(G) = \frac{|V|}{2} - \max_{U \subseteq V} \frac{o(G-U) - |U|}{2}$$

where $o(G - U)$ is the number of components of $G - U$ with an odd number of vertices.

Proof: We have already seen the easy direction that for any U, $\nu(G) \leq \frac{|V|}{2} - \frac{o(G-U)-|U|}{2}$ by noticing that $o(G - U) - |U|$ is the number of nodes from the odd components in $G - U$ that must remain unmatched.

Therefore, it is sufficient to show that $\nu(G) = \frac{|V|}{2} - \max_{U \subseteq V} \frac{o(G-U)-|U|}{2}$. Any reference to left-hand side (LHS) or right-hand side (RHS) will be in reference to this inequality. Proof via induction on $|V|$. Base case of $|V| = 0$ is trivial.

Case 1: There exists $v \in V$ such that v is in every maximum matching. Let $G' = (V', E') = G - v$, then $\nu(G') = \nu(G) - 1$ and by induction, there is $U' \subseteq V'$ such that the RHS of the formula is equal to $\nu(G') = \nu(G) - 1$. It is easy to verify that $U = U' \cup \{v\}$ satisfies equality in the formula for G.

Case 2: For every $v \in G$, there is a maximum matching that misses it. By Claim 3 below, $\nu(G) = \frac{|V|-1}{2}$ and that there is an odd number of vertices in the entire graph. If we take $U = \emptyset$, then the theorem holds. □

Claim 3 *Let $G = (V, E)$ be a graph such that for each $v \in V$ there is a maximum matching in G that misses v. Then, $\nu(G) = \frac{|V|-1}{2}$. In particular, $|V|$ is odd.*

Proof: G is necessarily connected. By way of contradiction, assume there exists two vertices $u \neq v$ and a maximum matching M that avoids them. Among all such choices, choose M, u, v such that $dist(u, v)$ is minimized. If $dist(u, v) = 1$ then M can be grown by adding uv to it. Therefore there exists a vertex t, $u \neq t \neq v$, such that t is on a shortest path from u to v. Also, by minimality of distance between u and v we know that $t \in M$.

By the assumption, there is at least one maximum matching that misses t. We are going to choose a maximum matching N that maximizes $N \cap M$ while missing t. N must cover u, or else N, u, t would have been a better choice above. Similarly, N covers v. Now $|M| = |N|$ and we have found one vertex $t \in M - N$ and two $u, v \in N - M$, so there must be another vertex $x \in M - N$ that is different from all of the above. Let $xy \in M$. N is maximal, so xy can't be added to it. Thus, we must have that $y \in N$ and that means $y \neq t$. Let $yz \in N$. Then we have that $z \in N - M$ because $xy \in M$ and $z \neq x$.

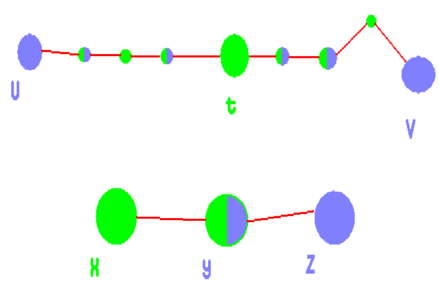

Figure 1: Green vertices are in M. Blue vertices are in N.

Consider the matching $N' = N - yz + xy$. We have that $|N'| = |N|$ and N' avoids t and $|N' \cap M| > |N \cap M|$. This is a contradiction. □

2 Algorithm for Maximum Cardinality Matching

We now describe a polynomial time algorithm for finding a maximum cardinality matching in a graph, due to Edmonds. Faster algorithms are now known but the fundamental insight is easier to see in the original algorithm. Given a matching M in a graph G, we say that a node v is M-exposed if it is not covered by an edge of M.

Definition 4 *A path P in G is M-alternating if every other edge is in M. It can have odd or even length. A path P is M-augmenting if it is M-alternating and both ends are M-exposed.*

Lemma 5 *M is a maximum matching in G if and only if there is no M-augmenting path.*

Proof: If there is an M-augmenting path, then we could easily use it to grow M and it would not be a maximum matching.

In the other direction, assume that M is a matching that is not maximum by way of contradiction. Then there is a maximum matching N, and $|N| > |M|$. Let H be a subgraph of G induced by the edge set $M \Delta N = (M - N) \cup (N - M)$ (the symmetric difference). Note that the maximum degree of a node in H is at most 2 since a node can be incident to at most one edge from $N - M$ and one edge from $M - N$. Therefore, H is a disjoint collection of paths and cycles. Furthermore, all paths are M-alternating (and N-alternating too). All cycles must be of even length, since they alternate edges from M and N too. At least one of the paths must have more N edges than M edges because $|N| > |M|$ and we deleted the same number of edges from N as M. That path is an M-augmenting path. $\square$

The above lemma suggests a greedy algorithm for finding a maximum matching in a graph G. Start with a (possibly empty) matching and iteratively augment it by finding an augmenting path, if one exists. Thus the heart of the matter is to find an *efficient* algorithm that given G and matching M, either finds an M-augmenting path or reports that there is none.

Bipartite Graphs: We quickly sketch why the problem of finding M-augmenting paths is relatively easy in bipartite graphs. Let $G = (V, E)$ with A, B forming the vertex bipartition. Let M be a matching in G. Let X be the M-exposed vertices in A and let Y be the M-exposed vertices in B. Obtain a directed graph $D = (V, E')$ by orienting the edges of G as follows: orient edges in M from B to A and orient edges in $E \setminus M$ from A to B. The following claim is easy to prove.

Claim 6 *There is an M-augmenting path in G if and only if there is an X-Y path in the directed graph D described above.*

Non-Bipartite Graphs: In general graphs it is not straight forward to find an M-augmenting path. As we will see, odd cycles form a barrier and Edmonds discovered the idea of shrinking them in order to recursively find a path. The first observation is that one can efficiently find an alternating *walk*.

Definition 7 *A walk in a graph $G = (V, E)$ is a finite sequence of vertices $v_0, v_1, v_2, \ldots, v_t$ such that $v_i v_{i+1} \in E, 0 \le i \le t - 1$. The length of the walk is t.*

Note that edges and nodes can be repeated on a walk.

Definition 8 *A walk $v_0, v_1, v_2, \ldots, v_t$ is M-alternating walk if for each $1 \le i \le t - 1$, exactly one of $v_{i-1}v_i$ and $v_i v_{i+1}$ is in M.*

Lemma 9 *Given a graph $G = (V, E)$, a matching M, and M-exposed nodes X, there is an $O(|V| + |E|)$ time algorithm that either finds a shortest M-alternating X-X walk of positive length or reports that there is no such walk.*

Proof Sketch. Define a directed graph $D = (V, A)$ where $A = \{(u, v) : \exists x \in V, ux \in E, xv \in M\}$. Then a X-X M-alternating walk corresponds to a X-$N(X)$ directed path in D where $N(X)$ is the set of neighbors of X in G (we can assume there is no edge between two nodes in X for otherwise that would be a shortest walk). Alternatively, we can create a bipartite graph with $D = (V \cup V', A)$ where V' is a copy of V and $A = \{(u, v') \mid uv \in E \setminus M\} \cup \{(u', v) \mid uv \in M\}$ and find a shortest X-X' directed path in D where X' is the copy of X in V'. $\square$

What is the structure of an X-X M-alternating walk? Clearly, one possibility is that it is actually a path in which case it will be an M-augmenting path. However, there can be alternating walks that are not paths as shown by the figure below.

One notices that if an X-X M-alternating walk has an even cycle, one can remove it to obtain a shorter alternating walk. Thus, the main feature of an alternating walk when it is not a path is the presence of an *odd* cycle called a *blossom* by Edmonds.

Definition 10 *An M-flower is an M-alternating walk $v_0, v_1, \ldots, v_t$ such that $v_o \in X$, t is odd and $v_t = v_i$ for some even $i < t$. In other words, it consists of an even length $v_0, \ldots, v_i$ M-alternating path (called the stem) attached to an odd cycle $v_i, v_{i+1}, \ldots, v_t = v_i$ called the M-blossom. The node v_i is the base of the stem and is M-exposed if $i = 0$, otherwise it is M-covered.*

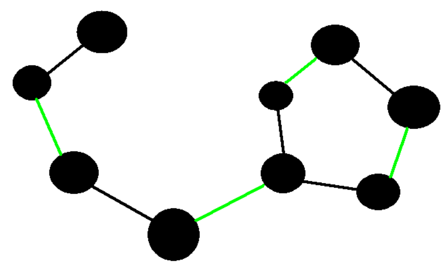

Figure 2: A M-flower. The green edges are in the matching

Lemma 11 *A shortest positive length X-X M-alternating walk is either an M-augmenting path or contains an M-flower as a prefix.*

Proof: Let $v_0, v_1, \ldots, v_t$ be a shortest X-X M-alternating walk of positive length. If the walk is a path then it is M-augmenting. Otherwise let i be the smallest index such that $v_i = v_j$ for some $j > i$ and choose j to be smallest index such that $v_i = v_j$. If $v_i, \ldots, v_j$ is an even length cycle we can eliminated it from the walk and obtain a shorter alternating walk. Otherwise, $v_0, \ldots, v_i, \ldots, v_j$ is the desired M-flower with v_i as the base of the stem. □

Given a M-flower and its blossom B (we think of B as both a set of vertices and an odd cycle), we obtain a graph G/B by shrinking B to a single vertex b and eliminating loops and parallel edges. It is useful to identify b with the base of the stem. We obtain a matching M/B in G/B which consists of eliminating the edges of M with both end points in B. We note that b is M/B-exposed iff b is M-exposed.

Theorem 12 *M is a maximum matching in G if and only if M/B is a maximum matching in G/B.*

Proof: The next two lemmas cover both directions. □

To simplify the proof we do the following. Let $P = v_0, \ldots, v_i$ be the stem of the M-flower. Note that P is an even length M-alternating path and if $v_0 \neq v_i$ then v_0 is M-exposed and v_i is

M-covered. Consider the matching $M' = M\Delta E(P)$, that is by switching the matching edges in P into non-matching edges and vice-versa. Note that $|M'| = |M|$ and hence M is a maximum matching in G iff M' is a maximum matching. Now, the blossom $B = v_i, \ldots, v_t = v_i$ is also a M'-flower but with a degenerate stem and hence the base is M'-exposed. For the proofs to follow we will assume that $M = M'$ and therefore b is an exposed node in G/B. In particular we will assume that $B = v_0, v_1, \ldots, v_t = v_0$ with t odd.

Proposition 13 *For each v_i in B there is an even-length M-alternating path Q_i from v_0 to v_i.*

Proof: If i is even then $v_0, v_1, \ldots, v_i$ is the desired path, else if i is odd, $v_0 = v_t, v_{t-1}, \ldots, v_i$ is the desired path. That is, we walk along the odd cycle one direction or the other to get an even length path. □

Lemma 14 *If there is an M/B augmenting path P in G/B then there is an M-augmenting path P' in G. Moreover, P' can be found from P in $O(m)$ time.*

Proof:

Case 1: P does not contain b. Set $P' = P$.

Case 2: P contains b. b is an exposed node, so it must be an endpoint of P. Without loss of generality, assume b is the first node in P. Then P starts with an edge $bu \notin M/B$ and the edge bu corresponds to an edge $v_i u$ in G where $v_i \in B$. Obtain path P' by concatenating the even length M-alternating path Q_i from v_0 to v_i from Proposition 13 with the path P in which b is replaced by v_i; it is easy to verify that is an M-augmenting path in G. □

Lemma 15 *If P is an M-augmenting path in G, then there exists an M/B augmenting path in G/B.*

Proof: Let $P = u_0, u_1, \ldots, u_s$ be an M-augmenting path in G. If $P \cap B = \emptyset$ then P is an M/B augmenting path in G/B and we are done. Assume $u_0 \neq v_0$ - if this is not true, flip the path backwards. Let u_j be the first vertex in P that is in B. Then $u_0, u_1, \ldots, u_{j-1}, b$ is an M/B augmenting path in G/B. Two cases to verify when $u_j = v_0$ and when $u_j = v_i$ for $i \neq 0$, both are easy. □

Remark 16 *The proof of Lemma 14 is easy when b is not M-exposed. Lemma 15 is not straight forward if b is not M-exposed.*

From the above lemmas we have the following.

Lemma 17 *There is an $O(nm)$ time algorithm that given a graph G and a matching M, either finds an M-augmenting path or reports that there is none. Here $m = |E|$ and $n = |V|$.*

Proof: The algorithm is as follows. Let X be the M-exposed nodes. It first computes a shortest X-X M-alternating walk P in $O(m)$ time — see Lemma 9. If there is no such walk then clearly M is maximum and there is no M-augmenting path. If P is an M-augmenting path we are done. Otherwise there is an M-flower in P and a blossom B. The algorithm shrinks B and obtains G/B and M/B which can be done in $O(m)$ time. It then calls itself recursively to find an M/B-augmenting path or find out that M/B is a maximum matching in G/B. In the latter case, M is a maximum matching in G. In the former case the M/B augmenting path can be extended to an

M-augmenting path in $O(m)$ time as shown in Lemma 14. Since G/B has at least two nodes less than G, it follows that his recursive algorithm takes at most $O(nm)$ time. $\square$

By iteratively using the augmenting algorithm from the above lemma at most $n/2$ times we obtain the following result.

Theorem 18 *There is an $O(n^2m)$ time algorithm to find a maximum cardinality matching in a graph with n nodes and m edges.*

The fastest known algorithm for this problem has a running time of $O(m\sqrt{n})$ and is due to Micali and Vazirani with an involved formal proof appearing in [3]; an exposition of this algorithm can be found in [2].

References

[1] A. Schrijver. Combinatorial Optimization. Springer-Verlag Berlin Heidelberg, 2003

[2] P. Peterson and M. Loui. The General Maximum Matching Algorithm of Micali and Vazirani. *Algorithmica*, 3:511-533, 1998.

[3] V. Vazirani. A Theory of Alternating Paths and Blossoms for Proving Correctness of the $O(|E|\sqrt{|V|})$ General Graph Maximum Matching Algorithm. *Combinatorica*, 14(1):71–109, 1994.