**CS/ISYE/MATH 7510: Graph Algorithms** Spring 2009

# MATCHINGS #1— January 12, 2009

*Lecturer: Robin Thomas* *Scribe: Noah Streib*

**Abstract**

A few theorems on matchings culminating with an algorithm of Edmonds to find a maximum cardinality matching using augmenting paths and blossoms. For a reference, see [1].

## 1 Matching Theorems

**Theorem 1.** *(Hall) Let $G$ be a bipartite graph with bipartition (A,B). There exists a complete matching from $A$ to $B$ if and only if $|N(X)| \geq |X|$ for all $X \subseteq A$.*

**Definition 2.** *Let $G$ be a bipartite graph with bipartition (A,B) and let $M$ be a perfect matching in $G$. We denoted by $D(G,M)$ the digraph obtained by directing all edges from $A$ to $B$ and contracting the edges of $M$.*

Note that given a digraph $D$ this process can be reversed. Create a bipartite graph $G$ as follows: for each vertex $v \in V(D)$, split $v$ into $v^+, v^- \in V(G)$ such that there is a one-to-one correspondence between edges incident with $v^+$ and edges with head $v$ in $D$, and a one-to-one correspondence between edges incident with $v^-$ and edges with tail $v$ in $D$. Then add an edge with ends $v^+$ and $v^-$.

**Definition 3.** *A graph $G$ is strongly $k$-connected for some nonnegative integer $k$ if $G \backslash X$ is strongly connected for each $X \subseteq V$ such that $|X| \leq k-1$.*

**Theorem 4.** *Let $G$ be a connected bipartite graph, let (A,B) be a bipartition of $G$, let $M$ be a perfect matching in $G$, and let $k \geq 0$ be an integer. Then the following are equivalent.*
*(i) every matching of size at most $k$ is contained in a perfect matching,*
*(ii) for every $X \subseteq A$, either $|N(X)| \geq |X| + k$ or $N(X) = B$,*
*(iii) the digraph $D(G,M)$ is strongly $k$-connected.*

*Proof.* Exercise. Use Hall's theorem. □

Note that $M$ is an arbitary matching. Thus, $D(G,M)$ is strongly $k$-connected if and only if $D(G,M')$ is strongly $k$-connected for all perfect matchings $M$ and $M'$.

**Definition 5.** *A graph $G$ is $k$-extendable if every matching of size at most $k$ is contained in a perfect matching. (Note that in a large portion of the literature, $k$-extendable uses "exactly $k$" instead of "at most $k$").*

**Theorem 6.** *(Tutte's 1-factor) A graph $G$ has a perfect matching if and only if $o(G\backslash X) \leq |X|$ for all $X \subseteq V(G)$, where $o(H)$ is the number of components in a graph $H$ that have an odd number of vertices.*

Unfortunately, Tutte's theorem is not directly helpful for finding perfect matchings algorithmically, as we can not test all subsets of $V(G)$ in polynomial time.

**Definition 7.** *A path $P$ in a graph $G$ is $M$-alternating if edges in $P$ alternate between $M$ and $E(G)\backslash M$. The path $P$ is $M$-augmenting if it is $M$-alternating with end-vertices unsaturated by $M$.*

**Theorem 8.** *Let $M$ be a matching in a graph $G$. Then $M$ is maximum if and only if there does not exist an $M$-augmenting path.*

*Proof.* (Sketch) The forward direction is trivial, as we can simply switch the edges along the path to obtain a larger matching. For the converse, go by contradiction and consider the symmetric difference of the perfect matchings. The components have maximum degree two, and thus are paths or cycles. Notice that odd paths are $M$-augmenting. $\square$

## 2 Maximum Cardinality Matching Algorithm

Using the previous theorem, we can find maximum matchings if we can solve the following problem algorithmically.

$M$-AUGMENTING PATH PROBLEM.
Instance. Graph $G$, matching $M$.
Question. Is there an $M$-augmenting path?

**Definition 9.** *Let $G$ be a graph and $M$ a matching in $G$. Let $r$ be an $M$-unsaturated vertex. An $M$-alternating tree rooted at $r$ is a tree $T$ that satisfies the following:*
*(i) $T$ is a subgraph of $G$ containing $r$,*
*(ii) every subpath of $T$ originating in $r$ is $M$-alternating, and*
*(iii) if an edge $e \in M$ is incident with a vertex of $T$ then $e \in E(T)$.*

Let $A(T) := \{v \in V(G) : d_T(r,v) \text{ is odd}\}$, and let $B(T) := \{v \in V(G) : d_T(r,v) \text{ is even}\}$, where $d(u,v)$ is the distance from $u$ to $v$ in the $M$-alternating tree $T$. Then we have the following lemma.

**Lemma 10.** *Let $G$ be a graph with matching $M$, and let $T$ be a maximal $M$-alternating tree. Assume that there does not exist $uv \in E(G)$ such that $u, v \in B(T)$. Then there is no $M$-augmenting path starting at $r$.*

*Proof.* Delete $A(T)$. By maximality, every vertex in $B(T)$ is isolated. The result follows. $\square$

Note that the existence of a $B(T) - B(T)$ edge yields an odd cycle. Thus, the lemma above applies in particular to bipartite graphs. To deal with these odd cycles, Edmonds created the following definition.

**Definition 11.** *Let $k$ be a positive integer. An $M$-blossom is a cycle $C$ of length $2k+1$ containing $k$ edges in $M$ and one $M$-unsaturated vertex.*

Notice that the existence of a $B(T) - B(T)$ edge creates a cycle of length $2k+1$ with $k$ matching edges, and we can swap matching edges on the path from $r$ to $C$ to force the base of the cycle to be $M$-unsaturated, thus obtaining an $M$-blossom.

**Lemma 12.** *Let $M$ be a matching in $G$ and $C$ an $M$-blossom. Then $M$ is a maximum matching in $G$ if and only if $M \backslash E(C)$ is a maximum matching in $G/E(C)$ (where the notation $G/E(C)$ means that we contract all edges of $C$).*

*Proof.* Exercise. □

The following algorithm answers the $M$-AUGMENTING PATH PROBLEM described at the beginning of the section, and in the affirmative case yields the $M$-augmenting path.

Algorithm to find an $M$-augmenting path or to determine there isn't one.
We may assume that there exists an $M$-unsaturated vertex, as else $M$ is a perfect matching and we are done.
STEP 1. For every $M$-unsaturated vertex $r$, find a maximal $M$-alternating tree $T$ rooted at $r$. Check if it has an $M$-augmenting path. Return "yes" and the $M$-augmenting path if it does.
STEP 2. Else, if there does not exist a $B(T) - B(T)$ edge in $E(G)$, then we may apply Lemma 10 above to conclude that there is no $M$-augmenting path in $T$, and we can proceed to the next $M$-unsaturated vertex. If such an edge exists, use the swapping procedure described above to create matching $M'$ with $C$ an $M'$-blossom.
STEP 3. Contract the blossom (keeping in mind Lemma 12) and go back to step 2 using $G/E(C)$ and $M' \backslash E(C)$. (This produces a new matching in $G/E(C)$. We can convert this back to a matching in $G$ by swapping matching edgs along $C$ to make sure that the unsaturated vertex in $C$ is the vertex incident with the matching edge from $E(G) \backslash E(C)$.) Return the $M$-augmenting path in $G$ if one is found.
STEP 4. If we have exhausted all $M$-unsaturated vertices without returning an $M$-augmenting path, then return $M$ and that there is no $M$-augmenting path.

Notice that we can use Edmond's algorithm to proof Tutte's theorem. In each case, $A(T)$ will be the violating set if there does not exist a perfect matching. For example, if there are no $B(T) - B(T)$ edges, then the proof of Lemma 10 tells us that deleting $A(T)$ leaves at least $|A(T)| + 1$ odd components; namely $B(T)$, as each vertex in $B(T)$ is a singleton.

## References


[1] William J. Cook, William H. Cunningham, William R. Pulleyblank, Alexander Schrijver. *Combinatorial Optimization.* John Wiley & Sons, Inc., New York, NY, 1998.