Combinatorial Networks
Week 9, May 13-14

Tutte's Theorem

- **Recall.** Perfect matching (PM) is a matching with $\frac{n}{2}$ edges.
- **Definition.** For a subset $S \subseteq V(G)$, let $q(G-S)$ be the number of odd components in $G-S$.
- **Observation.** For any $S \subseteq V(G)$ and any matching M in G,

$$2|M| \leq |V(G)| - (q(G-S) - |S|) = |V(G)| - q(G-S) + |S|.$$

- **Tutte - Berge Formula.**

$$\max_{M} 2|M| = \min_{M} |V(G)| - (q(G-S) - |S|).$$

- **Tutte's Theorem.** G has PM if and only if $q(G-S) \leq |S|$ for $s \leq V(G)$. (the condition is also called Tutte's condition.)
- **Proof of Tutte's Theorem.** If G has a PM M, then by the Observation,

$$|V(G)| = 2|M| \leq |V(G)| - q(G-S) + |S|. \quad for\ \forall S$$

  Thus $|V(G-S)| \leq |S|$, $\forall S \subseteq V(G)$. Suppose G has no PM, then we want to find a "bad" set S such that $|S| < q(G-S)$. Firstly, assume $|V(G)|$ is even, otherwise, $S = \emptyset$ is bad.

  - **Claim 1.** We may assume: G is edge-maximal with No PM.
    Proof : Let G' be obtained from G by adding edges and G' has a bad S.
    Then S is also bad for G. We have

$$q(G-S) \geq q(G'-S) > |S|.$$

    It is because any odd component of $G'-S$ is a union of some component of $G-S$,and one of them must be odd.
  - **Claim 2.** G has a bad S if and only if$\exists S$ such that all components of $G-S$ are complete and every vertex $u \in S$ is adjacent to all vertices of $G-u$. $(*)$
    Proof : $\Longrightarrow$ it is easy to verify under claim 1.
    $\Longleftarrow$ $\exists S$ with $(*)$, then either S is bad or $|V(G)|$ is odd or G has PM.
    Let S satisfies $(*)$, if $q(G-S) > |S|$, then S is bad, done. So $q(G-S) \leq |S|$ and $|V(G)|$ is even.
    By the property $(*)$, we can construct a PM in G, a contradiction. So claim 2 is proved.

– **Claim 3.** G has a $S \subseteq V(G)$ with property $(*)$.

Proof : Let S be the set of vertices u which is adjacent to all other vertices. So it suffices to show: any component D of $G - S$ is complete.

Suppose D is not, $\exists a, a^{'} \in D$ such that $aa^{'} \notin E(G)$, let P be the shortest path in D from a to $a\prime$, let $P = abc \cdots a^{'}$. Now look at $a, b, c \in D$, where $ac \notin E(G)$. Since $b \notin S$, $\exists d$ such that $bd \notin E(G)$.

By claim 1, $G + ac$ has PM M_1 and $G + bd$ has PM M_2 which show that $M_1 \triangle M_2 =$ a collection of even cycles.

Take $P = d \cdots v$ be a maximal path in G starting at d and containing edges alternating between M_1 and M_2.

If the last edge in P is in $M_1 \Longrightarrow v = b$. Let $c = p + bd$.

If the last edge in P is in $M_2 \Longrightarrow v \in \{a, c\}$. Let $c = p + vb + bd$.

In each case, c is even and the only edge not in G is $bd \Longrightarrow M = M_2 \triangle C$ is PM in G. This is a contradiction. Then, Tutte's Theorem is proved.

Finding max matching in general graphs

Finding max matching in polynomial time in general graphs.

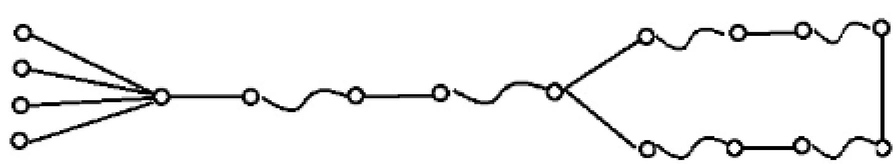
- **Bergs's Theorem.** M is max if and only if G has no M-augmenting path. On the other side, if we find a M-augmenting path P, then $M \triangle P$ is a lager matching.

- **Corollary.** To design a poly time algorithm, it's enough to design a poly-time algorithm that given M, either

  1. find a M-augmenting path, or
  2. "Prove" that there is no M-augmenting path.

  Once we have such algorithm, then repeat this until we end with a max matching after at most $\frac{n}{2}$ steps.

- **Definition 1.** Given M, stem + blossom = flower.



note: $M \triangle S$, flower with $S = \emptyset$, $|M \triangle S| = |M|$.

- **Definition 2.** Given G and a flower $F = (S, B)$ with $S = \emptyset$, let $G \backslash B$ be the graph obtained by contracting B into a new vertex b^*. Let $M \backslash B = M - E(B)$.

- **Fact.** b^* is not matched in $M \backslash B$.

- **Lemma 1.** $\exists M$-augmenting path in G if and only if $\exists(M\backslash B)$-augmenting path in $G\backslash B$. Moreover, give a $(M\backslash B)$-augmenting path, we can easily find a M-augmenting path.

  Proof: $\Longrightarrow \exists M$-augmenting path P in G.

  If $P \cap B = \emptyset$, then P remains the $M\backslash B$-augmenting path in $G\backslash B$. So $P \cap B \neq \emptyset$.

  There is an end v of P which is not in B.

  Let u be the last vertices in P but not in B.

  Then $P^* = vPub^*$ is $M\backslash B$-augmenting path.

  $\Longleftarrow \exists M\backslash B$-augmenting path Q.

  If $b^* \notin Q$, then Q is still M-augmenting path in G. So $b^* \in Q$.

  Then b^* must be an end of Q.

  Since $S = \emptyset$, b is unmatched in G. Then easily we can construct a path Q^* from Q in G with end b, which is M-augmenting path. So the lemma is proved.

  Now we design an algorithm, which for given matching M, either

  (1) Find an M-augmenting path. or

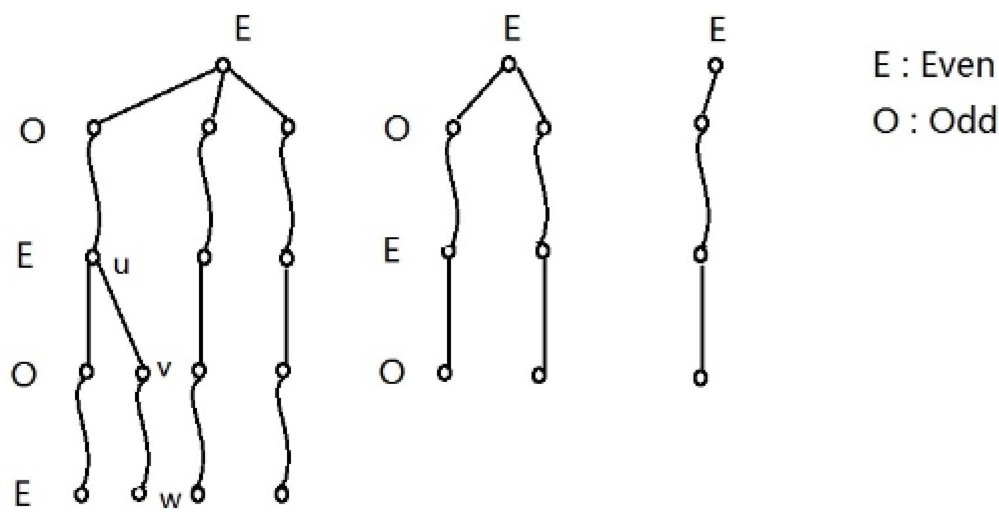
  (2) Find a blossom (with $S = \emptyset$). or

  (3) "Prove" there is no M-augmenting path.

- **Edmond's Algorithm.** Given G and a matching M, it will conduct a collection of vertex-disjoint trees (where vertices are labeled as "even" or "odd") and other vertices, which are unlabeled.

  Initially, all unmatched vertices are labeled as "even", all matched vertices are "unlabeled". While (there is an "even" vertex u such that one of its edges (u, v) is not explored.)

  (A) If v is unmatched, then we have an M-augmenting path from v to the root of its tree.



  (B) If v is matched and unlabeled, let $vw \in M$, then we will add uv, vw to the tree, and label v as "odd", label w as "even".

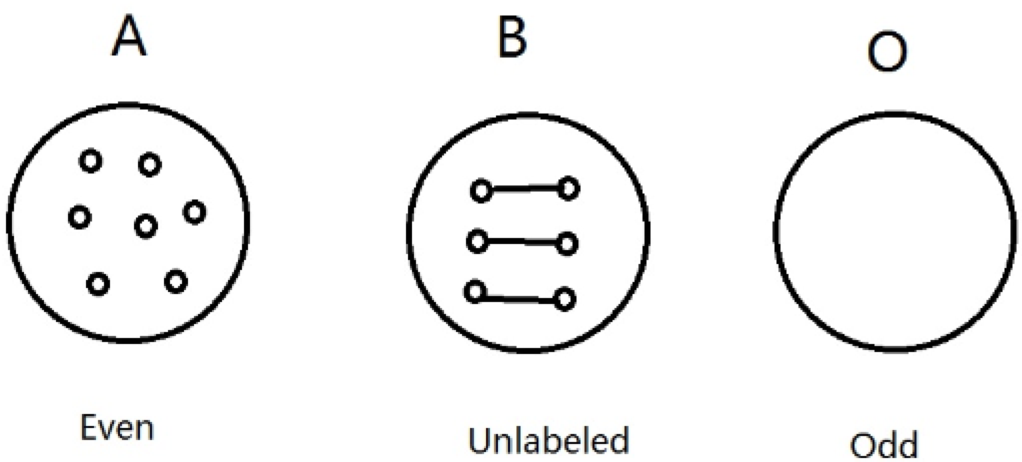
(C) If v is "odd", do nothing.

(D) If v is "even" and is in the same tree as u, then we have a flower.

(E) If v is "even" and is in different tree, then we find an M-augmenting path.

Let $O = \{\text{odd vertices}\}$, $A = \{\text{even vertices}\}$, $B = \{\text{unlabeled vertices}\}$.

- **Remark 1.** When adding an edge vw to a tree, then $vw \in M$ and one vertex is "even", the other is "odd".
- **Remark 2.** No M-augmenting path nor flower if and only if A is independent.
- **Remark 3.** No edges between A and B.



- **Lemma 2.** If the algorithm didn't find M-augmenting path nor flower, then M is maximum.

  Prove:Since unlabeled vertices come with edges of M, by remark 1, $|M| = |O| + \frac{1}{2}|B|$. At the same time, by remark 1,2,3, we have $q(G - O) = |A|$.

  Now, $2|M| = 2|O| + |B| + |A| - |q(G - O)| = |V| + |O| - |q(G - O)|$.

  By Tutte - Berge formula, we know M is maximum. Then lemma 2 is proved.

- **Theorem.** $\exists O(n^2m)$ - time algorithm to find the max matching in general graph.(Hint: Using Edmond's algorithm in $O(n^2)$ rounds.)