An 1.5-approximation for optimal TSP

We improve upon the 2-approximation TSP algorithm that we saw earlier. The new algorithm is based on the observation that one can obtain an Eulerian Tour with a better approximation factor than doubling the MST. The algorithm is due to Christofides [19].

Plan:

- Augment the MST so that it has an Eulerian Tour (A cycle of edges that traverse every edge)

- Now modify this Eulerian tour to be a TSP by deleting repeated vertices.

- We already saw that the above operation does not increase cost under triangle prop.
Fact 1. Any connected graph with all vertex degree equal to an even number has an Eulerian tour which can be found in linear time $O(V+E)$.

Proof. - Start with any vertex and continue traversing edges not visited so far. Because of even parity of degrees, one is guaranteed to leave a vertex if entered except the first one.

- When we reach the first one we discover a tour.

- We may not finish covering all edges. Start a new tour with a vertex that has an unvisited edge and continue.

- At the end, the final tour is the concatenation of all tours found.
- Our goal is to find edges that can augment the MST to have an Eulerian tour.

- From Fact 1, we set for adding edges to the odd degree vertices in MST.

- The edges we add come from a perfect matching.

**Perfect Matching**: Given a graph $G = (V, E)$, a perfect matching $E' \subseteq E$ is a set of edges so that every vertex in $V$ is incident to exactly one edge in $E'$.

- $E' \subseteq E$
- $E'$ is a Matching.

Prove: Every complete connected graph has a perfect matching iff it has even number of vertices.
Lemma! A minimum weight perfect matching in a graph $G=(V,E)$ has weight at most half the cost (weight) of minimum TSP of $G$.

$$\text{Cost (min. Perfect match)} \leq \frac{1}{2} \text{ min. TSP}.$$ 

Proof. - Since $G$ has even number of vertices (otherwise no perfect match could exist), TSP has even vertices.

- Split TSP into two sequence of edges of alternating edges in the tour.

- The sequence with the smaller wt. is $\leq \frac{1}{2} \text{ cost(TSP)}$.

- Alternating sequence of edges constitutes a perfect matching.

- min. wt. perfect matching is even smaller.

- TSP - perfect matching
Algorithm (G is a complete graph)

Step 1. Construct MST $T$ of $G = (V, E)$

Step 2. Let $D \subseteq V$ be the set of odd degree vertices in $T$. Let $G' = (D, E')$ be the subgraph induced by $E$ on $D$.

Step 3. Compute a perfect min. weight matching $E'' \subseteq E'$ in $G'$.

Step 4. Augment $T$ by adding edges $E''$ to $T$.

Step 5. Compute an Eulerian tour in the augmented $T$.

Step 6. Delete repeated vertices in the Eulerian tour to get an approximate TSP.
Theorem. The algorithm computes a 1.5-approximation of min. TSP.

Proof. - The Eulerian tour has edges from MST T and the edges from perfect min. wt. matching computed in Step 3.

- $wt.(T) \leq wt.(TSP)$ because deleting an edge from TSP creates a spanning tree.
- $wt.(\text{Matching}) \leq \frac{1}{2}(TSP)$ by Lemma 1.
- Therefore $wt.(\text{Eulerian tour}) \leq 1.5 \times (TSP)$.
- Step 6 makes the computed TSP even smaller than the Eulerian tour.