# Steiner Tree Approximation

ECE 1762 Algorithms and Data Structures
Fall Semester, 2013

[**Approximation Algorithms, Steiner Tree Approximation**] Given a connected, weighted, undirected graph $G = (V, E)$ and a subset $R \subseteq V$ (of "required" vertices), a **minimum Steiner tree** of $G$ is a tree of minimum weight that contains all the vertices in $R$. (It may or may not contain the remaining vertices).

Finding a minimum Steiner tree is NP-hard in general. Consider a class of approximation algorithms that uses the following heuristic strategy:

1. Compute the complete distance graph $G_1 = (R, R \times R)$ between vertices in $R$; each edge $(u, v)$ in $G_1$ is weighted with the length of the shortest path from $u$ to $v$ in $G$.
2. Compute a minimum spanning tree $G_2$ of $G_1$.
3. Map the graph $G_2$ back into $G$ by substituting for each edge of $G_2$ a corresponding shortest path in $G$. Call the resulting graph $G_3$.
4. Compute a minimum spanning tree $G_4$ of $G_3$.
5. Iteratively delete all leaves in $G_4$ that are not vertices in $R$.

Of all the minimum Steiner trees for $G$ and $R$, let $T_{opt}$ be the one with the minimum number of leaves. Let $T_{approx}$ be the Steiner tree obtained using the strategy outlined above. If $w(T)$ denotes the total cost of a tree $T$ (i.e. the sum of all its edge weights), prove that $w(T_{approx}) \leq 2(1 - \frac{1}{l})w(T_{opt})$, where $l$ is the number of leaves in $T_{approx}$.
*Hint*: Consider a clockwise circular traversal of $T_{opt}$; such a traversal will start and end at the same vertex and will go along each edge in $T_{opt}$ exactly twice.

**[Solution]**

Traversing $T_{opt}$ in, say, the clockwise direction yields a circuit $C$ that goes over each edge in $T_{opt}$ exactly twice. Therefore, the cost of $C$ is $2w(T_{opt})$. We can partition $C$ into segments, such that each segment starts and ends at a leaf and does not contain any leaves in the middle. Now, delete the longest such segment from $C$. The remainder (say, $C'$) of $C$ still traverses every edge of $T_{opt}$. Since the average cost of a segment of $C$ is $2w(T_{opt})/l$ (and the length of the deleted segment is at least as much as the average), the cost of $C'$ is at most $2w(T_{opt})(1 - \frac{1}{l})$.

Now, if a segment in $C'$ runs between required segments $u$ and $v$, its cost is at least as much as that of the shortest path between $u$ and $v$. Thus, if we replace each segment in $C'$ with the edge between its end points in $G_1$, we obtain a graph that spans $G_1$ and whose length is at most $2w(T_{opt})(1 - \frac{1}{l})$. $G_2$ is a spanning tree of $G_1$ and so its cost does not exceed the cost of $G_1$. The transformations of steps 3 to 5 also do not increase the cost of the graph. Thus, the heuristic yields a Steiner tree whose cost is at most $2w(T_{opt})(1 - \frac{1}{l})$.