Approximation Algorithms Lecture 6

Contents

1 Recap

Steiner tree problem

2 Content

2.1 Travelling Salesman Problem (TSP)

Variant 1: Given an undirected graph $G = (V, E)$ and a function $\ell : E \to \mathbb{R}^+$, find a cycle (need not be simple) in G which visits all vertices and has the smallest length.

Variant 2: Given a complete graph G on n vertices, and a function $\ell : E \to \mathbb{R}^+$, find a simple cycle in G which visits all vertices and has the smallest length.

Both variants are interreducible to one another.

Claim 0.1

TSP is NP-hard.

Proof.
For Variant 1:
Claim: Hamiltonian Cycle $<_P$ TSP (i.e., a polynomial time algorithm for TSP gives a polynomial time algorithm for Hamiltonian cycle).

Recall that the HCP asks for a simple cycle which includes all vertices in a graph. Define the input to TSP as (G, ℓ) where $\ell(e) = 1$ if $e \in E$. If the answer to the TSP is n, then the graph has a Hamiltonian cycle. If more, then it doesn't.

For Variant 2: Replace the condition with $\ell(e) = 1$ if $e \in E$, 2 otherwise. $\square$

Claim 0.2

It is NP-hard to approximate TSP to within any approximation factor α.

Proof. We shall show that if there exists a polynomial time α-approximation for TSP, then there is a polynomial time algorithm for HCP.

Call α-TSP the algorithm which gives a solution within α times the optimum.

Let $G' = (V, V \times V)$, and $\ell(e) = 1$ if $e \in E$ and αn if $e \notin E$. If G is Hamiltonian, then an optimal solution to (G', ℓ) has value $= n$. So the solution returned by the α-TSP has length $\leq \alpha n$. So this solution can't have any edge that is not in E. Hence this solution is a Hamiltonian cycle in G and has value n (which is in fact the minimum possible). Hence, if the α-TSP algorithm doesn't return n, then there is no solution. And if the α-TSP algorithm returns n, we are clearly done (trivial bounding). $\square$

Gap Instance: An instance where the optimum solution and the second-best solution have a large gap (say γ). Running an α-competitive algorithm gives an answer within α of the optimal answer, however if $\gamma > \alpha$, the algorithm is forced to output the best solution, which is why these instances can be important.

Note that this reduction breaks down when we require that ℓ is a metric.

2.2 Metric TSP

The same problem as above, but ℓ is a metric.