More Approximation Algorithms

CS255
Chris Pollett
May 10, 2006.

Outline

- General TSP
- Randomized Approximation Algorithms

General TSP

Theorem. If **P** $\neq$ **NP**, then for an constant $d \geq 1$, there is no p-time approximation algorithm with approximation ratio d for general *TSP*.

Proof. Suppose that for some number $d \geq 1$, there was an approximation algorithm A for general *TSP* with the given approximation ratio. Without loss of generality, we can assume d is an integer. We will then show how to use A to solve instances of *HAM-CYCLE*. Since *HAM-CYCLE* is **NP**-complete, this will imply the result…

More Proof

Let $G = (V, E)$ be an instance of the *HAM-CYCLE* problem. Let $G' = (V, E')$ be the complete graph on V. Assign a cost to each edge in E′ as follows:

$$c(u,v) = \begin{cases} 1 & \text{if } \{u,v\} \in E \\ d|V| + 1 & \text{otherwise.} \end{cases}$$

This instance (G', c) of the *TSP* optimization problem can be created in p-time in the *HAM-CYCLE* instance length. If the original graph has a hamiltonian cycle, then there is a tour following its edges of cost $|V|$. On the other hand, if no such tour exists, then an tour uses at least one edge not in E, so has cost $(d|V| + 1) + (|V| - 1) > d|V|$. Since our approximation algorithm needs to find a tour within a factor of d of the smallest one, if there is a hamiltonian cycle in G when we run A the tour output will have cost $\leq d|V|$. On the other hand, if the graph G does not have a hamiltonian cycle our algorithm on this instance will return a value $> d|V|$.

Randomized Approximation Algorithms

- We say a randomized algorithm for a problem has an **approximation ratio** of $r(n)$ if for any input size n, the expected cost C of the solution produced by the randomized algorithm is within a factor of $r(n)$ of the cost C^* of an optimal solution.
- We call a randomized algorithm that achieves an approximation ratio of $r(n)$ a **randomized $r(n)$-approximation algorithm**.
- Let *MAX-3SAT* be the problem of determining given a 3-*CNF* formula an assignment which makes as many clauses as possible evaluate to 1.

Randomized Approximation Algorithm for MAX-3SAT

Theorem. Given an instance of *MAX-3SAT* with n variables and m clauses, the randomized algorithm that independently sets each variable to 1 with probability 1/2 and to 0 with probability 1/2 is an randomized 8/7-approximation algorithm.

Proof. Define the indicator random variable $Y_i = I\{\text{clause } i \text{ is satisfied}\}$. Since no literal appears more than once in the same clause, and since we assume that no variable and its negation appear in the same clause, the settings of the three literals are independent. A clause in not satisfied only if all three of its literals are set to 0. We thus have

Pr{clause i is not satisfied } =1/8.
Pr{clause i is satisfied} = 7/8.
$E[Y_i] = 7/8$.

Let $Y = \sum_i Y_i$. Then

$$E[Y] = E[\sum_i Y_i] = \sum_i E[Y_i] = \sum_i 7/8 = 7m/8.$$

As m is an upper bound on the number of possible clauses that could be satisfied, this gives the result.