CSC 373 - Algorithm Design, Analysis, and Complexity **Summer 2016**

Lalla Mouatadid

Some Inapproximability Results

Being exposed to approximibility, it is natural to ask whether we can approximate every NP-hard problem, or how well we can hope to approximate any problem is polynomial time. In this lecture, we will show that some problems do **not** have a polynomial-time c-approximation algorithm for any constant $c \geq 1$, **unless** $P = NP$. We will also see how some reductions do not preserve approximations.

We first consider the **Travelling Salesperson Problem** (TSP). Before presenting the problem formally, we need to define a Hamilton Cycle and the **Hamilton Cycle Problem**:

Given a graph $G(V, E)$, a **Hamilton Cycle** of G is a cycle in the graph that starts at an arbitrary vertex v, visits every vertex in G exactly once then returns to v.

Problem: The Hamilton Cycle Problem (the decision problem)
Input: A graph $G(V, E)$.
Problem: Return 1 iff G has a Hamilton cycle.

Recall a complete graph is just a graph whose vertices are **all** pairwise adjacent: a clique on n vertices.

We state the following theorem without proof[1]:

Theorem 1. *The Hamilton Cycle Problem is NP-Complete.*

The Travelling Salesperson Problem is defined as:

The Travelling Salesperson Problem
Input: An edge-weighted complete graph $G(V, E, w)$ where $w : E \to R^+$
Problem: Return a Hamilton Cycle $\mathcal{C}$ on G where $\sum_{e \in \mathcal{C}} w(e)$ is minimized.

Clearly, by Theorem 1, the TSP is NP-hard (the optimization problem), however as we will show next, it cannot be approximated in polynomial time by any constant $c \geq 1$, unless $P = NP$.

How can we prove such a claim: By contradiction! If TSP has no constant c factor approximation algorithm, then there is **no** algorithm A such that:

$$OPT(G) \leq A(G) \leq c \cdot OPT(G)$$

for any graph G. We will show that if we do have such an algorithm, then we can solve the Hamilton Cycle Problem is polynomial time, thus contradicting Theorem 1.

Proof. Suppose there exists a constant c factor approximation algorithm A for TSP. This means (since TSP is a minimization problem) that given a graph $G(V, E)$, algorithm A outputs a Hamilton cycle of total weight **at most** $c \cdot OPT(G)$ (by definition of c-approximation).

Now suppose we're given an instance of the Hamilton Cycle Problem, a graph $G(V, E)$. We construct a graph $G'(V, E')$ where G' is the complete graph on V the vertex set of G, and we assign positive weights to the edges $e \in E'$ as follows:

[1] Since you already proved it in A3.

$$w(e) = \begin{cases} 1 & \text{if } e \in E \\ cn+1 & \text{if } e \notin E \end{cases}$$

Let's run algorithm A on G'. Since A is a c-approximation, we know that:

$$OPT(G') \leq A(G') \leq c \cdot OPT(G')$$

Now consider the graph G. We have two possible cases: Either G has a Hamilton cycle or it does not.

1. If G has a Hamilton Cycle, then by our above construction, G' has a TSP solution of value n, since we're only using edges with weight $w(e) = 1$. Thus the optimal solution to $A(G')$ is n. Therefore if $A(G')$ output a cycle with value at most cn then there **must be** a Hamilton Cycle in G.

2. If G does not have a Hamilton Cycle, then the algorithm **must** have used one of the edges e where $e \in E' \backslash E$ and $w(e) = cn + 1$. Therefore $A(G')$ would output a Hamilton Cycle with total value **at least** $cn + 1$, grater than the c-approximation !

Therefore to check if G has a Hamilton Cycle in poly-time, we just need to run $A(G')$ and check the value of the output. Ha! A polynomial time algorithm for the Hamilton Cycle decision problem. So unless $P = NP$, the c-approximation algorithm A does not exist. □

This example captures the limitations of approximability. We next show that if even if we have strong duality results, some reductions are **not approximation preserving**. What do we mean by this? Well let's consider two complementary problems we've seen before: Vertex Cover and Independent Set.

Prove to yourself that for any graph $G(V, E)$, G has a vertex cover of size k if and only if G has an indepedent set of size $n - k$. In fact, Gallai proved the following theorem:

Theorem 2. *Let $G(V, E)$, be a graph on n vertices, α the size of the smallest vertex cover and β the size of its largest independent set. Then $n = \alpha + \beta$.*

It follows from Gallai's Theorem that the MIS (maximum independent set) of G is $\beta = n - \alpha$; and we've developed a simple polynomial time 2-approximation algorithm for MVC (minimum vertex cover). So intuitively, we should be able to somehow conclude a constant approximation algorithm out of this for the MIS problem. Let A denote the 2-approximation algorithm for MVC, and let ALG denote the value returned by $A(G)$. We could compute ALG and return $n - ALG$ for the MIS; maybe $n - ALG$ is a good approximation for the MIS. It turns out this does not tell us *anything* useful with respect to the MIS, and here's why:

By the definition of approximation, we know that:

$$\begin{aligned} OPT \leq ALG \leq 2OPT &\implies \alpha \leq ALG \leq 2\alpha \quad \text{(Since } \alpha \text{ is the minimum VC, thus the OPT solution.)} \\ &\implies n - \beta \leq ALG \leq 2(n - \beta) \quad \text{(Using Gallai's equality.)} \\ &\implies n - ALG \leq \beta \end{aligned}$$

So what the last inequality tells us is that $n - ALG$ (which recall is what we hoped to use to approximate the MIS) is upper bounded by the optimal value of the MIS, namely β. Well duh! We already know that; our goal is to approximate how close we are to the optimal β.