Complexity: Approximation Algorithms

Kunwar Shaanjeet Singh Grover

4 Aug 2020

1 Introduction

What happens when you discover a problem is NP Complete or NP Hard? You can try to think of a polynomial time algorithm and try to prove that it gives an answer within some factor of the correct answer. That's what we are going to do today.

We will take a bunch of NP Complete problems and try to come up with some interesting hueristics and try to prove that they give answer within some constant factor.

1.
2. Vertex Cover
3. Set cover
4. Partition

2 Approximation Algos and Schemes

An algorithm for a problem of size n has an approximation ratio $\rho(n)$ if for any input, algo produces a solution with cost C s.t max $\left(\frac{C}{Copt}, \frac{Copt}{C}\right) \leq \rho(n)$

An approximation scheme takes as input $\epsilon < 0$ and for any fixed ϵ, the scheme is a $(1+\epsilon)$ - approximation algorithm.

$O(n^{\frac{2}{\epsilon}})$ running time - Polynomial approx time scheme (PTAS)
Polynomial in n but not necessarily in ϵ

Fully PTAS: poly in n and $\frac{1}{\epsilon}$
Example: $O\left(\frac{n}{\epsilon^2}\right)$

2.1 Vertex Cover

Given: Undirected Graph $G(V, E)$

(Vertex Cover 1)

Problem: Find a subset $V' subset V$ s.t. if (u, v) is an edge of G, then either $u \in V'$ or $v \in V'$ or both. Find V' so $|V'|$ is minimum.

(Vertex Cover 2)

Optimal Solution: $k! = n$
Algo could pick all bottom vertices. Solution:

$$k!\left(\frac{1}{k}+\frac{1}{k-1}+\cdots+1\right)\approx k!\log k$$

2.1.1 Approx Vertex Cover

$$C \leftarrow \phi$$
$$E' \leftarrow E$$

Pick $(u, v) \in E$ arbitarily

$$C \leftarrow C \cup \{u\} \cup \{v\}$$

Delete from all edges incedent on u or v.
Return C

Approx-Vertex Cover is a 2-approximation algo (Is within a factor of 2 from the optimal answer).

Let A denote the edges that are picked. $C = 2|A|$ vertices are picked. Show that $Copt \geq |A|$.

We need to cover every edge in A since they are picked this way. Thus, we need to pick a different vertex from each edge in A. Thus $Copt \geq |A|$.

2.2 Set Cover

Given: A set X and a family of (possibly overlapping) subsets $S_1, S_2, \ldots, S_m \subseteq X$. s.t.

$$\cup_{i=1}^{m} S_i = X$$

Find $C \subseteq \{1, 2, \ldots, m\}$ s.t.

$$\cup_{i \in C} = X$$

while minimizing $|C|$.
(Figure Set Cover)

Optimum Solution: S_3, S_4, S_5.

2.2.1 Approximate Set Cover

This is a $(ln(n) + 1)$ approx algo.

Pick largest S_i, remove all elements of S_i from X and other S_j and repeat.

Proof: Assume there is a cover $Copt, |Copt| = t$. Let X_k be a set of element in iteration $k(X_0 = X)$. $\forall k, X_k$ can be covered by t sets.