CMSC858F: Algorithmic Lower Bounds: Fun with Hardness Proofs
Fall 2014

Instructor: Mohammad T. Hajiaghayi
Scribe: Manish Purohit

October 14, 2014

1 Overview

In this lecture, we continue with some further examples of Approximation Preserving Reductions and Gap Preserving Reductions. In particular, we show that the k-forest problem is as hard as the Minimum k-edge coverage problem. We then show that the APR reduction from Max-3SAT to Independent Set shown previously also proves that Vertex Cover is APX-hard. Further, we show that the Prize Collecting Steiner Tree problem remains APX-hard even on planar graphs (in fact on series-parallel graphs).

2 Hardness of k-Forest

We first introduce the Dense k-subgraph problem.

Dense k-subgraph: Given a graph $G = (V, E)$ and an integer $k \leq |V|$, find a subset $S \subseteq V$ of k vertices (i.e. $|S| = k$) that induces the most number of edges.

- Very poorly understood in terms of approximability
- Best known upper bound is $O(n^{\frac{1}{4}+\epsilon})$ in time $O(n^{\frac{1}{\epsilon}})$ for any constant $\epsilon > 0$ [Bhaskara et al., 2010].
- Best known lower bound is that it does not admit a PTAS unless $\mathsf{NP} \subseteq \cap_{\epsilon>0}\mathsf{BPTIME}(2^{n^{\epsilon}})$ [Khot, 2006].
- Most people believe that the problem should be hard - i.e. there is some constant c such that dense k-subgraph is hard to approximate better than $O(n^c)$.

- Many problems have been proven to be dense k-subgraph hard!

We will now assume that the dense k-subgraph problem is hard ($\Omega(n^c)$) and proceed to prove the hardness of the k-Forest problem. Let us first consider a minimization variant of the dense k-subgraph problem that will be useful in the reduction.

Min k-edge coverage: Given a graph $G = (V, E)$ and an integer $k \leq |E|$, find the smallest subset of vertices that induces at least k edges.

- If there is a polynomial time f-approximation algorithm for minimum k-edge coverage problem, then there is a $2f^2$-approximation algorithm for the dense k-subgraph problem [Hajiaghayi and Jain, 2006].
- Combined with the believed hardness of dense k-subgraph, the above theorem claims that $\exists$ a constant c such that minimum k-edge coverage cannot be approximated better than $O(n^c)$.

2.1 APR reduction from minimum k-edge coverage to k-forest

We first recall the definition of the k-forest problem.

k-Forest: Given a graph $G = (V, E)$, l demand pairs $\{(s_i, t_i)\}_{i=1}^{l}$, and an integer $k \leq l$, find the subgraph $H \subseteq G$ with least number of edges such that at least k of the l demand pairs are connected in H.

Theorem 1 *There is an approximation preserving reduction from minimum k-edge coverage to the k-forest problem.*

Proof: Let $< G = (V, E), k >$ denote an instance of minimum k-edge coverage. We create an instance of k-forest as follows - Create a star with a new vertex r as the root, and every vertex of the graph G as a leaf. Further, for every edge $(u, v) \in E$, create a demand pair (u, v). Refer to Figure 1 for an example.

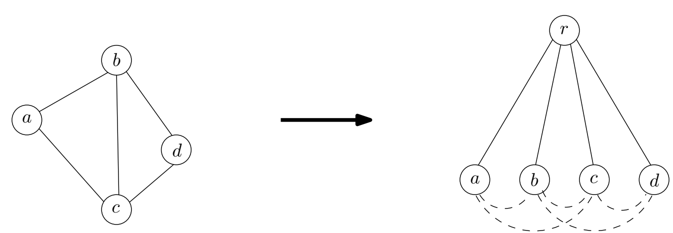

Figure 1: APR Reduction from minimum k-edge coverage to k-forest

Now, suppose that the minimum k-edge coverage instance had a solution of size S. Then the sub-star composed of the root r and the S vertices in the minimum

k-edge coverage solution is a subgraph with S edges that satisfies at least k demand pairs. Similarly, given any solution of the k-forest instance, we can easily obtain a solution to the original minimum k-edge coverage instance of the same cost. ■

3 APX-Hardness of Vertex Cover

Recall the reduction from Max-3SAT to Independent Set seen in the previous lecture. Given a 3-SAT instance ϕ with m clauses, we construct a graph G_ϕ with $3m$ vertices such that $\mathsf{opt}_{max-3sat}(\phi) = \mathsf{opt}_{ind.set}(G_\phi)$.

However, for any graph G, we have that $\mathsf{opt}_{vc}(G) = n - \mathsf{opt}_{ind.set}(G)$ where $\mathsf{opt}_{vc}(G)$ denotes the optimal vertex cover of G and n is the number of vertices of G. Hence, using the same reduction as above, we have that -
$\mathsf{opt}_{vc}(G_\phi) = 3m - \mathsf{opt}_{max-3sat}(\phi)$.

By the PCP theorem, we know that there is a gap between satisfying m clauses and $(1 - \epsilon_0)m$ clauses in a 3-SAT instance. We can now exploit this gap to obtain the APX-hardness of vertex cover as follows -

a) If $\mathsf{opt}_{max-3sat}(\phi) = m$, then $\mathsf{opt}_{vc}(G_\phi) = 2m$.

b) If $\mathsf{opt}_{max-3sat}(\phi) < (1 - \epsilon_0)m$, then $\mathsf{opt}_{vc}(G_\phi) = (2 + \epsilon_0)m$

Hence, it is NP-hard to approximate vertex cover with a factor better than $\frac{2+\epsilon_0}{2} = 1 + \frac{\epsilon_0}{2}$.

4 APX-Hardness of Prize Collecting Steiner Forest in Series-Parallel Graphs

We first recall the definition of the prize collecting steiner forest problem.

Prize Collecting Steiner Forest: Given a graph $G = (V, E)$, costs on edges $c : E \to \mathbb{Z}^+ \cup \{0\}$, l demand pairs $D = \{(s_i, t_i)\}_{i=1}^{l}$, and a penalty function $\pi : D \to \mathbb{Z}^+ \cup \{0\}$ on the demands, the prize collecting steiner forest problem is to find a subgraph H that minimizes $\sum_{e \in E(H)} c_e + \sum_{i \in ViolatedDemands} \pi_i$ where $ViolatedDemands$ is the set of demands that are not satisfied i.e. not connected by H. Essentially, we want a cheap subgraph that connects the demand pairs, however, we are allowed to leave a few pairs disconnected as long as we are willing to pay the appropriate penalty.

We will prove that the prize collecting steiner forest problem is APX hard on series parallel graphs by a reduction from Vertex Cover on 3-regular graphs. We first need the following theorem by Alimonti and Kann [2000] -

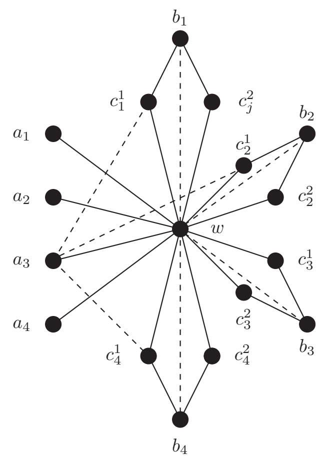

Figure 2: Illustrating the reduction from 3-Regular Vertex Cover to PCSF

Theorem 2 *The minimum vertex cover problem is APX-hard even on 3-regular graphs.*

Theorem 3 (Bateni et al. [2011]) *There is a gap preserving reduction from vertex cover on 3-regular graphs to prize collecting steiner forest on series-parallel graphs.*

Proof: Let $G = (V, E)$ be a 3-regular graph that denotes an instance of vertex cover. Let v_i denote the i^{th} vertex of G and e_j denote the j^{th} edge. Further, since G is 3-regular, we have that $m = 3n/2$.

We now construct an instance $\mathcal{I} = (H, D, \pi)$ of prize collecting steiner forest as follows. The graph H consists of the vertices

- a_i for $1 \leq i \leq n$
- b_j, c_j^1, c_j^2 for $1 \leq j \leq m$
- central vertex w

and the edges

- $\{w, a_i\}$ of cost 2 for $1 \leq i \leq n$
- $\{w, c_j^1\}, \{w, c_j^2\}, \{c_j^1, b_j\}, \{c_j^2, b_j\}$ of cost 1 for $1 \leq j \leq m$

The demands D consist of -

- $\{w, b_j\}$ with penalty 3 for $1 \leq j \leq m$
- $\{a_i, c_j^l\}$ with penalty 1 if a_i and c_j^l correspond to the same vertex $(\forall i, j, l)$

Figure 2 provides an example of this reduction.

We now prove that -
(1) Given a vertex cover of size k for G, a solution for the prize collecting steiner forest instance of total cost $2m + 2n + k$ can be constructed, and
(2) Given a solution to the prize collecting steiner forest instance of total cost at most $2m + 2n + k$, a vertex cover of G of size at most k can be constructed

Proof of (1): Let C be a vertex cover of G of size k. Let T be a tree in H that contains

- edge $\{w, a_i\}$ if and only if $v_i \notin C$
- edges $\{w, c_j^1\}, \{c_j^1, b_j\}$ if and only if $e_j^1 \notin C$
- edges $\{w, c_j^2\}, \{c_j^2, b_j\}$ if and only if $e_j^1 \in C$

Now, the edge costs of T are exactly $2m + 2(n - k)$. Further, observe that all demands $\{w, b_j\}$ are satisfied by T. Further all the three demands for each vertex $v_i \notin C$ are satisfied. Hence, T pays a penalty of at most $3k$ that leads to a total cost of at most $2m + 2n + k$.

Proof of (2): Left as an exercise

Proof that gap is preserved: We still need to prove that the gap is preserved. Since G is 3-regular, we know that the vertex cover of G must be at least $\frac{n}{3}$ and thus there is a hard instance of the problem for which we cannot distinguish between the following two cases unless P = NP:

Case 1: The graph G has a vertex cover of size cn for some constant $c > 0$
Case 2: Every vertex cover of the graph G has size $\geq c(1 + \epsilon_0)n$, for some constant $\epsilon_0 > 0$

Now, by the above reduction, we cannot distinguish between the following two cases unless P=NP:

Case 1: The instance (H, D, π) has a PCSF of cost $2m + 2n + cn = (5 + c)n$
Case 2: The instance (H, D, π) has a PCSF of cost at least $2m+2n+c(1+\epsilon_0)n = (5 + c(1 + \epsilon_0))n$

Hence, the reduction is Gap preserving and prize collecting steiner forest in series-parallel graphs is hard to approximate within a factor $\frac{5+c(1+\epsilon_0)}{5+c} = 1+\epsilon_1$.
■

References

P. Alimonti and V. Kann. Some apx-completeness results for cubic graphs. *Theoretical Computer Science*, 237(1):123–134, 2000.

M. Bateni, C. Chekuri, A. Ene, M. T. Hajiaghayi, N. Korula, and D. Marx. Prize-collecting steiner problems on planar graphs. In *Proceedings of the* *twenty-second annual ACM-SIAM symposium on Discrete Algorithms*, pages 1028–1049. SIAM, 2011.

A. Bhaskara, M. Charikar, E. Chlamtac, U. Feige, and A. Vijayaraghavan. Detecting high log-densities: an o (n 1/4) approximation for densest k-subgraph. In *Proceedings of the 42nd ACM symposium on Theory of computing*, pages 201–210. ACM, 2010.

M. T. Hajiaghayi and K. Jain. The prize-collecting generalized steiner tree problem via a new approach of primal-dual schema. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 631–640. ACM, 2006.

S. Khot. Ruling out ptas for graph min-bisection, dense k-subgraph, and bipartite clique. *SIAM Journal on Computing*, 36(4):1025–1071, 2006.