

Practice Problems on Approximation Algorithms

October 29, 2002

Below are two practice problems related to approximation algorithms.

The front page has the problems and the rest gives the solutions. You can use these solutions as a guide as to how you should write-up your solutions. These problems will be more valuable to you if you first solve them and then check the solutions.

Practice Problems

1. An **independent set** of a graph $G = (V, E)$ is a subset $V' \subseteq V$ of vertices such that each edge in E is incident on at most one vertex in V' . The **independent set problem** is to find a maximum-size independent set in G . Recall that a **clique** of a graph $G = (V, E)$ is a subset $V' \subseteq V$ of vertices, each pair of which is connected by an edge in E . The **clique problem** is to find a maximum-size clique in G .

Suppose that you were given an approximation algorithm A for the independent set problem that has a ratio bound of two. Describe how you can use A to obtain a 2-approximation algorithm for the clique problem.

2. In this problem we consider the optimization version of the set covering problem where your goal is to find a subset F of \mathcal{F} of minimum cardinality for which $\cup_{f \in F} f = X$. In this problem you are to design an approximation algorithm that has a ratio bound of 3 for a special case of the set covering problem where each object is contained in at most 3 sets. Be sure to clearly state your algorithm (code not required), argue that it runs in polynomial time, and prove that it has a ratio bound of 3.

Hint: Think about the relationship between the set cover problem and the vertex cover problem discussed in class along with the approximation algorithm for the optimization version of the vertex cover problem.

Solutions (solve the problems before reading this)

1. Suppose that you were given a 2-approximation algorithm A for the independent set problem. We use show $\text{CLIQUE} \leq_p \text{INDEPENDENT-SET}$ and use this reduction to construct an approximation algorithm for clique using approximation algorithm A . Let $G = (V, E)$ be the input for clique and let $S \subseteq V$ be a maximum sized clique in G . In our transformation, we use the complementary graph \overline{G} as the input for independent set. We now argue that S is a maximum sized independent set in G . By definition of S being a clique in G it means that for all $u, v \in S, \{u, v\} \in E$. By definition of \overline{G} , for all $u, v \in S, \{u, v\} \notin E$ and hence S is an independent set in \overline{G} . We now prove that there is no larger independent set in \overline{G} . Suppose not. Let S' be an independent set in \overline{G} where $|S'| > |S|$. However, if S' is an independent set in \overline{G} then this implies that for all $u, v \in S', \{u, v\} \notin E$. Then it follows that for all $u, v \in S', \{u, v\} \in E$ and hence S' is a clique in G which contradicts that S was a maximum clique.

We now construct a clique approximation algorithm, **Clique-approx** by using this reduction to transform the input G for clique to an input \overline{G} for independent set. We then apply approximation algorithm A for independent set to \overline{G} . Let $S \subseteq V$ be the independent set returned by A . Finally, we return S as the output for **Clique-approx**. We now prove that **Clique-approx** is a 2-approximation algorithm. Let S_* be a maximum sized clique in G and let $C_* = |S_*|$. From the proof above we know that S_* forms an independent set in \overline{G} . Hence, since A is a 2-approximation it immediately follows that $|S| \geq C_*/2$. Since S is returned it then follows that $C_*/S \leq 2$ as required for proving that **Clique-approx** is a 2-approximation algorithm for clique.

NOTE: There is NOT a known 2-approximation algorithm for independent set (or clique).

2. Here we use the following simple heuristic of repeatedly picking an uncovered object and placing all sets that contain it in the cover.

```
Let  $C \leftarrow \emptyset$ 
While there is an uncovered object in  $X$ 
    Pick any  $x \in X$  that is uncovered
    Let  $C = C \cup \{F \mid F \in \mathcal{F} \text{ and } x \in F\}$ 
Output  $C$ 
```

Clearly this procedure runs in polynomial time. We now argue that it has a ratio bound of 3. Let A denote the set of objects picked at the top of the while loop. Since each object is only contained in three sets of \mathcal{F} it immediately follows that $|C| \leq 3|A|$. Now the optimal cover, C^* , must include at least one of the at most 3 sets that contain the object in A . Furthermore, note that no two objects in A are both contained in the same set (easily proven by induction) and thus $|A| \leq |C^*|$. Putting these two inequalities together we get that $|C|/3 \leq |A| \leq |C^*|$ and thus $|C| \leq 3|C^*|$ as desired.