Algorithms and Complexity April 4, 2017
UT Austin CS 331
Prof. Dana Moshkovitz Problem Set 6

Problem Set 6

This problem set is due at **10:00 am** on **Tuesday, April 11th**.

Problem 6- 1: Job assignment revisited

You are given a matrix M such that $M_{(w,\ t)} =$ percentage of task t that worker w completes in an hour. No worker can work for more than 10 hours, multiple workers can be assigned to the same task and workers can be assigned to multiple tasks. Write a linear program to find an assignment of workers to tasks which minimizes the total time taken to complete all the tasks, assume that it is possible to complete all the tasks. (Note: You do not need to write an algorithm to do this, just give a linear programming formulation of the problem.)

Solution:

Let $x_{w,t}$ be the amount of hours worker w spends on task t. The problem can be formulated as follows:

Minimize

$$\sum_w \sum_t x_{w,t}$$

Subject to

$$\begin{aligned}
-\sum_t x_{w,t} &\geq -10 && \forall w \\
\sum_w M_{(w,t)} x_{w,t} &\geq 100 && \forall t \\
-\sum_w M_{(w,t)} x_{w,t} &\geq -1 && \forall t \\
x_{w,t} &\geq 0 && \forall w, t
\end{aligned}$$

Problem 6- 2: Reductions and algorithms

For the following two problems, suppose that given a graph $G = (V, E)$ it is *not* possible to determine the existence of an independent set of size $|V|/(\log^3 |V|)$ in polynomial time.

- Is it possible to determine the existence of a vertex cover of size $|V| - 1000$ in polynomial time? If yes, give a polytime algorithm, if no, reduce the problem of finding an independent set of size $|V|/(\log^3 |V|)$ to the problem of finding a vertex cover of size $|V| - 1000$.

  Solution:

  Yes, it's possible to determine the existence of a vertex cover of size $|V| - 1000$ in polynomial time. We proved in class that finding a vertex cover of size $|V| - 1000$ is reducible to finding an independent set of size 1000 We can brute force check all combinations of 1000 vertices against the set of edges for each combination to see if they form an independent set in polynomial time. Consider a function that, given a set of 1000 vertices, checks each edge to make sure that no two vertices in the set are connected by an edge. This function is $O(|E|)$. We need to call that function once for each combination of 1000 vertices.
  Definition of combination: $\binom{n}{k} = \frac{n!}{k!(n-k)!}$. This gives us

  $$\tbinom{n}{1000} = \frac{n!}{1000!(n-1000)!} = \frac{n(n-1)(n-2)\ \ldots\ (n-999)}{1000!} \leq \frac{n^{1000}}{1000!}$$

  There are less than $\frac{n^{1000}}{1000!}$ combinations. Thus, the time complexity to check each combination against the set of edges is $O(|V|^{1000} \times |E|)$. $|E|$ is bounded by $|V|^2$, so the time complex to brute force check for an independent set of size 1000 is $O(|V|^{1002})$. Consider an algorithm that first generates all 1000 possibilities using 1000 for-loops. Within the final for-loop, the algorithm calls the aforementioned function.

- Is it possible to determine the existence of a clique of size $|V|/(\log^3 |V|) + 1000$ in polynomial time? If yes, give a polytime algorithm, if no, reduce the problem of finding an independent set of size $|V|/(\log^3 |V|)$ to the problem of finding a clique of size $|V|/(\log^3 |V|) + 1000$.

  Solution:

  Independent Set $\leq_p$ Clique

  Let G be an arbitrary graph $G = (V, E)$
  Create $G' = (V', E')$ s.t. $V' = V$ and $E' = V \times V - E$

  There is an independent set S of size k for any $k \leq |V|$ on graph G, if and only if there is a corresponding clique S' of size k on graph G'. This is true by the definition of independent set, clique and the formation G'.

By definition of independent all vertices in S do not have an edge between any two pairs of vertices. G' is formulated so that all pairs of vertices that did not have an edge between them in G has an edge between them in G', and all vertices that had an edge betweem than in G does not have an edge between them in G'. The same vertices of S in G' (denoted as S') is a clique in G' since there is an edge form each vertex in S' to every other vertex in S'.

Independent Set of size $|V|/(\log^3 |V|)$ is reducible in polynomial time to a Clique of size $|V|/(\log^3 |V|)$.

(a) Let k specified in the statements above be $|V|/(\log^3 |V|)$

(b) Formulate G' as specified above

(c) Call the black box/oracle to find a clique with G' and size k

(d) Return the oracle answer for clique on G' as the answer for independent set.

This reduction is polynomial time because 1. setting a variable is constant time, 2. formulating G' is polynomial time $O(|V| + |E|)$, 3. there is one call to the black box.

Independent Set of size $|V|/(\log^3 |V|)$ is reducible in polynomial time to a Clique of size $|V|/(\log^3 |V|) + 1000$.

(a) Let k specified in the statements above be $|V|/(\log^3 |V|)$

(b) Formulate G' as specified above

(c) Add 1000 nodes to G' and connect them with every other node (including the 1000) in G'

(d) Call the black box/oracle to find a clique with G' and size $k + 1000$

(e) Return the oracle answer for clique on G' as the answer for independent set.

This reduction is polynomial time because 1. setting a variable is constant time, 2. formulating G' is polynomial time $O(|V|+|E|)$, 3. Adding 1000 nodes and connecting them $(1000 \times (|V| + 999)$ edges) is polynomial time, 4. there is one call to the black box.