Tsinghua University

Discrete Optimistic

Fall 2024

Homework 1

Lin Zejin May 10, 2025

- **Collaborators:** I finish this homework by myself.

Problem 1. (a) When OPT $\geq c$, assume with $\frac{1}{T}$ algorithm A outputs a solution of value at least s. $T \in O(poly(n))$ Run algorithm A for $T \cdot n$ iterations. Then with $(1 - \frac{1}{T})^{Tn} < e^{-n}$ probability, the algorithm A outputs a solution of value less than s.

So with at least $1 - e^{-n}$ probability, the algorithm A outputs a solution of value at least s.

(b)

$$s = \mathbb{E}[outputs] \leq \Pr[outputs \geq s - \frac{1}{n^a}] \cdot poly(n) + (1 - \Pr[outputs \geq s - \frac{1}{n^a}]) \cdot (s - \frac{1}{n^a})$$

Then

$$\Pr[outputs \geq s - \frac{1}{n^a}] \geq \frac{\frac{1}{n^a}}{poly(n) - s + \frac{1}{n^a}} = \frac{1}{n^a(poly(n) - s) + 1}$$

Here we end the proof.

Problem 2.

Problem 3. (a) Let $k = c$, $U = \{1, 2, \cdots, c\}^q$ where q large enough. Introduce

$$S_{i,b} = \{e \in U : e_i = b\},\ i \in [q], b \in [c]$$

Choose $S_{i,t}, 1 \leq t \leq c$ and the coverage is 1.

$x^*_{i,b} = \frac{1}{q}$ will also achieves coverage 1.

Then we cover each $j \in U$ with probability

$$1 - (1 - \frac{1}{c})^c$$

So the expected coverage of rounding is

$$1 - (1 - \frac{1}{c})^c$$

AS c large enough, the expected coverage of rounding is $1 - \frac{1}{e}$.

(b) With instance $k = c$, $U = \{0, 1\}^q$, $n = 2^q$ and

$$S_{i,b} = \{e \in U : e_i = b\},\ i \in [q], b = 0, 1$$

The LP solution $x^*_{i,b} = \frac{1}{q}$.
$\alpha x^*_{i,b} = \frac{(1-\epsilon)\ln n}{q} = (1-\epsilon)\ln 2 < \ln 2$.
Then

$$\Pr[j \text{ is covered}] = 1 - (1-\alpha x^*_{i,b})^q < 1 - (1-\ln 2)^q < 1 - (2^{-1.5})^{\log_2 n} = 1 - n^{-3/2}$$

So

$$\Pr[U \text{ is all covered}] < (1 - n^{-3/2})^n < 1 - n^{-\frac{1}{2}+\epsilon}$$

as n large enough. So the randomized rounding algorithm may not be able to find a feasible solutionwith probability at least $n^{-\frac{1}{2}+\epsilon}$.

Problem 4. (a)
(b) No, since the rounding algorithm gives a solution with expected value large than $(1-\frac{1}{e})$LP. So

$$\text{OPT} \geq (1-\frac{1}{e})\text{LP}$$

always holds.

Problem 5. (a) Suffices to prove the decision problem that if there exists a clique of size k is whether or not NP-complete.
For an 3-SAT instance with clauses $c_1, \cdots, c_m$ and variables $x_1, \cdots, x_n$, we can construct a graph G with vertices $c_1, \cdots, c_m, x_1, \bar{x}_1, x_2, \bar{x}_2, \cdots, x_n, \bar{x}_n$ and additional clauses $x_{j_1} \wedge x_{j_2} \wedge x_{j_3}$ if there is some c_j is composed by $x_{j_1}, x_{j_2}, x_{j_3}$ or some of its nagation. Let $k = 2m + n$
First construct a complete graph.
Remove the edges between x_i and $\bar{x}_i$
Remove those edges that connnects $c_j = a \vee b \vee c$ and $\bar{a} \wedge \bar{b} \wedge \bar{c}$.
For a clause $a \wedge b \wedge c$, which means a, b, c holds in the same time, we remove the edges between $a \wedge b \wedge c$ and those additional clauses that contains one of $\bar{a}, \bar{b}, \bar{c}$
Remove the edges between $a \wedge b \wedge c$ and $\bar{a}, \bar{b}, \bar{c}$
Then, that a graph is clique is equivalent to this conditions:
(1) If $a \wedge b \wedge c$ belongs to it, then additional clauses that contains $\bar{a}, \bar{b}, \bar{c}$ cannot belong to the same clique, and $c_j = \bar{a} \vee \bar{b} \vee \bar{c}$ cannot belong if it exists.
(2) If x_i belongs to it, then $\bar{x}_i$ cannot belong to the same clique and those additional clauses contains $\bar{x}_i$ cannot belong to the same clique.
Certainly, the size of the clique in this graph cannot be larger than $2m + n$ since each clauses c_j corresponds to 8 additional clauses but at most one of them is contained.
Therefore, if $3 - SAT$ is satifiable, then we choose all true variables and all clauses that is true in the solution. Then c_j will be chosen all and exactly one of eight addtional clauses that corresponds to c_j will be chosen, so $k = 2m + n$ is reachable.

If there exists a clique of size $k = 2m + n$, which means choosing n of independent variables, c_j and exactly one of eight additional clauses that corresponds to c_j. Those clauses will be TRUE when the variables that is chosen is assuemd TRUE.

So it is a feasible instance for max-clique problem.

Therefore, It is NP-Hard.

(b) We have proved in (a) that if the instance has optimum 1, then the induced instance has optimum 1.

Now we suffice to prove soundness.