Advanced Algorithms 10/11, 2019

Exercise 04

Lecturer: Mohsen Ghaffari *Teaching Assistant: Vaclav Rozhon*

1 Scheduling with Constant Machines (leftover from last week)

Give a Fully Polynomial-Time Approximation Scheme (FPTAS) for the variant of the minimum makespan scheduling problem in which the number m of machines is a fixed constant.

2 Warm-up [Recommended]

It will be useful to recall the Chernoff bound that says that for independent Bernoulli distributed random variables $X_1, \ldots, X_n$ with $\mathrm{E}[X_i] = p$ we have

$$\Pr\left[|\sum_{1}^{n} X_i - pn| \geq \varepsilon pn\right] \leq 2\mathrm{e}^{-\frac{\varepsilon^2 pn}{3}}$$

for $0 < \varepsilon < 1$.

1. *Target shooting*

   Suppose we have a ground set S and we wish to estimate the size of a subset $T \subseteq S$. We can do it by repeatedly sampling from S: we sample m times a uniform element from S and let X_i to be an indicator for the event of the i-th sample being an element of T. One can use the Chernoff bound to prove that the variable $(X_1 + X_2 + \ldots X_m)/m$ is within $(1+\varepsilon)$ multiplicative error of $|T|/|S|$ with probability at least $1 - \delta$ if we set

   $$m = \Theta\left(\frac{|S|}{|T|}\varepsilon^{-2}\log(1/\delta)\right).$$

   (a) Argue that the lemma really follows from the Chernoff bound.

   (b) Argue that if we sample only $O(\frac{|S|}{|T|})$ elements, with constant probability we never sample an element from T throughout the whole procedure and, hence, we cannot get a reasonable estimate.

2. Suppose that we have a FPRAS that returns value that is in the correct range $[(1-\varepsilon)\mathrm{OPT}, (1+\varepsilon)\mathrm{OPT}]$ with probability $2/3$. Recall the median trick from the lecture and show how to amplify the probability to $1-\delta$ with $O(\log 1/\delta)$ calls of the original algorithm.

3 Counting satisfying assignments

In the lecture we have seen a FPRAS that estimates the number of satisfying assignments of a given DNF formula. Suppose that we wish to solve this problem and we know that one of the clauses contains only 10 variables. Can you propose a faster algorithm than the one you know from the lecture?

4 Counting independent sets [Recommended]

By an independent set of a graph G we mean a subset $S \subseteq V(G)$ of its vertices such that there is no edge with both endpoints in S.

In the lecture we have seen an algorithm that estimates the number of colorings of a given graph if there is a way to sample them. In this exercise we wish to do the same for the problem of counting the number of independent sets. This means following:

You are given a graph G and want to design a FPRAS for the number of independent sets in G. You are given a procedure that takes some graph as an input and outputs a uniformly random independent set from this graph as an answer.