CmpE 587: Intro to Research in TCS **Boğaziçi University, Fall 2014**

Homework 1

Due: October 8, 11:59pm.

Email your PDF as an attachment to cmpe587homework@gmail.com

The attached file should be called `homework1-yourfirstname-yourlastname.pdf`

Homework policy: Please try to do the homework by yourself. If you get stuck, working in a group of two is okay, three at the most. Naturally, acknowledge any sources you worked with at the top of your solutions. LaTeX typesetting with pdf output is mandatory. Questions about the homework, LaTeX, or other course material can be asked on Piazza.

Solve four of the problems #1–5 below (your choice), and also do problem #6.

1. Recall the Min-Vertex-Cover problem: given an undirected graph, select a subset S of vertices so that each edge of the graph touches at least one vertex from S. Recall the Greedy algorithm: repeatedly choose the vertex with highest degree, put it into S, and delete all edges it covers. Recall that an *α-factor approximation algorithm* for Min-Vertex-Cover is one that guarantees to output a vertex cover S with $|S| \leq \alpha \cdot \mathrm{OPT}$, where OPT denotes the cardinality of the smallest vertex cover. Show that for every $\alpha > 1$, the Greedy algorithm is *not* an α-factor approximation algorithm.

   Remark: our definition of the Greedy algorithm is not 100% complete since we didn't specify how to break ties if there is more than one vertex of maximum degree. For your solution, you may help yourself out by assuming that ties might be broken in the worst possible way for the algorithm. For a bonus point, try to solve the problem even assuming that any ties that exist are magically broken in the most *favorable* way for the algorithm.

2. Recall the Max-Coverage problem: Given is a list of m subsets of the elements $\{1, 2, \ldots, n\}$. (Assume each element is in at least one subset.) Also given is a parameter $1 \leq k \leq n$. The task is to choose k of the subsets so that their union covers as many elements as possible. Recall that we showed that the Greedy algorithm for this problem is an efficient $(1 - 1/e)$-factor approximation algorithm.

   The Min-Set-Cover problem is similar to Max-Coverage, except there is no parameter k given in the input. Instead, the task is to choose as *few* subsets as possible so that *all* elements $\{1, 2, \ldots, n\}$ are covered by their union. As usual, an "α-factor approximation algorithm" for Min-Set-Cover would mean an algorithm guaranteed to return a collection of at most $\alpha \cdot \mathrm{OPT}$ input subsets which cover all elements, where OPT denotes the smallest possible cardinality of a collection of input subsets covering all elements.

   Suppose that there exists a poly(n)-time $(1 - 1/e + \epsilon)$-factor approximation algorithm for Max-Coverage, for some $\epsilon > 0$. Show that there exists a poly(n)-time $((1 - \delta) \ln n)$-factor approximation algorithm for Min-Set-Cover, for some $\delta > 0$ depending only on ϵ. (For a bonus point, show that we can take $\delta = 2\epsilon$.)

   Remark: As mentioned in class, $(1 - 1/e + \epsilon)$-factor-approximating Max-Coverage is actually known to be NP-hard. In fact, it's also known that $((1-\delta) \ln n)$-approximating Min-Set-Cover is NP-hard. By virtue of this homework problem, the latter result is strictly stronger than the former (do you see why?). It's also known that the natural Greedy algorithm *is* an efficient $(\ln n)$-factor approximation algorithm for Min-Set-Cover.

3. Recall our sketch of the proof that there is an interactive proof system for the Graph-Non-Isomorphism problem. Given two labeled n-vertex graphs G_0, G_1, it involved considering the set

$$S = \{\text{labeled graphs } H : H \cong G_0 \text{ or } H \cong G_1\}$$

where $\cong$ denotes that two graphs are isomorphic. A key claim—which was actually not quite true—was that $|S| = n!$ if $G_0 \cong G_1$ and $|S| = 2n!$ if $G_0 \not\cong G_1$. This is actually not quite true; for example, suppose G_0 and G_1 are both the *complete graph* (in which all pairs of vertices are connected). Then S only contains the complete graph, so $|S| = 1$.

The "problem" occurs if G_0 and G_1 have *automorphisms.* An automorphism of G is a permutation π of the vertices such that $\pi(G) = G$. Suppose we change the definition of S to the following:

$$S = \{(H, \pi) : H \cong G_0 \text{ or } H \cong G_1, \text{ and } \pi \text{ is an automorphism of } H\}.$$

Show that now it *is* true that $|S| = n!$ if $G_0 \cong G_1$ and $|S| = 2n!$ if $G_0 \not\cong G_1$. Also, give a one- or two-sentence explanation for why we still have $S \in \mathsf{NP}$.

4. In the same sketch of the interactive proof system for the Graph-Non-Isomorphism problem, we were a little vague about the "hashing" step. Abstractly, we have a set S with cardinality either K or $K/2$ (think of $K = 2n!$) and the Prover is trying to convince the Verifier that $|S| = K$. Let's say the elements of S are encoded with m-bits strings. Also, let's fix $k \in \mathbb{N}$ so that 2^k is a bit bigger than K; writing $p = K/2^k$, let's say k is chosen so that $\frac{1}{4} < p \leq \frac{1}{2}$.

The rough idea is that the Verifier should pick a "random hash function" $\boldsymbol{f} : \{0,1\}^m \to \{0,1\}^k$, a random string $\boldsymbol{y} \in \{0,1\}^k$, and ask the Prover for an element $w \in S$ such that $\boldsymbol{f}(w) = \boldsymbol{y}$. If $|S| = K/2$ then the probability that the Prover can do it is at most $\frac{1}{2}p$. (You should try to remember why!) On the other hand, if $|S| = K$ and $\boldsymbol{f}$ hashed all elements of S to unique k-bit strings, then the probability that the Prover can do it would be p. However there are two difficulties: (i) perhaps the random hash function has some collisions on S; (ii) if the Verifier really chooses a completely random function $\boldsymbol{f}$, it will take exponential space to write down. In this problem, we (partly) explain how to fix these two difficulties.

Let $\mathcal{F}_{m,k}$ be a *collection* of functions $f : \{0,1\}^m \to \{0,1\}^k$. We will consider randomly choosing one $\boldsymbol{f} \in \mathcal{F}_{m,k}$. We say that $\mathcal{F}_{m,k}$ is a *pairwise independent hash family* if for all distinct $x, x' \in \{0,1\}^m$ and all $y, y' \in \{0,1\}^k$ we have $\mathbf{Pr}[\boldsymbol{f}(x) = y \text{ and } \boldsymbol{f}(x') = y'] = 2^{-2k}$. Intuitively, this says that if you pick a random function from the collection $\mathcal{F}_{m,k}$, not only does each fixed string get hashed to a random string, each fixed pair of strings gets hashed to a random pair of strings. So the process is "somewhat like" picking a completely random function.

Here is a fact that we will prove later in the course:

Fact. *For any $m \geq k$ there is an "efficient" pairwise independent hash family $\mathcal{F}_{m,k}$. In particular, $|\mathcal{F}_{m,k}| = 2^{2m}$, each function in $\mathcal{F}_{m,k}$ has a "name" s which is $2m$ bits long, and the associated function f_s is easy to compute.*

So instead of picking a completely random function, the Verifier instead will pick a random name $\boldsymbol{s} \in \{0,1\}^{2m}$ and tell it to the Prover; i.e., tell the Prover to use the hash function $f_{\boldsymbol{s}}$. This takes care of problem (ii) above. It remains to worry about problem (i).

Prove the following: When $|S| = K$ we have that $\mathbf{Pr}_{\boldsymbol{s},\boldsymbol{y}}[\exists w \in S : f_{\boldsymbol{s}}(w) = \boldsymbol{y}] \geq \frac{3}{4}p$.

Thus the Prover can succeed with probability $\frac{3}{4}p$ when $|S| = K$ but only with probability $\frac{1}{2}p$ when $|S| = K/2$. By trying the whole experiment many times and seeing if the Prover succeeds with probability bigger or smaller than $\frac{5}{8}p$, the Verifier can tell the difference between the two cases with very high probability.

5. In 1997, Håstad showed the following theorem from the field of "Probabilistically Checkable Proofs":

   Håstad's PCP Theorem: For all constants $\epsilon > 0$, there is a "Probabilistically Checkable Proof" system for the SAT problem which works as follows. Given an input CNF formula ϕ of size n, a Prover writes down a certain "proof string" $\pi \in \{0,1\}^m$, where $m \leq \text{poly}(n)$. Then a probabilistic poly(n)-time "Verifier" algorithm acts as follows. First, it reads the input formula ϕ. Next, it flips $\ell \leq O(\log n)$ random coins, obtaining a uniformly random string in $\{0,1\}^\ell$. Based on this random string, it chooses three distinct coordinates $1 \leq \boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k} \leq m$ and a bit $\boldsymbol{b} \in \{0,1\}$. It then reads the three proof-bits $\pi_{\boldsymbol{i}}$, $\pi_{\boldsymbol{j}}$, $\pi_{\boldsymbol{k}}$ and outputs

   $$\text{"YES" if } \pi_{\boldsymbol{i}} + \pi_{\boldsymbol{j}} + \pi_{\boldsymbol{k}} = \boldsymbol{b} \pmod 2,$$
   $$\text{"NO" if } \pi_{\boldsymbol{i}} + \pi_{\boldsymbol{j}} + \pi_{\boldsymbol{k}} \neq \boldsymbol{b} \pmod 2.$$

   The guarantee is: If ϕ is a satisfiable CNF formula then $\mathbf{Pr}[\text{Verifier outputs YES}] \geq 1 - \epsilon$. If ϕ is *not* a satisfiable CNF formula then $\mathbf{Pr}[\text{Verifier outputs YES}] \leq \frac{1}{2} + \epsilon$.

   Show that Håstad's theorem implies the following fact: For all constants $\delta > 0$, it is NP-hard to $(\frac{1}{2} + \delta)$-factor approximate the Max-3Lin(mod 2) problem.

   (Recall the Max-3Lin(mod 2) problem is: Given as input a system of equations, each of the form
   $$x_i + x_j + x_k = b \pmod 2$$
   over n variables $x_1, \ldots, x_n$, find a 0/1-assignment to the variables to maximize the number of satisfied equations.)

6. This problem is a "practice with LaTeX" problem. Your task is to typeset the content of the following two pages yourself, using LaTeX (starting with the section title *An "anticoncentration" theorem*). Make it the last two pages of your homework solution file. Endeavor to match the below typesetting as closely as possible (although if you make your typesetting look *more* attractive than the below, that's okay). You don't have to understand the proof, you just have to typeset it!

An "anticoncentration" theorem

Here is a result in probability theory. It can also be deduced from the Berry–Esseen Theorem, which we will discuss later in class.

Theorem 1. *Assume* $a_0, a_1, \dots, a_n \in \mathbb{R}$ *satisfy*

$$\sum_{j=1}^{n} a_j^2 = 1, \qquad \max_{1\le j\le n}\{\,|a_j|\,\} = \epsilon.$$

Let $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n$ *be i.i.d. random variables, each being* $+1$ *with probability* $\frac{1}{2}$ *and* -1 *with probability* $\frac{1}{2}$*. Then if* $\boldsymbol{X} = a_0 + a_1\boldsymbol{x}_1 + \cdots + a_n\boldsymbol{x}_n$*, it holds that* $\mathbf{Pr}[|\boldsymbol{X}| \le \epsilon] \le 2.74\epsilon$*.*

Proof. The proof is a streamlining of one due to Petrov [Pet95, Theorem 2.14]. It will be convenient to rescale the a_j's so that $\epsilon = 1$; we then want to show

$$\mathbf{Pr}[|\boldsymbol{X}| \le 1] \le \frac{2.74}{\sigma},$$

where $\sigma \coloneqq \sqrt{\sum_{j=1}^{n} a_j^2}$. Define the functions $f, g\colon \mathbb{R} \to \mathbb{R}^{\ge 0}$ by

$$f(x) = \frac{2(1-\cos x)}{x^2}, \qquad g(t) = \begin{cases} 1-|t| & \text{if } |t| \le 1, \\ 0 & \text{else.} \end{cases}$$

(The function f has a removable discontinuity at 0.) Integration by parts shows that f is the inverse Fourier transform of h; i.e.,

$$f(x) = \int_{-\infty}^{\infty} e^{-itx} g(t)\,dt.$$

By considering the first two terms of the Taylor series for $\cos x$ we see that $f(x) \ge \frac{11}{12}$ on $[-1,1]$; hence $\frac{12}{11}f(x) \ge 1_{x\in[-1,1]}$ for all $x \in \mathbb{R}$. We therefore have

$$\begin{aligned}
\mathbf{Pr}[|\boldsymbol{X}| \le 1] &\le \mathbf{E}\left[\tfrac{12}{11}f(\boldsymbol{X})\right] \\
&= \tfrac{12}{11}\,\mathbf{E}\left[\int_{-\infty}^{\infty} e^{-it\boldsymbol{X}} g(t)\,dt\right] \\
&= \tfrac{12}{11}\int_{-\infty}^{\infty} e^{-ita_0} g(t)\,\mathbf{E}\left[e^{-it\boldsymbol{X}'}\right]\,dt \qquad \text{(writing } \boldsymbol{X}' = \boldsymbol{X} - a_0\text{)} \\
&= \tfrac{12}{11}\left|\int_{-\infty}^{\infty} e^{-ita_0} g(t)\,\mathbf{E}\left[e^{-it\boldsymbol{X}'}\right]\,dt\right| \qquad \text{(the quantity is already real and nonnegative)} \\
&\le \tfrac{12}{11}\int_{-\infty}^{\infty} \left|e^{-ita_0}\right| \cdot g(t) \cdot \left|\mathbf{E}\left[e^{-it\boldsymbol{X}'}\right]\right|\,dt \\
&\le \tfrac{12}{11}\int_{-1}^{1} \left|\mathbf{E}\left[e^{-it\boldsymbol{X}'}\right]\right|\,dt, \qquad (1)
\end{aligned}$$

where the last inequality used the fact that $\left|e^{-ita_0}\right| \le 1$, $g(t) = 0$ outside $[-1,1]$, and $g(t) \le 1$

otherwise. But

$$\begin{aligned}
\mathbf{E}\left[e^{-it\boldsymbol{X}'}\right] &= \mathbf{E}\left[\exp\Big(-it\sum_{j=1}^{n} a_j\boldsymbol{x}_j\Big)\right] \\
&= \mathbf{E}\left[\prod_{j=1}^{n}\exp(-ita_j\boldsymbol{x}_j)\right] \\
&= \prod_{j=1}^{n}\mathbf{E}[\exp(-ita_j\boldsymbol{x}_j)] && \text{(independence)} \\
&= \prod_{j=1}^{n}\big(\tfrac{1}{2}\exp(-ita_j)+\tfrac{1}{2}\exp(+ita_j)\big) \\
&= \prod_{j=1}^{n}\cos(a_jt). && (2)
\end{aligned}$$

Substituting (2) into (1) and using $\cos x \le \exp(-\frac{1}{2}x^2)$ for $x \in [-1,1]$ (which can be seen from the Taylor expansion), we get

$$\begin{aligned}
\mathbf{Pr}[|\boldsymbol{X}| \le 1] &\le \tfrac{12}{11}\int_{-1}^{1}\prod_{j=1}^{n}\exp\left(-\tfrac{1}{2}a_j^2t^2\right)\,dt \\
&= \tfrac{12}{11}\int_{-1}^{1}\exp\left(-\tfrac{1}{2}\sigma^2t^2\right)\,dt \\
&\le \tfrac{12}{11}\int_{-\infty}^{\infty}\exp\left(-\tfrac{1}{2}\sigma^2t^2\right)\,dt \\
&= \tfrac{12\sqrt{2\pi}}{11\sigma}\int_{-\infty}^{\infty}\tfrac{1}{\sqrt{2\pi}}\exp\left(-\tfrac{1}{2}u^2\right)\,du && \text{(changing variables: } t = u/\sigma) \\
&= \tfrac{12\sqrt{2\pi}}{11\sigma} \le \tfrac{2.74}{\sigma}. && \square
\end{aligned}$$

References

[Pet95] Valentin Petrov. *Limit theorems of probability theory.* Oxford Science Publications, 1995.