University of Freiburg
Dept. of Computer Science
Prof. Dr. F. Kuhn
S. Faour, M. Fuchs, A. Malyusz

Algorithm Theory
Sample Solution Exercise Sheet 11

Due: Friday, 19th of January 2023, 10:00 am

Exercise 1: Max Cut *(10 Points)*

Let $G = (V, E)$ be a simple undirected graph. Consider the following randomized algorithm: Every node $v \in V$ joins set S with probability 1/2. You can assume that $(S, V \setminus S)$ actually forms a cut i.e., $\emptyset \neq S \neq V$.

(a) Show that with probability at least 1/3 this algorithm outputs a cut which is a 4-approximation to the maximum cut (i.e., the cut of maximum possible size) *(5 Points)*
Hint: Apply the Markov inequality to the number of edges that do ***not*** *cross the cut. For a non-negative random variable X, the Markov inequality states that for all $t > 0$ we have*

$$P(X \geq t) \leq \frac{E[X]}{t}$$

(b) How can you use the above's algorithm to devise a 4-approximation with probability at least $1 - \left(\frac{2}{3}\right)^k$ for any integer $k > 0$? *(4 Points)*

(c) How would you choose k from the previous subtask to make sure your algorithm computes a 4-approximation **with high probability**[1] ? *(1 Point)*

Sample Solution

(a) Let X be random variable that indicates the number of edges that do not cross the cut and let $n = |V|$ and $m = |E|$. Further, let

$$X_e := \begin{cases} 1 & e \text{ is not crossing the cut} \\ 0 & \text{otherwise} \end{cases}$$

Since the endpoints of en edge join the cut independently with probability 1/2, the probability that the end points are in the same cut is also 1/2 and hence $P(X_e = 1) = 1/2$. Further, note that $E[X_e] = 0 \cdot P(X_e = 0) + 1 \cdot P(X_e = 1) = P(X_e = 1)$.

$$E[X] = E\left[\sum_{e \in E} X_e\right] = \sum_{e \in E} E[X_e] = \sum_{e \in E} P(X_e = 1) = \frac{1}{2} \cdot \sum_{e \in E} 1 = \frac{m}{2}$$

Hence, by the Markov inequality

$$P\left(X \geq \frac{3m}{4}\right) \leq \frac{E[X]}{3m/4} = \frac{4m}{6m} = \frac{2}{3}$$

[1]We use the term **with high probability** in the context of graphs with n nodes and for *any* given constant $c > 0$ if the algorithms succeeds with probability at least $1 - \frac{1}{n^c}$.

Let Y be the number of edges that cross the cut. Note that every edge either contributes to X or to Y, thus, $m = X + Y$. It is easy to see that an upper bound on the number of edges that cross the cut is m. This implies that even the max-cut is of size at most m. We denote the size of the max-cut by OPT.

$$P(Y > OPT/4) \geq P(Y > m/4) = P(m - X > m/4) = P(X < 3m/4) = 1 - P(X \geq 3m/4) \geq \frac{1}{3}$$

Hence, the size of the cut $(S, V \setminus S)$ is a 4-approximation to the max cut with probability at least 1/3.

(b) The guarantee a 4-approximation to the max-cut with prob. at least $1-(2/3)^k$ we use the above's algorithm as a 'blackbox', run it k times and output the largest of the k cuts. By the above's analysis, a single pass fails with prob. at most 2/3 to archive a 4-approximation. Hence, the probability that all k runs fail is $(2/3)^k$. In other words, the probability that at least one of the constructed cuts is a 4-approximation to the max-cut is at least $1 - (2/3)^k$.

(c) If we choose $k = \lceil c \cdot \log_{3/2} n \rceil$, the algorithm will output a 4-approximation *with high probability* since we have success probability of $\geq 1-(2/3)^k = 1-(2/3)^{\lceil \log_{3/2} n^c \rceil} \geq 1-(2/3)^{\log_{3/2} n^c} = 1-1/n^c$.

Exercise 2: Balls into Bins *(10 Points)*

Assume we have n bins and n balls (for $n \geq 2$). We now throw all the balls uniformly at random into the bins. In the following we want to show that the maximum number of balls per bin is at most $O(\log n)$ with high probability. For that we define the *maximum load* L by $\max_{1 \leq j \leq n} Y_j$ where (random variable) Y_j stands for the number of balls in bin j.

(a) For a given bin j, what is the expected number of balls in j? (i.e., compute $E[Y_j]$) *(2 Points)*

(b) Use a Chernoff Bound to show that $P(Y_j \geq 2e \cdot \log_2 n) \leq 1/n^{2e}$. *(6 Points)*
Chernoff Bound: Suppose $X_1, X_2, \ldots, X_N$ are *independent* random variables taking values in $\{0, 1\}$. Let X denote $\sum_{i=1}^{N} X_i$ and let $\mu = E[X]$ be this sums expected value. Then for any $\delta > 0$,

$$P\left(X \geq (1+\delta) \cdot \mu\right) \leq \left(\frac{e^\delta}{(1+\delta)^{(1+\delta)}}\right)^\mu$$

(c) Show that the maximum load L is small, i.e., show that $P(L < 2e \cdot \log_2 n) > 1 - \frac{1}{n^4}$. Use a Union Bound! *(2 Points)*

Sample Solution

a) Let X_i for $1 \leq i \leq n$ be the random variable that is 1 if ball i was thrown in bin j and else is 0. By our assumptions we have that for a given j all X_i are independent and $P(X_i = 1) = 1/n$. Thus,

$$E[Y_j] = E\left[\sum_{i=1}^{n} X_i\right] = \sum_{i=1}^{n} E[X_i] = \sum_{i=1}^{n} (0 \cdot P(X_i = 0) + 1 \cdot P(X_i = 1)) = \sum_{i=1}^{n} \frac{1}{n} = 1$$

b) By the previous task we know that $\mu = E[Y_j] = 1$. Also note that by the assumption that $n \geq 2$

we have that $\log_2 n \geq 1$. We can now proof the statement by choosing $\delta = 2e \cdot \log_2 n - 1$:

$$
\begin{aligned}
P(Y_j \geq 2e \cdot \log_2 n) &= P(Y_j \geq (1 + \underbrace{2e \cdot \log_2 n - 1}_{\delta}) \cdot \mu) \\
&\leq \left(\frac{e^{-1} \cdot e^{2e \cdot \log_2 n}}{(2e \cdot \log_2 n)^{2e \cdot \log_2 n}} \right)^1 \\
&= \frac{1}{e} \cdot \left(\frac{e}{2e \cdot \log_2 n} \right)^{2e \cdot \log_2 n} \\
&\leq \left(\frac{1}{2} \right)^{2e \cdot \log_2 n} = \frac{1}{2^{\log_2 (n^{2e})}} = \frac{1}{n^{2e}}
\end{aligned}
$$

c) If there is some j s.t. $Y_j \geq 2e \cdot \log_2 n$, the maximum load is higher than we want. We will use a union bound here:

$$
P(L \geq 2e \cdot \log_2 n) = P\left(\bigvee_{j=1}^{n} (Y_j \geq 2e \cdot \log_2 n) \right) \leq \sum_{j=1}^{n} P(Y_j \geq 2e \cdot \log_2 n) \leq n \cdot 1/n^{2e} = 1/n^{2e-1} \leq 1/n^4
$$