# Randomized Algorithms and Probabilistic Methods: Graded Homework 1

Kevin Klein, collaborators: Ramon Braunwarth, Fabian Gerhardt

October 23rd 2017

## Exercise 1

Let's define $C = \bigcup_v C_v$. If we can find a partition $C = C_A \cup C_B$ such that $\forall v \in A : C_v \cap C_A \neq \emptyset$ and $\forall v \in B : C_v \cap C_B \neq \emptyset$, we have found a trivial admissable coloring. The admissable coloring could be constructed by coloring each node with an arbitrary color from said intersection. We now want to show that given the minimal length for $C_v$, a random partitioning can lead to an admissable coloring.

We assign each $c \in C$ to either $C_A$ or $C_B$ u.a.r. This is a partition as every color is assigned to exactly one set. When inspecting node $v \in A$, we define *success* to be the existence of $c \in C_V \cap C_A \neq \emptyset$. The case for which $v \in B$ follows by symmetry. Success for all nodes implies that each node can chose a color which is not chosen in the class containing its neighboring nodes, i.e. there is an admissable coloring.

$$
\begin{aligned}
X_i &= \textit{success} \text{ for node } i \\
&= \mathbb{1}[C_i \cap C_{class(i)} \neq \emptyset] \\
\Pr[X_i] &= 1 - \Pr[\text{all } c \in C_i \text{ have been mapped to } C - C_{class(i)}] \\
&= 1 - \frac{1}{2^{|C_i|}} \\
X &= \#\textit{successes} \text{ among all } 2n \text{ nodes} \\
&= \sum_{i=1}^{2n} X_i \\
\mathbb{E}[X] &= \mathbb{E}[\sum_{i=1}^{2n} X_i] \\
&= \sum_{i=1}^{2n} \mathbb{E}[X_i] && \text{(by linearity of expectation)} \\
&= \sum_{i=1}^{2n} \Pr[X_i] \\
&= \sum_{i=1}^{2n} (1 - \frac{1}{2^{|C_i|}})
\end{aligned}
$$

We can now make use of our knowledge about the size of each $C_v$.

$$
\begin{aligned}
|C_v| > log_2 n + 1 &\Rightarrow \frac{1}{2^{|C_v|}} < \frac{1}{2^{log_2 n + 1}} = \frac{1}{2n} \\
&\Rightarrow 1 - \frac{1}{2^{|C_v|}} > 1 - \frac{1}{2n}
\end{aligned}
$$

We can now use this this bound to reformulate the expectation.

$$
\begin{aligned}
\mathbb{E}[X] &> \sum_{i=1}^{2n} (1 - \frac{1}{2n}) \\
&> 2n(1 - \frac{1}{2n}) \\
&> 2n - 1
\end{aligned}
$$

By the probabilistic method, there has to be at least one realistion of $X$ attaining value at least $2n$. The latter implies *success* for each node, which, as we have argues, implies the existence of an admissable coloring.

# Exercise 2

Firstly, we observe that there are $\binom{n}{3}$ possible edges in total, $\binom{n-1}{2}$ possible edges containing a node $i$ and $\binom{n-2}{1}$ possible edges containing nodes $i$ and $j$.

$$\begin{aligned}
X_i &= \text{node } i \text{ is isolated} \\
\Pr[X_i] &= \Pr[\text{all possible edges containing } i \text{ have not been added}] \\
&= (1-p)^{\binom{n-1}{2}} \\
&= \mathbb{E}[X_i] \\
X &= \#\text{isolated nodes} \\
\mathbb{E}[X] &= \mathbb{E}[\sum_{i=1}^{n} X_i] \\
&= \sum_{i=1}^{n} \mathbb{E}[X_i] && \text{(by linearity of expectation)} \\
&= n\mathbb{E}[X_i]
\end{aligned}$$

1. Say $p = (1+\epsilon)\frac{2ln(n)}{n^2}$

$$\begin{aligned}
\lim_{n\to\infty} \mathbb{E}[X] &= \lim_{n\to\infty} n(1-p)^{\binom{n-1}{2}} \\
&\leq \lim_{n\to\infty} ne^{-p\binom{n-1}{2}} \\
&\leq \lim_{n\to\infty} ne^{\frac{-2(1+\epsilon)ln(n)(n-1)(n-2)}{2n^2}} \\
&\leq \lim_{n\to\infty} nn^{\frac{-(1+\epsilon)(n-1)(n-2)}{n^2}} \\
&\leq \lim_{n\to\infty} nn^{-(1+\epsilon)} \to 0
\end{aligned}$$

As $\mathbb{E}[X]$ cannot be negative, we conclude that:

$$\lim_{n\to\infty} \mathbb{E}[X] \to 0$$

As the $X$ is guaranteed to be non-negative, we can apply the first moment method:

$$\Pr[X=0] = 1 - o(1)$$

2. Say $p = (1-\epsilon)\frac{2ln(n)}{n^2}$

Firstly, let's demonstrate a useful inequality.

$$\begin{aligned}
&n^4 \leq e^{n^2} \text{ which holds trivially for large } n \\
&\Rightarrow 4ln(n) \leq n^2 \\
&\Rightarrow \frac{2ln(n)}{n^2} \leq \frac{1}{2} \\
&\Rightarrow (1-\epsilon)\frac{2ln(n)}{n^2} \leq \frac{1}{2}
\end{aligned}$$

$$
\begin{aligned}
\lim_{n\to\infty} \mathbb{E}[X] &= \lim_{n\to\infty} n(1-p)^{\binom{n-1}{2}} \\
&\geq \lim_{n\to\infty} ne^{-p(1-p)\frac{(n-1)(n-2)}{2}} && \text{(via Fact 2, as } 0 \leq p \leq \frac{1}{2}\text{)} \\
&\geq \lim_{n\to\infty} ne^{-(1-\epsilon)\frac{2ln(n)}{n^2}(1-(1-\epsilon)\frac{2ln(n)}{n^2})\frac{(n-1)(n-2)}{2}} \\
&\geq \lim_{n\to\infty} nn^{-(1-\epsilon)(1-(1-\epsilon)\frac{2ln(n)}{n^2})}
\end{aligned}
$$

We observe that $\lim_{n\to\infty} \frac{2ln(n)}{n^2} \to 0$ and $\lim_{n\to\infty} 1-(1-\epsilon)\frac{2ln(n)}{n^2} \to 1$. Hence:

$$
\lim_{n\to\infty} \mathbb{E}[X] \geq \lim_{n\to\infty} nn^{-(1-\epsilon)} \to \infty
$$

In order to compute the variance, let's first have a look at the probavility of two distinct nodes $i$ and $j$ being isolated at the same time. All edges that would either involve i and j must not be added. To avoid double-counting, we shall not forget to take the edges containing both i and j into consideration.

$$
\begin{aligned}
\Pr[i \text{ and } j \text{ are isolated}] &= (1-p)^{\binom{n-1}{2}+\binom{n-1}{2}-\binom{n-2}{1}} \\
&= (1-p)^{(n-1)(n-2)-(n-2)} \\
&= (1-p)^{(n-2)(n-1-1)} \\
&= (1-p)^{(n-2)(n-2)} \qquad (1)
\end{aligned}
$$

Furthermore, we will introduce a hepful inequality.

$$
\begin{aligned}
&\forall n > 0 : \frac{n-1}{2} < n - \frac{2ln(n)}{n} \\
&\Rightarrow \frac{n-1}{2} < n-(1-\epsilon)\frac{2ln(n)}{n} \\
&\Rightarrow \frac{n-1}{2} < n(1-(1-\epsilon)\frac{2ln(n)}{n^2}) \\
&\Rightarrow \frac{n-1}{2} < n(1-p) \\
&\Rightarrow \frac{n(n-1)}{2} < n^2(1-p) \\
&\Rightarrow \binom{n}{2}(1-p)^{n-2} < n^2(1-p)^{n-1} \\
&\Rightarrow \binom{n}{2}(1-p)^{(n-2)(n-2)} < n^2(1-p)^{(n-1)(n-2)} \\
&\Rightarrow \binom{n}{2}(1-p)^{(n-2)(n-2)} < \mathbb{E}[X]^2 \qquad (2)
\end{aligned}
$$

Combining this knowledge allows us to bound the variance of $X$.

$$\begin{aligned}
\mathrm{Var}[X] &= \mathbb{E}[X^2] - (\mathbb{E}[X])^2 \\
&= \mathbb{E}[\sum_{i=1}^{n}\sum_{j=1}^{n} X_i X_j] - (\mathbb{E}[X])^2 \\
&= \sum_{i=1}^{n}\sum_{j=1}^{n} \mathbb{E}[X_i X_j] - (\mathbb{E}[X])^2 && \text{(by linearity of expectation)} \\
&= \sum_{i=1}^{n} \mathbb{E}[X_i] + \sum_{(i,j)\in\binom{V}{2}} \mathbb{E}[X_i X_j] - (\mathbb{E}[X])^2 \\
&= \mathbb{E}[X] + \sum_{(i,j)\in\binom{V}{2}} \Pr[i \text{ and } j \text{ are isolated}] - (\mathbb{E}[X])^2 \\
&= \mathbb{E}[X] + \binom{n}{2}(1-p)^{(n-2)(n-2)} - (\mathbb{E}[X])^2 && \text{(via 1)} \\
&< \mathbb{E}[X] + (\mathbb{E}[X])^2 - (\mathbb{E}[X])^2 && \text{(via 2)} \\
&< \mathbb{E}[X]
\end{aligned}$$

As we know that for relevant $p$, $\lim_{n\to\infty} \mathbb{E}[X] \to \infty$, it holds that $\mathbb{E}[X] \in o(\mathbb{E}[X]^2)$. Thus, $\mathrm{Var}[X] \in o(\mathbb{E}[X]^2)$. The second moment method implies that:

$$\Pr[X = 0] = o(1)$$

.

Summing up, we have:

$$\Pr[X=0] = \begin{cases} 1 - o(1) & p \geq (1+\epsilon)\frac{2ln(n)}{n^2} \\ o(1) & p \leq (1-\epsilon)\frac{2ln(n)}{n^2} \end{cases}$$

which corresponds to the definition of a sharp threshold.

# Exercise 3

1. We draw $n^{\frac{1}{4}}$ vertices u.a.r. from $V$. We observe that per edge, there are at most $n-1$ conflicting edges, i.e. edges with the same color. This follows directly from the fact that each node can have at most one edge of the same color. In total, there are at most $n\binom{n}{2}$ possible pairs of conflicting edges. The probability of drawing two conflicting edges is equal to the probability of choosing the four nodes of the two edges. The two edges cannot be conflicting by drawing less than 4 nodes, because that would

imply an improper edge-coloring.

$$\begin{aligned}
\mathbb{E}[\#\text{conflicting edge pairs}] &= \mathbb{E}[\sum_{i=1}^{n\binom{n}{2}} \mathbb{1}[\text{pair } i \text{ is drawn}]] \\
&\leq \sum_{i=1}^{n\binom{n}{2}} \mathbb{E}[\mathbb{1}[\text{pair } i \text{ is drawn}]] \text{ (by linearity of expectation)} \\
&\leq \sum_{i=1}^{n\binom{n}{2}} \Pr[\text{pair } i \text{ is drawn}] \\
&\leq \sum_{i=1}^{n\binom{n}{2}} \Pr[\text{the four nodes connecting the edges are drawn}] \\
&\leq \sum_{i=1}^{n\binom{n}{2}} (\frac{n^{\frac{1}{4}}}{n})^4 \text{ (independece of node-draws)} \\
&\leq n\binom{n}{2}\frac{1}{n^3} = \frac{n^2(n-1)}{2}\frac{1}{n^3} \\
&< 1
\end{aligned}$$

As the number of conflicting edge pairs is a non-negative integer, there needs to be a scenario in which it takes on the value 0. This concludes the proof.

2. We draw $log(n)$ nodes u.a.r. from $V$ into $A$ and $Cn/log(n)$ nodes u.a.r. into $F$. The number of possible conflicting pairs can now be bounded by the number of possible edges and the number of times a color can appear in the graph, i.e. $\binom{n}{2}\frac{n}{log(n)^3}$.It is essential to observe that a pair of potentially conflicting edges have to have at least two nodes inside of $A$ and they all

need to be not part of $F$.

$$
\begin{aligned}
&\mathbb{E}[\# \text{ conflicting pairs}]\\
&\leq \mathbb{E}[\sum_{i=1}^{\binom{n}{2}\frac{n}{log(n)^3}} \mathbb{1}[\text{pair } i \text{ is in H}]\\
&\leq \sum_{i=1}^{\binom{n}{2}\frac{n}{log(n)^3}} \Pr[\text{all nodes from } i \text{ are not in } F \text{and at least two in } A]\\
&\leq \binom{n}{2}\frac{n}{log(n)^3}(1-\frac{C\cdot n/log(n)}{n})^2(\frac{log(n)}{n})^4\\
&\leq \binom{n}{2}\frac{n}{log(n)^3}\cdot 1\cdot(\frac{log(n)}{n})^4\\
&\leq \frac{n^3}{log(n)^3}\frac{log(n)^4}{n^4}\\
&\leq \frac{log(n)}{n}\\
&< 1
\end{aligned}
$$

As the number of conflicting edge pairs is a non-negative integer, there needs to be a scenario in which it takes on the value 0. This concludes the proof.

# Exercise 4

1. Having $i$ elements already in her list per position, the probability of Alice's next guess being a zero-guess is $(\frac{n-i-1}{n-i})^n$. We now denote $X_i$ to be the number of guesses required to obtain the $i$th zero-guess. Observe that this expectation is the inverse of the the latter probability. $X$ will be refered to as the number of required guesses to arrive at the last, i.e. $n-1$th zero-guess.

$$
\begin{aligned}
X &= \sum_{i=1}^{n-1} X_i\\
\mathbb{E}[X] &= \mathbb{E}[\sum_{i=1}^{n-1} X_i]\\
&= \sum_{i=1}^{n-1}\mathbb{E}[X_i] && \text{(by linearity of expectation)}\\
&= \sum_{i=0}^{n-2}\frac{1}{(\frac{n-i-1}{n-i})^n}\\
&\geq \sum_{i=n-2}^{n-2}(\frac{n-i}{n-i-1})^n = (\frac{2}{1})^n
\end{aligned}
$$

Hence we have shown that $\mathbb{E}[X] \in \Omega(2^n)$.

2. The probability that a guess $g \in_{u.a.r.} \{1 \dots n\}^n$ is a radom guess:

$$\Pr[t(a,g) = 0] = (\frac{n-1}{n})^n$$

Thusly $C \cdot e \cdot nlog(n)$ guesses lead to $C \cdot e \cdot nlog(n)(\frac{n-1}{n})^n$ zero-guesses in expecation. We observe that:

$$\lim_{n\to\infty} (\frac{n-1}{n})^n = \lim_{m\to\infty} (\frac{1}{1+\frac{1}{m}})^m \to \frac{1}{e}$$

Therefore, for large enough $n$, we can expect $C \cdot e \cdot nlog(n) \cdot e = C \cdot nlog(n)$ many zero-guesses. Note that the Chernoff Bounds tell us that the probability of the number of zero-guesses deviating from the just established mean decreases exponentialy. Encountering for each position, each possible code $c \neq a_i$ as part of a zero-guess implies being able to determine Paul's codeword without any further guesses. In such a scenario we know for each po2sition the $n-1$ codes that cannot be part of the correct code. This leaves only one possible code per position, i.e. the sough-after code. Our aim is now to show that the former statement holds whp.

At this point, we want to point out that the guess for a position i is independent of the guesses for all other positions. At the same time, the zero guesses are independent of each other. Both those observations follow directly from the fact that the initial guesses are indepenently and u.a.r. drawn from $\{1 \dots n\}^n$.

We will calculate a convenient limit first.

$$\begin{aligned}
&\lim_{n\to\infty} (n-1)(\frac{n-2}{n-1})^{C\cdot nlog(n)} \\
&= \lim_{m\to\infty} (m+1)(\frac{m}{m+1})^{C(m+2)log(m+2)} \\
&= \lim_{m\to\infty} (m+1)((\frac{1}{1+\frac{1}{m}})^m(\frac{1}{1+\frac{1}{m}})^2)^{C\cdot log(m+2)} \\
&= \lim_{m\to\infty} (m+1)(\frac{1}{e}\cdot 1)^{C\cdot log(m+2)} \\
&= \lim_{m\to\infty} (m+1)(\frac{1}{m+2})^{C} \\
&\to 0 && \text{(for } C > 1)
\end{aligned}$$

$$\begin{aligned}
\Pr[X_p] &= \Pr[\text{for position } p \text{ each } c_i \neq a_p \text{ has been drawn among } k \text{ zero-guesses, drawn u.a.r.}] \\
&= 1 - \Pr[\text{for position } p \text{ some } c_i \neq a_p \text{ has not been drawn among } k \text{ zero-guesses, drawn u.a.r.}] \\
&\geq 1 - \sum_{c_i=1, c_i\neq a_p}^{n} \Pr[\text{for position } p, c_i \text{ has not been drawn among } k \text{ zero-guesses, drawn u.a.r.}] \\
&\text{(via union bound)} \\
&\geq 1 - (n-1)(\frac{n-2}{n-1})^k \\
&\to 1 \text{ (with large enough } n, k = C \cdot nlog(n), C > 1 \text{ and )}
\end{aligned}$$

$$
\begin{aligned}
&\Pr[\text{for each position } p \text{ each } c_i \neq a_p \text{ has been drawn among } k \text{ zero-guesses, drawn u.a.r.}] \\
&= \prod_{p=1}^{n} \Pr[X_p] \\
&= (1-(n-1)(\frac{n-2}{n-1})^k)^n \text{ (via independence of positions)} \\
&\to 1 \text{ (via the previous limit)}
\end{aligned}
$$

Summing up, we have established that we can expect $C \cdot log(n)$ many zero-guesses from $C \cdot e \cdot log(n)$ total guesses, which are sufficient to determine the codeword whp.

3. Our strategy is to run the algorithm $\mathcal{A}$ from 4.2 $C \cdot nlog(n)$ many times. We can easily check whether we have found the codeword that way. If not, which is very unlikely, we run a trivial deterministic algorithm. The deterministic algorithm iterates over all positions and tests all colors per position, all other colors fixed. This has runtime $O(n^2)$. Hence our total runtime is:

$$\Pr[\mathcal{A} \text{ is successful}] \cdot C \cdot nlog(n) + (1 - \Pr[\mathcal{A} \text{ is successful}]) \cdot C' \cdot n^2$$

$$
\begin{aligned}
&1 - \Pr[\mathcal{A} \text{ is successful}] \\
&= \Pr[\text{at least one } c \neq a_i \text{ has not been drawn among } C \cdot nlog(n) \text{ guesses}] \\
&\leq n(n-1)(\frac{n-2}{n-1})^{C \cdot nlog(n)} \text{ (by union bound over positions and different} c) \\
&\leq n(n-1)(1-\frac{1}{n-1})^{C \cdot nlog(n)} \\
&\leq n(n-1)(e^{-\frac{1}{n-1}})^{C \cdot nlog(n)} \text{ (by Fact 1)} \\
&\leq n(n-1)n^{-\frac{C \cdot n}{n-1}}
\end{aligned}
$$

Combining this with our runtime:

$$
\begin{aligned}
(1 - \Pr[\mathcal{A} \text{ is successful}]) \cdot C' \cdot n^2 &\leq C' \cdot n^3 (n-1) n^{-\frac{C \cdot n}{n-1}} \\
&\leq C' \cdot n^4 n^{-C}
\end{aligned}
$$

Clearly, if we choose $C$ to be large, i.e. $> 4$, this term quickly tends to zero. Hence the only term left in our runtime estimation is a constant factor of $nlog(n)$. Thereby the expected runtime of this algorithm is in $O(nlog(n))$.

4.