15-492: Parallel Algorithms **Lecturer:** Kanat Tangwongsan
Topic: Probability Bounds **Date:** September 27th
Scribe: Kanat Tangwongsan

10.1 Lecture Outline

This lecture focuses mainly on Chernoff bounds and their applications. In particular, we will discuss examples of showing high-probability bounds. The notes contain more proofs than we discuss in class.

- Basic bounds: Markov's inequality, Chebyshev's inequality
- Chernoff bounds
- Examples

10.2 Basic Bounds

First we will look at some basic bounds. These are fundamental building blocks for the Chernoff bounds. Recall the following Markov's inequality:

Theorem 10.2.1 (Markov's inequality) *For any random variable $X \geq 0$,*

$$\mathbf{Pr}[X > \lambda] < \frac{\mathbf{E}[X]}{\lambda}$$

The proof is standard. Note that we can substitute any positive function $f : X \to \mathbb{R}_+$ for X. Therefore,

$$\mathbf{Pr}[f(X) > f(\lambda)] < \frac{\mathbf{E}[f(X)]}{f(\lambda)}$$

Note also that when f is a non-decreasing function, we get that $\mathbf{Pr}[X > \lambda] = \mathbf{Pr}[f(X) > f(\lambda)]$. If we carefully pick $f(X)$, we can obtain better bounds. As an example, we pick $f(X) = X^2$ and obtain the Chebyshev's inequality:

$$\mathbf{Pr}[|X - \mathbf{E}[X]| \geq \lambda] = \mathbf{Pr}\left[(X - \mathbf{E}[X])^2 \geq \lambda^2\right] \leq \frac{\mathbf{E}\left[(X - \mathbf{E}[X])^2\right]}{\lambda^2} = \frac{\mathbf{Var}(X)}{\lambda^2}$$

Theorem 10.2.2 (Chebyshev's inequality) *For any random variable $X \in \mathbb{R}$,*

$$\mathbf{Pr}[|X - \mathbf{E}[X]| \geq \lambda] \leq \frac{\mathbf{Var}(X)}{\lambda^2}$$

10.3 Chernoff Bounds

We will now focus on Chernoff bounds. Chernoff bounds are typically tighter than Markov's inequality and Chebyshev's inequality, in part because they require stronger assumptions.

Let X be a sum of n *independent* random variables $\{X_i\}$, with $\mathbf{E}[X_i] = p_i$. For this lecture, we assume that $X_i \in \{0,1\}$ for all $i \leq n$. Similar bounds can be achieved when X_i's are arbitrary bounded random variables. Let μ denote the expected value of X, thus $\mu = \mathbf{E}[\sum X_i] = \sum \mathbf{E}[X_i] = \sum p_i$

The derivation here was not gone over in class. To establish the bound, we pick $f(X) = e^{tX}$ (for some $t < 0$) and apply Markov's inequality:

$$\mathbf{Pr}[X < (1-\delta)\mu] = \mathbf{Pr}\left[e^{tX} > e^{(1-\delta)t\mu}\right] \leq \frac{\mathbf{E}\left[e^{tX}\right]}{e^{(1-\delta)t\mu}}$$

It remains to specify the value of t and estimate $\mathbf{E}\left[e^{tX}\right]$. We estimate $\mathbf{E}\left[e^{tX}\right]$ first.

$$\begin{aligned}\mathbf{E}\left[e^{tX}\right] &= \mathbf{E}\left[e^{t\sum X_i}\right] = \mathbf{E}\left[\prod_i e^{tX_i}\right] = \prod_i \mathbf{E}\left[e^{tX_i}\right] = \prod_i (p_i e^t + (1-p_i)\cdot 1) \\ &= \prod_i (1 + p_i(e^t - 1))\end{aligned}$$

Since $1 + x \leq e^x$, we have that

$$\mathbf{E}\left[e^{tX}\right] \leq \prod_i \exp\left\{p_i(e^t-1)\right\} = \exp\left\{(e^t-1)\sum_i p_i\right\} = \exp\left\{(e^t-1)\mu\right\}$$

Putting them together, we obtain $\mathbf{Pr}[X < (1-\delta)\mu] \leq \exp\{(e^t-1)\mu - (1-\delta)t\mu\}$. This expression is minimized when $t = \ln(1-\delta)$. By considering the Taylor's expansion of $\ln(1-\delta)$, we know that, for $\delta \in (0,1)$, $(1-\delta)\ln(1-\delta) > -\delta + \delta^2/2$; therefore, we have proved the following theorem:

Theorem 10.3.1

1. *For* $0 < \delta < 1$,
$$\mathbf{Pr}[X < (1-\delta)\mu] \leq e^{-\mu\delta^2/2}.$$
2. *For* $\delta > 0$,
$$\mathbf{Pr}[X < (1-\delta)\mu] < \left(\frac{e^{-\delta}}{(1-\delta)^{1-\delta}}\right)^{\mu}.$$

We can derive the following bounds similarly (by being a little more careful):

Theorem 10.3.2 (Chernoff bounds)

1. *For any* $\delta > 0$,
$$\mathbf{Pr}[X \geq (1+\delta)\mu] < \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\mu}.$$

2. *For any* $0 < \delta \leq 1$,

$$\mathbf{Pr}[X \geq (1+\delta)\mu] \leq e^{-\mu\delta^2/3}.$$

3. *For* $R \geq 6\mu$,

$$\mathbf{Pr}[X \geq R] \leq 2^{-R}.$$

10.4 Coin Flipping

We now look at an application of Chernoff bounds. Suppose we have n unbiased coin. Let X_i be the indicator random variable such that $X_i = 1$ if the i^{th} flip is a heads and 0 otherwise. Let $X = \sum_i X_i$. Thus X counts the number of heads among the n flips and so $\mathbf{E}[X] = n/2$. We apply Chernoff bounds to study the deviation from the mean.

Consider that

$$\mathbf{Pr}\left[|X - n/2| \geq \frac{1}{2}\sqrt{12n\ln n}\right] \leq 2\exp\left\{-\frac{1}{3} \times \frac{n}{2} \times \frac{12\ln n}{n}\right\} = \frac{2}{n^2}$$

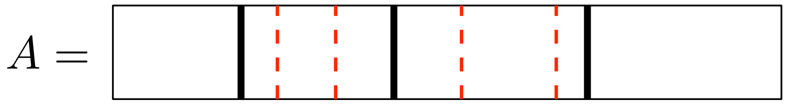
Therefore, the number of heads concentrate very tightly around the mean ($n/2$). It is a good exercise to compare the power of Chebyshev's bound with the Chernoff bounds. You should find that Chernoff gives a much stronger bound on the probability of deviation than Chebyshev. This is because Chebyshev only uses pairwise independence between the random variables, whereas Chernoff uses full independence. Full independence can sometimes imply exponentially better bounds.

10.5 Oversampling in Sample Sort (Revisited)

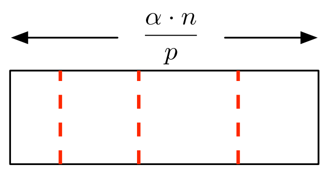
Recall the oversampling technique from previous lectures. We pick $s \cdot p$ candidate pivots uniformly at random, sort the candidates, and take every s element to be our real pivots. We then use these pivots to partition the data. Here's where Chernoff comes in.

Theorem 10.5.1 *Using the oversampling technique, all "bucket" has at most* $\alpha\frac{n}{p}$ *elements with probability at least* $1 - n \cdot \exp\left\{-(1-1/\alpha)^2\frac{s\alpha}{2}\right\}$. *Therefore, with* $s > 8\ln n$, *all bucket has at most* $2n/p$ *elements with probability at least* $1 - \frac{1}{n}$.

Proof: Suppose we want to partition an array A. In the figure below, we assume that A is sorted. The vertical bars are the candidate elements with the solid ones being the actual pivots. Following the algorithm we just described, we know that between any two consecutive solid bars there are exactly $s - 1$ candidates.



Now consider a chunk B of any $\frac{\alpha \cdot n}{p}$ consecutive elements from the sorted version of A. It is easy to see that B goes into a single bucket—and therefore there is a bucket with more than $\alpha n/p$ elements—if B contains less than s candidates.



Let X be the number of candidate elements in B. We will estimate $\mathbf{Pr}[X < s]$ using Chernoff. To do this, let X_i be the indicator random variable that the i-th element of B is a candidate, and so $X = \sum_i X_i$. By linearity of expectation, we know that

$$\mu = \mathbf{E}[X] = \sum_{i=1}^{\alpha \cdot n/p} \mathbf{E}[X_i] = \frac{\alpha \cdot n}{p} \times \frac{s \cdot p}{n} = \alpha \cdot s,$$

as each element has a probability of $s \cdot p/n$ of being a picked as a candidate. Applying Chernoff, we get

$$\mathbf{Pr}[X < s] = \mathbf{Pr}\left[X < \frac{1}{\alpha} \cdot \mu\right] \leq \exp\left\{-\left(1 - \frac{1}{\alpha}\right)^2 \frac{s\alpha}{2}\right\}.$$

By union bound, the probability that a bucket has more than $\alpha n/p$ elements is upper-bounded by $n \cdot \exp\left\{-(1 - 1/\alpha)^2 \frac{s\alpha}{2}\right\}$. ■

10.6 Randomized Lazy Sort

Instead of analyzing a standard version of randomized quick sort, we will consider the following variant of quick sort, called lazy sort:

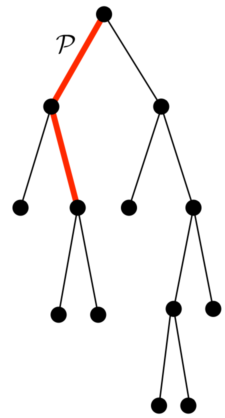
Rand-LSort(A) =

0. If $|A| \leq 1$ **return** A.
1. Pick $e \in A$ uniformly at random.
2. Split A into $A_<$ and $A_>$.
3. Go to Step 1 if $\max\{|A_<|, |A_>|\} > \frac{3}{4}|A|$.
4. **return** Rand-LSort$(A_<)$++$\{e\}$++Rand-LSort$(A_>)$

Note that in this sorting algorithm, when we make a recursive call, we are guaranteed that the problem size is reduced by at least $n/4$. Therefore, the recursion-tree depth is clearly bounded by $O(\log_{4/3} n)$.

Our first goal is to show that the running time is $O(n \log n)$ in expectation. From the observation above, we only have to show that actually we aren't going through steps 1 – 3 too many times. Note that there are $|A|/2$ elements that we can pick and satisfy the condition in step 3. So, in expectation, steps 1 – 3 are repeated only 2 times per recursive call (this is a geometric random variable with $p = 1/2$). Hence, the running time is expected $O(n \log n)$.

Our next goal is to show that the running time of the algorithm is $O(n \log n)$ with high probability. Consider the recursion tree for this algorithm.



Let x be an element from the array we want to sort. Let $\mathcal{P}$ be the path in the recursion tree that x "visits" (an element x "visits" a node if x shows up in the input of that node). From before we know for sure that $|\mathcal{P}| \leq \log_{4/3} n < 8 \ln n$.

Let T_x count how many times x participates in step 2 in total. Since each time that x participates in step 2 contributes 1 unit to the running time, summing up T_x over all x gives the running time. We will show that $\mathbf{Pr}[T_x > 32 \ln n]$ is small for all x. This will, in turn, imply that the algorithm runs in $32 n \ln n$ steps with "high probability."

To bound $\mathbf{Pr}[T_x > 32 \ln n]$, we will analyze the following game. A person starts at the root node, trying to walk down the path. At each step he flips a coin. With probability $\frac{1}{2}$ he advances to the next node down the path. Otherwise he stays put. Note that T_x corresponds exactly to the number of steps this person takes to reach the endpoint. Now we compute the probability that we take more than $32 \ln n$ steps to get to the end of $\mathcal{P}$. This is upper-bounded by the probability that flipping $32 \ln n$ unbiased coins and getting less than $8 \ln n$ heads (recall $|\mathcal{P}| < 8 \ln n$). Let X be the number of heads among the $32 \ln n$ flips. Applying Chernoff, we have

$$\mathbf{Pr}[T > 32 \ln n] \leq \mathbf{Pr}[X < 8 \ln n] = \mathbf{Pr}[X < (1 - 1/2) 16 \ln n] \leq \exp\left\{-\frac{1}{4} \times 8 \ln n\right\} = \frac{1}{n^2}.$$

By union bound, $\mathbf{Pr}[\exists x, T_x > 32 \ln n] \leq \sum_x \mathbf{Pr}[T_x > 32 \ln n] = \frac{1}{n^2} \times n = \frac{1}{n}$. Therefore, with probability *at least* $1 - 1/n$, $T_x \leq 32 \ln n$ for all x. It follows that the running time of the algorithm (i.e., $\sum_x T_x$) is upper-bounded by $32 n \ln n$ with probability at least $1 - 1/n$.