Lecture 2

1 Introduction

In last Tuesday's lecture, we proved that if n balls are thrown into n bins at random, then the maximum load is $O(\log n)$ with high probability. More precisely, we showed that the probability that all bins have less than $2\log_2 n$ balls at the end, is greater than or equal to $1-2/n^2$, for $n \geq 2^e$. In general, when we state that an event happens with high probability, we mean that the event happens with probability greater than or equal to $1 - \frac{1}{n^c}$ for any $n \geq n_0$, for some positive constants c and n_0.

In today's lecture, we examine the maximum load when a different protocol is used for throwing balls into bins. For each ball in sequence, we will choose two random bins, and assign the current ball to the bin that is least loaded at the moment (i.e. the one that currently has the least number of balls in it). We will show that when this protocol is used the maximum load is $O(\log\log n)$ with high probability.

2 Best-of-Two Analysis

In our analysis, we show that when $n/8$ balls are thrown into n bins, the maximum load is at most $O(\log\log n)$ with high probability. Without much extra work, one can show that when n balls are thrown into n bins, the max load will be at most $O(8\log\log n) = O(\log\log n)$ with high probability, but we omit further details. In general, for any positive c, it can be shown that the maximum load is $O(\log\log n)$ with high probability.

To analyze this problem, we first represent our balls and bins problem as a graph G. For each bin, we create one vertex, for a total of n vertices. For each ball, we create one edge between the two bins that it looks at when choosing which bin to be placed in, for a total of $n/8$ edges. By analyzing the graph corresponding to our balls and bins problem, we can prove the maximum load is at most $O(\log\log n)$ with high probability.

To start, imagine the following process:

```
While (G has nodes)
    Remove any node with degree less than or equal to 12
    Remove any edge connected to a removed node
Endwhile
```

Now for any node removed in iteration i of the while loop, we can prove the following statement by induction:

LEMMA 1
If a node v is removed in iteration i, then the final load of the bin corresponding to node v is at most $13i$.

Proof: The base case is easy as the first set of nodes removed has degree at most 12, and therefore must have load at most 12. To prove the inductive step, we assume that any node deleted in an iteration $\leq (i-1)$ has load at most $13(i-1)$, and prove that any new node deleted in iteration i has load at most $13i$. We know that any node v deleted in iteration i has at most 12 edges remaining, when looking at the graph at iteration i, but the node v may have had many more edges connected to nodes that have been deleted. Thus, we cannot simply bound the load of v by the degree of v in the starting graph G, since we do not know the degree of v in G. However, note that v is connected to at most 12 nodes with load $> 13(i-1)$, and all other edges connect to nodes with load $\leq 13(i-1)$. Now, if node v ever obtains strictly more than $13(i-1)$ balls, then no edges connecting v to deleted nodes, which have load at most $13(i-1)$, can contribute any load to v. Furthermore, the only edges that can contribute load to v after v has more than $13(i-1)+1$ balls are the 12 edges not yet deleted. Therefore, the load of v is at most $13(i-1)+1+12 = 13i$, and we have proved the inductive step. $\Box$

From here, if we can prove that the process above terminates in $O(\log\log n)$ iterations with high probability, then we will have proved that the max load is $O(\log\log n)$ with high probability. To prove this statement, we start by proving two lemmas:

Lemma 2
The largest connected component in G is size $O(\log n)$ with high probability.

Proof: To prove that the largest connected component of the graph G, representing our balls and bins experiment, is at most $O(\log n)$ with high probability, we analyze the probability of a connected component of size k occurring in G. In order for G to have a connected component of size k, we know that some set S of k nodes must have $k-1$ edges hit S (i.e. both endpoints of the edge lie in S). For a specific set S of k nodes and a specific set of $k-1$ edges, the probability that those $k-1$ edges all hit S is $\leq (\frac{k}{n})^{2(k-1)}$. Now, with a union bound, we can conclude that Prob[*Any* set of k nodes has $k-1$ edges hit it] $\leq$ (# subsets of nodes of size k)*(# subsets of edges of size $k-1$)*Prob[A specific set of k nodes is hit with a specific set of $k-1$ edges]. Now, using the fact that $\binom{n}{k} \leq (ne/k)^k$, we can show:

$$\begin{aligned}\text{Prob[Any set of k nodes has k-1 nodes hit it]} &\leq \binom{n}{k}\binom{n/8}{k-1}(\frac{k}{n})^{2(k-1)} \\ &\leq (\frac{ne}{k})^k(\frac{ne/8}{k-1})^{k-1}(\frac{k}{n})^{2(k-1)} \\ &\leq (\frac{n}{k})(\frac{e^2}{8})^{k-1}\end{aligned}$$

Finally, setting $k \geq 10\log_{\frac{8}{e^2}} n$, we have that Prob[$G$ has a connected component of size $\geq 10\log_{\frac{8}{e^2}} n$] $\leq \frac{1}{n^8}$, for n sufficiently large. Therefore, G's largest connected component is $O(\log n)$ with high probability. $\Box$

Lemma 3

For all subsets of $(10 \log_{\frac{8}{e^2}} n)$ nodes or fewer, the average degree in the induced subgraph is less than 6 with high probability.

Proof: To prove that small subsets of nodes have average degree less than 6 with high probability, we use analysis similar to the previous lemma. For a fixed subset of k nodes and fixed set of $3k$ edges, the probability that all $3k$ edges hit the subset, causing the subset to have average degree ≥ 6, is less than $((\frac{k}{n})^2)^{3k}$. Applying a union bound as before, we have Prob[*Any* set of k nodes has average degree ≥ 6] $\leq$ (# subsets of nodes of size k)*(# subsets of edges of size $3k$)*Prob[A specific set of k nodes is hit with a specific set of $3k$ edges]. Now, using the fact that $\binom{n}{k} \leq (ne/k)^k$, we can show:

$$\begin{aligned}
\text{Prob[Any set of k nodes has 3k edges hit it]} &\leq \binom{n}{k}\binom{n/8}{3k}(\frac{k}{n})^{6k} \\
&\leq (\frac{ne}{k})^k(\frac{ne/8}{3k})^{3k}(\frac{k}{n})^{6k} \\
&\leq (\frac{k}{n})^{2k}
\end{aligned}$$

Now, for $k \leq 10 \log_{\frac{8}{e^2}} n$, it can be shown that $(\frac{k}{n})^{2k} \leq \frac{1}{n^2}$, for sufficiently large n, and so we can conclude by union bound, the Prob[Any subset of $10 \log_{\frac{8}{e^2}}$ or fewer nodes has average degree ≥ 6] $\leq \frac{k}{n^2} \leq 1/n$, for sufficient large n. Thus, we have proved that the average degree in any subset of $(10 \log_{\frac{8}{e^2}} n)$ or fewer nodes is less than 6, with high probability. $\square$

With the two lemmas, we know that with high probability, every connected component in G is size $O(\log n)$ and every subgraph of a connected component has average degree less than 6. The last step is to show that in a graph where all connected components have size $O(\log n)$ and all subgraphs of connected components have average degree less than 6, the process described above terminates in at most $O(\log \log n)$ iterations.

Note that if we look at a particular connected component, at most half the nodes in the connected component have degree ≥ 12, otherwise the average degree will exceed 6. Furthermore, that means half the nodes in each connected component are deleted in each iteration of the process described above. If half the nodes in each connected component is deleted each iteration, then all nodes in all connected components will be deleted after $O(\log \log n)$ iterations. Therefore, the process terminates in $O(\log \log n)$ iterations, and combined with our first lemma, we have the final theorem:

Theorem 4

When the best-of-two *protocol is used to assign $n/8$ balls to n bins, the maximum load is $O(\log \log n)$ with high probability.*