Occupancy problems: Beating the $O(\log n)$ bound for balls and bins

Randomized Algorithms, Week 3, Lecture 2

Amitabha Bagchi

January 30, 2004

In the previous lecture we saw that given n balls and n bins, if we throw each ball into a bin uniformly at random, the maximum number of balls in any bin is $O(\frac{\log n}{\log \log n})$.

Today we will see that by placing the balls just a little more carefully we can get an exponentially better bound. Our placement strategy will be as follows:

- For each ball choose 2 bins uniformly at random from among the bins.
- Place the ball in the less full bin.
- If both bins are equally full, place the ball in any one of them arbitrarily.

For this strategy, which we will call *Best of Two*, we will show that the number of balls in the maximum bin is $O(\log \log n)$ which is exponentially smaller than the bound we showed for the simple strategy of throwing balls in bins uniformly.

Specifically the theorem we will show is:

Theorem 3.2.1. *If we throw $\frac{n}{512}$ balls into n bins using the* Best of Two *strategy the maximum number of balls in any bin is $O(\log \log n)$.*

Proof. Let us look upon the process of filling the bins as a "witness" graph G. The vertex set V of the graph is the set of all bins. The edge set E is determined by putting edges between two bins chosen a ball i.e. $E = \{e_i = (u, v) |$ ball i chooses bins u and $v\}$.

G is a random graph formed by *Best of Two.* In order to show the theorem we will first show that there is no large connected component in G.

Claim 3.2.2. *The size of G's largest connected component is $O(\log n)$ with probability at least $1 - \frac{1}{2n}$.*

Proof of Claim 3.2.2. We first determine the probability that a given set of $k + 1$ nodes form a connected component. We know that for n nodes to be connected they must have at least $n - 1$ edges between them (because every minimally connected graph on n nodes is a tree and a tree has exactly $n - 1$ edges.) Since having k edges is a necessary requirement for $k + 1$ nodes to be connected we can conclude that:

$$\Pr[\text{A given set of } k+1 \text{ nodes is connected}] \le \Pr[\text{at least } k \text{ edges fall within these } k+1 \text{ nodes}]$$

Satisfy yourself that:

$$\Pr[\exists \text{ a set of } k+1 \text{ connected nodes}] \leq \binom{\frac{n}{512}}{k} \cdot \left(\left(\frac{8k}{n}\right)^2\right)^k$$

Therefore we have, using Inclusion-Exclusion that:

$$\Pr[\exists \text{ a set of } k+1 \text{ connected nodes}] \leq \binom{n}{k+1} \cdot \binom{\frac{n}{512}}{k} \cdot \left(\frac{8k}{n}\right)^{2k}$$

We use two inequalities to simplify this expression:

$$\binom{n}{k+1} \leq n \cdot \binom{n}{k}$$

and

$$\binom{n}{k} \leq \left(\frac{en}{k}\right)^k$$

Hence we have:

$$\begin{aligned}
\Pr[\exists \text{ a set of } k+1 \text{ connected nodes}] &\leq n \cdot \left(\frac{en}{k}\right)^k \cdot \left(\frac{en}{512k}\right)^k \cdot \left(\frac{8k}{n}\right)^{2k} \\
&\leq n \cdot \left(\frac{e^2}{8}\right)^k \\
&\leq \frac{1}{2n} \quad \left(\text{by choosing } k = \frac{\log \sqrt{2}n}{\log \frac{8}{e^2}}\right)
\end{aligned}$$

□

Further we will show that the average degree of this large component is a constant with high probability.

Claim 3.2.3. *There is a constant c such that the average degree of any subgraph of size at least c is at most 5 with probability at least $1 - \frac{1}{2n}$.*

Proof of Claim 3.2.3. First we observe that:

$$\Pr[\text{A given set of } k \text{ nodes has at least } \frac{5k}{2} \text{ edges}] \leq \binom{\frac{n}{512}}{\frac{5k}{2}} \cdot \left(\left(\frac{8k}{n}\right)^2\right)^{5k/2}$$

Hence, by Inclusion Exclusion we get:

$$\begin{aligned}
\Pr[\exists \text{ a set of } k \text{ nodes with more than } \frac{5k}{2} \text{ edges}] &\leq \binom{n}{k} \cdot \binom{\frac{n}{512}}{\frac{5k}{2}} \cdot \left(\left(\frac{8k}{n}\right)^2\right)^{5k/2} \\
&\leq \left(\frac{en}{k}\right)^k \cdot \left(\frac{en}{256 \cdot 5k}\right)^{5k/2} \cdot \left(\frac{8k}{n}\right)^{2(5k/2)} \\
&\leq \left(\frac{8e^{7/2}}{20^{5/2}}\right)^k \cdot \left(\frac{8k}{n}\right)^{3k/2}
\end{aligned}$$

For values of k greater than $k_u = \dfrac{\log \sqrt{2}n}{\log \frac{20^{5/2}}{8e^{7/2}}}$ the first of these two terms is bounded by $1 - \frac{2}{n}$. For values lower than that we consider the second term. Differentiating it with respect to k we find that there is a constant c_1 such that $\left(\frac{8k}{n}\right)^{3k/2}$ is a decreasing function between c_1 and k_u. Similarly it is easy to see that there is a constant c_2 such that $\left(\frac{8c_2}{n}\right)^{3c_2/2} \leq \frac{2}{n}$. We set the constant c to be the max of the two values. $\square$

We now define a process of edge removals in G.

- While possible remove all vertices of degree at most 10.

This removal process proceeds in rounds. At each round we remove all the vertices whose current degree is 10. Claim 3.2.2 and Claim 3.2.3 then help us prove that the number of rounds of removal before we get to constant sized components is bounded.

Claim 3.2.4. *The removal process terminates in $O(\log\log n)$ steps leaving components of constant size.*

Proof of Claim 3.2.4. Since the average degree of a component with $l > c$ vertices is 5 where c is the constant from Claim 3.2.3, there have to be at least $l/2$ vertices with degree at most 10. Remove these in the first round and look at the remaning graph. While its size is greater than c we know that its average degree is 5 so we can make the same argument to halve its size. This continues till the components are smaller than c. Hence we require $\log(\frac{l}{c})$ steps before the removal process terminates leaving components of size at most c. From Claim 3.2.2 we know that l is $O(\log n)$. This completes the proof. $\square$

Before we relate G to the number of balls in a bin we define the *height* h_i of a ball i with respect to a set of balls S which have made bin choices. Ball i chooses two bins. Now when all the *Best of Two* decisions have been made for the balls in S except i the minimum of the numbers of balls in the two bins chosen by i is defined to be the height of i, denoted h_i.

Claim 3.2.5. *Assuming the removal process ends in a graph with at most c vertices, if edge e_i is removed in round t, the maximum height h_i of ball i is $h_i \leq 10t + c$.*

Proof of Claim 3.2.4. When the removal process ends we have all the vertices left having degree at most c and hence the height of a ball in any of the bins left can be at most c. Consider a bin j whose vertex v_j gets removed from the graph in round t. At any round $t' < t$ it had degree strictly greater than 10. The edges removed from v_j in round t' were to vertices with degree at most 10. Hence the height of any ball in v_j from among the balls whose edges were removed in round t' can be at most 10.

In this manner we see that each round that v_j lives through contributes at most 10 balls to v_j and resolving the left over component adds another c balls. $\square$

Putting together Claim 3.2.4 and Claim 3.2.5 we get the proof of Theorem 3.2.1. $\square$

Notes

The proof in this lecture is a version of the proof in Satish Rao's lecture notes [4]. That proof in turn is a simplified and insightful exposition of the "Witness Tree" method pioneered in the work of Karp, Luby and Meyer auf der Heide [2]. Mitzenmacher, Richa and Sitaraman [3] give an excellent survey of the various proof techniques used for best of two choice method and an update on the state of research in this area. Particulary interesting is Vöcking's paper on how an assymetric tie breaking rule can help improve the maximum height bound [5]. Czumaj, Riley and Scheideler [1] provide a more recent update on the state of research in this area and some interesting results for m balls and n bins.

References

[1] A. Czumaj, C. Riley, and C. Scheideler. Perfectly balanced allocation. In *Proc. of RANDOM-APPROX 2003*, number 2764 in Lecture Notes in Computer Science, pages 240–251. Springer, 2003.

[2] R. M. Karp, M. Luby, and F. Meyer auf der Heide. Efficient PRAM simulation on a distributed memory machine. *Algorithmica*, 16:245–281, 1996.

[3] M. Mitzenmacher, A. Richa, and R. Sitaraman. *Handbook of Randomized Computing*, volume I, chapter 9: The power of two random choices: A survey of techniques and results. Kluwer Academic Publishers, 2001. Available online at http://citeseer.nj.nec.com/mitzenmacher00power.html.

[4] S. Rao. Lecture 14. http://citeseer.nj.nec.com/492067.html, October 2001. Graduate Algorithms CS 270. University of California, Berkeley.

[5] B. Vöcking. How asymmetry helps load balancing. In *Proc. 40th Annual Symp. on Foundation of Computer Science (FOCS '99)*, pages 131–141, 1999.