Randomized Algorithms

Maximum loads and Poissonization

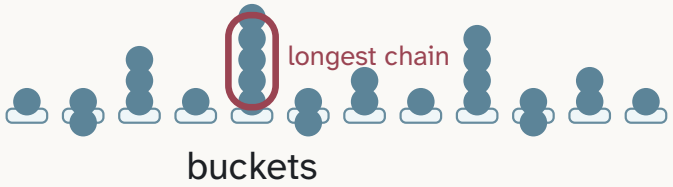
Alex Gittens
Department of Computer Science, Rensselaer Polytechnic Institute

Fall 2026

Separate chaining

We start with the standard hashing scheme with chaining.

1. The table has n buckets.
2. A key x is sent to bucket $h(x)$.
3. Bucket i stores a chain of all keys with $h(x) = i$.
4. A lookup searches one chain.



In the random model, the stored keys choose buckets independently and uniformly.

Performance question

If a lookup may have to scan an entire chain, how long can the longest chain be?

The occupancy model

Hashing with chaining is the same as throwing balls into bins:

$$\text{key} \longleftrightarrow \text{ball}, \qquad \text{bucket} \longleftrightarrow \text{bin}.$$

We store

$$m = \alpha n$$

keys in n buckets, where $\alpha > 0$ is a fixed constant. The load factor is

$$\alpha = \frac{m}{n}.$$

Let

$$L_i = \text{load of bucket } i, \qquad L_{\max} = \max_{1 \le i \le n} L_i.$$

The average load is fixed:

$$\frac{1}{n} \sum_{i=1}^{n} L_i = \frac{m}{n} = \alpha.$$

Average load is not enough

The average chain length is α, but the worst lookup is governed by

$$L_{\max} = \max_i L_i.$$

average load $= \alpha$ does not imply small worst load

Question

As n grows and α stays fixed, what is the typical size of $L_{\max}$?

This is the issue: randomized algorithms can have good local average behavior while their performance is set by the worst local outcome.

A quick experiment

For the special case $\alpha = 1$ and $n = 1000$, simulation gives the following distribution for $L_{\max}$.



Simulated from 5000 independent experiments; $\log n / \log \log n \approx 3.6$ for $n = 1000$.

PART 1

Find the threshold

1

Start with one bucket

For a fixed bucket i,

$$L_i \sim \text{Binomial}\left(m, \frac{1}{n}\right), \qquad m = \alpha n, \qquad \mathbb{E}L_i = \alpha.$$

Suppose $L_i \geq k$. Then there exists a set of k keys that all hash to bucket i:

$$\{L_i \geq k\} \subseteq \bigcup_{\substack{S \subseteq [m] \\ |S|=k}} \{\text{all keys in } S \text{ hash to bucket } i\}.$$

For a fixed set S of k keys,

$$\Pr(\text{all keys in } S \text{ hash to bucket } i) = n^{-k}.$$

If more than k keys land in bucket i, the same outcome appears in many events in the union. This overcounting is fine because we only need an upper bound.

A one-bucket tail bound

From the containment and the union bound,

$$\begin{aligned}\Pr(L_i \geq k) &\leq \binom{m}{k} n^{-k} \\ &\leq \left(\frac{em}{k}\right)^k n^{-k} \\ &= \left(\frac{e\alpha}{k}\right)^k.\end{aligned}$$

Equivalently,

$$\Pr(L_i \geq k) \leq \exp\{-k \log k + O(k)\},$$

where the $O(k)$ term depends on the fixed constant α.

Main term

The dominant term is $-k \log k$. The constant load factor only changes the lower-order $O(k)$ part.

From one bucket to all buckets

Let

$$Z_k = \#\{i : L_i \geq k\} = \sum_{i=1}^{n} \mathbf{1}\{L_i \geq k\}.$$

Then

$$\mathbb{E}Z_k = n \Pr(L_1 \geq k).$$

Linearity of expectation is enough here; no independence between bucket loads is needed.
Using the one-bucket bound,

$$\mathbb{E}Z_k \leq n \left(\frac{e\alpha}{k}\right)^k.$$

Taking logs gives

$$\boxed{\log \mathbb{E}Z_k \leq \log n - k \log k + O(k).}$$

This expression is the source of the threshold.

The threshold calculation

The maximum should be near the value of k where the expected number of buckets with load at least k is about one:

$$\mathbb{E}Z_k \approx 1.$$

Using the bound above, this suggests

$$n\left(\frac{e\alpha}{k}\right)^k \approx 1.$$

Taking logs,

$$\log n + k\log(e\alpha) - k\log k \approx 0,$$

so

$$k(\log k - \log(e\alpha)) \approx \log n.$$

For large k, the constant $\log(e\alpha)$ is lower order than $\log k$, giving the dominant balance

$$\boxed{k\log k \approx \log n.}$$

The predicted scale

Try

$$k \approx \frac{\log n}{\log \log n}.$$

Then

$$\log k \approx \log \log n,$$

and hence

$$k \log k \approx \frac{\log n}{\log \log n} \log \log n = \log n.$$

So the threshold calculation predicts

$$\boxed{L_{\max} \text{ should be of order } \frac{\log n}{\log \log n}.}$$

The theorem has two halves

Fix $\varepsilon > 0$ and define

$$k_+ = (1+\varepsilon)\frac{\log n}{\log\log n}, \qquad k_- = (1-\varepsilon)\frac{\log n}{\log\log n}.$$

We will prove

$$\Pr(L_{\max} \geq k_+) \to 0$$

and

$$\Pr(L_{\max} < k_-) \to 0.$$

Equivalently,

$$\Pr(k_- \leq L_{\max} \leq k_+) \to 1.$$

The upper half rules out buckets above the threshold. The lower half shows that a bucket below the threshold really does appear.

PART 2

Upper bound

2

Set the upper threshold

Recall

$$\mathbb{E}Z_k \le n\left(\frac{e\alpha}{k}\right)^k.$$

Set

$$k = k_+ = (1+\varepsilon)\frac{\log n}{\log\log n}.$$

Then

$$\log \mathbb{E}Z_{k_+} \le \log n + k_+ \log(e\alpha) - k_+ \log k_+.$$

Since α is fixed,

$$k_+ \log(e\alpha) = O(k_+),$$

so

$$\boxed{\log \mathbb{E}Z_{k_+} \le \log n - k_+ \log k_+ + O(k_+).}$$

Evaluate the main term

For

$$k_+ = (1+\varepsilon)\frac{\log n}{\log\log n},$$

we have

$$\log k_+ = \log\log n - \log\log\log n + O(1).$$

Therefore

$$\begin{aligned} k_+ \log k_+ &= (1+\varepsilon)\frac{\log n}{\log\log n}\left(\log\log n - \log\log\log n + O(1)\right) \\ &= (1+\varepsilon)\log n + o(\log n). \end{aligned}$$

Also,

$$k_+ = O\left(\frac{\log n}{\log\log n}\right) = o(\log n).$$

Finish the upper bound

Substituting into the expectation bound,

$$\begin{aligned}\log \mathbb{E}Z_{k_+} &\le \log n - k_+ \log k_+ + O(k_+) \\ &\le \log n - (1+\varepsilon)\log n + o(\log n) \\ &= -\varepsilon \log n + o(\log n) \to -\infty.\end{aligned}$$

Thus

$$\mathbb{E}Z_{k_+} \to 0.$$

Now Markov's inequality gives

$$\Pr(L_{\max} \ge k_+) = \Pr(Z_{k_+} \ge 1) \le \mathbb{E}Z_{k_+} \to 0.$$

So

$$\boxed{\Pr(L_{\max} \le k_+) \to 1.}$$

The upper bound is only half the story

We have shown

$$\Pr\left(L_{\max} \leq (1+\varepsilon)\frac{\log n}{\log\log n} \right) \to 1.$$

This does not yet show that the threshold is correct. The union bound might be loose.

What remains

We still need to prove that a bucket reaches

$$k_- = (1-\varepsilon)\frac{\log n}{\log\log n}$$

with high probability.

PART 3

Lower bound

3

The product argument we would like to use

Let

$$k_- = (1-\varepsilon)\frac{\log n}{\log\log n}, \qquad Z_{k_-} = \#\{i : L_i \geq k_-\},$$

and

$$p_{k_-} = \Pr(L_i \geq k_-).$$

If the events $\{L_i \geq k_-\}$ were independent across buckets, then

$$\Pr(Z_{k_-} = 0) = (1-p_{k_-})^n \leq e^{-np_{k_-}}.$$

So $np_{k_-} \to \infty$ would imply

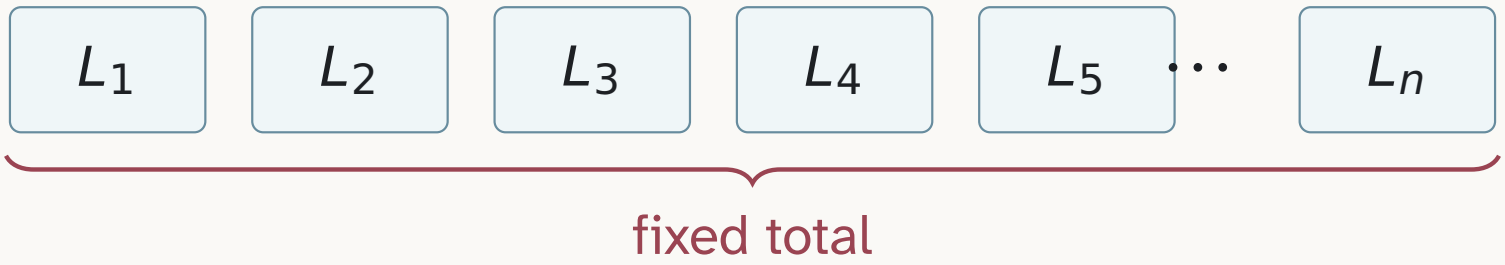
$$\Pr(Z_{k_-} = 0) \to 0.$$

That would prove $L_{\max} \geq k_-$ with high probability.

The bucket loads are dependent

The ball placements are independent. The bucket loads are not.

$$\boxed{L_1 + \cdots + L_n = m = \alpha n.}$$



If one bucket receives extra balls, those balls are unavailable to the other buckets. Therefore the equality

$$\Pr(Z_{k_} = 0) = (1 - p_{k_})^n$$

is not valid in the fixed-total model.

PART 4

Poissonization

4

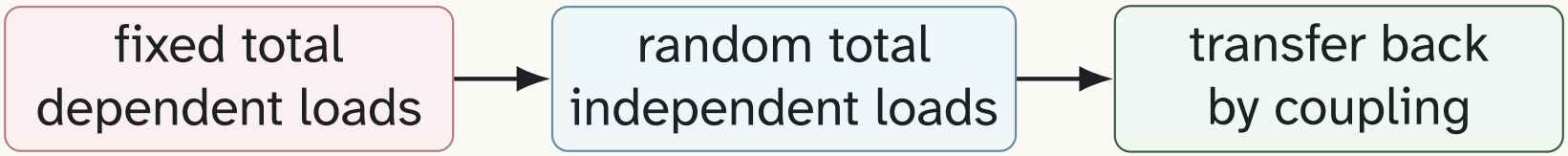
Replace the fixed total by a random total

The lower-bound calculation would be easy if the bucket loads were independent. The fixed-sum constraint is what prevents that.

The idea is to change the experiment in a controlled way:

$$\text{fixed total} \quad \longrightarrow \quad \text{random total.}$$

With the right random total, the individual bucket loads become independent.



This technique is called *Poissonization*.

Poissonizing the hash table

First keep the same average total number of keys:

$$N \sim \text{Poisson}(\alpha n).$$

Throw these N balls independently and uniformly into the n bins, and write the resulting loads as

$$\widetilde{L}_1, \ldots, \widetilde{L}_n.$$

The total now equals αn only on average:

$$\mathbb{E}N = \alpha n.$$

A fixed bin therefore has mean

$$\mathbb{E}\widetilde{L}_i = (\alpha n) \cdot \frac{1}{n} = \alpha.$$

The crucial fact is stronger than this mean calculation:

$$\boxed{\widetilde{L}_1, \ldots, \widetilde{L}_n \text{ are independent } \text{Poisson}(\alpha).}$$

The joint probability factors

To prove independence, we show that the joint probability factors into one-bin probabilities.
Fix nonnegative integers $\ell_1, \ldots, \ell_n$ and put

$$r = \ell_1 + \cdots + \ell_n.$$

For these loads to occur, the random total must be $N = r$. Given $N = r$, the loads are multinomial:

$$\Pr(\widetilde{L}_1 = \ell_1, \ldots, \widetilde{L}_n = \ell_n \mid N = r) = \frac{r!}{\ell_1! \cdots \ell_n!} \left(\frac{1}{n}\right)^r.$$

Also,

$$\Pr(N = r) = e^{-\alpha n} \frac{(\alpha n)^r}{r!}.$$

Poisson splitting

Multiplying the two factors gives

$$\begin{aligned}
&\Pr(\widetilde{L}_1 = \ell_1, \ldots, \widetilde{L}_n = \ell_n) \\
&= e^{-\alpha n} \frac{(\alpha n)^r}{r!} \frac{r!}{\ell_1! \cdots \ell_n!} \left(\frac{1}{n}\right)^r \\
&= e^{-\alpha n} \frac{\alpha^{\ell_1 + \cdots + \ell_n}}{\ell_1! \cdots \ell_n!} \\
&= \prod_{i=1}^{n} e^{-\alpha} \frac{\alpha^{\ell_i}}{\ell_i!}.
\end{aligned}$$

This is exactly a product of n independent $\mathrm{Poisson}(\alpha)$ probabilities. Hence

$$\widetilde{L}_1, \ldots, \widetilde{L}_n \overset{\text{ind}}{\sim} \mathrm{Poisson}(\alpha).$$

Lower the mean before coupling

The equal-mean Poisson model reveals the independence structure. For the coupling back to exactly $m = \alpha n$ balls, we use a slightly smaller mean.
Choose a fixed constant

$$0 < \beta < \alpha,$$

and let

$$N \sim \text{Poisson}(\beta n).$$

Then

$$\widetilde{L}_1, \ldots, \widetilde{L}_n \overset{\text{ind}}{\sim} \text{Poisson}(\beta).$$

With mean αn, the event $N \leq m$ is not overwhelmingly likely. With mean βn, the Poissonized experiment uses fewer than m balls with high probability.

The lower-bound goal in the Poisson model

Let

$$k_- = (1-\varepsilon)\frac{\log n}{\log\log n}, \qquad q_- := \Pr(\mathrm{Poisson}(\beta) \geq k_-).$$

Here q_- is the probability that one Poissonized bucket reaches the lower threshold.
Since the Poissonized bucket loads are independent,

$$\Pr(\widetilde{L}_{\max} < k_-) = (1-q_-)^n \leq e^{-nq_-}.$$

Therefore the Poissonized lower bound will follow once we show

$$\boxed{nq_- \to \infty.}$$

Lowering the mean from α to β helps with coupling. Now we check that it did not make heavy buckets too rare.

A single Poisson tail term is enough

A bucket can reach k_- balls by having exactly k_- balls, so

$$q_- = \Pr(\text{Poisson}(\beta) \geq k_-) \geq \Pr(\text{Poisson}(\beta) = k_-).$$

For a $\text{Poisson}(\beta)$ random variable,

$$\Pr(\text{Poisson}(\beta) = k_-) = e^{-\beta} \frac{\beta^{k_-}}{k_-!}.$$

Using $k_-! \leq k_-^{k_-}$ gives

$$q_- \geq e^{-\beta} \left(\frac{\beta}{k_-} \right)^{k_-}.$$

nq_- goes to infinity

From the previous slide,

$$\log(nq_-) \geq \log n - \beta + k_- \log \beta - k_- \log k_-.$$

For

$$k_- = (1-\varepsilon)\frac{\log n}{\log\log n},$$

we already computed

$$k_- \log k_- = (1-\varepsilon)\log n + o(\log n).$$

Since $\beta > 0$ is fixed,

$$k_- \log \beta = O(k_-) = o(\log n).$$

Thus

$$\log(nq_-) \geq \varepsilon \log n + o(\log n) o\infty.$$

So $nq_- \to \infty$.

A heavy bucket appears in the Poisson model

We have shown

$$nq_{-} \to \infty.$$

The independence calculation gave

$$\Pr(\widetilde{L}_{\max} < k_{-}) \leq e^{-nq_{-}}.$$

Therefore

$$\Pr(\widetilde{L}_{\max} < k_{-}) o 0, \qquad \text{or equivalently} \qquad \Pr(\widetilde{L}_{\max} \geq k_{-}) o 1.$$

Replacing α by any fixed $\beta > 0$ did not change the leading threshold: the dominant term is still $-k \log k$, while the mean contributes only an $O(k)$ term.

PART 5

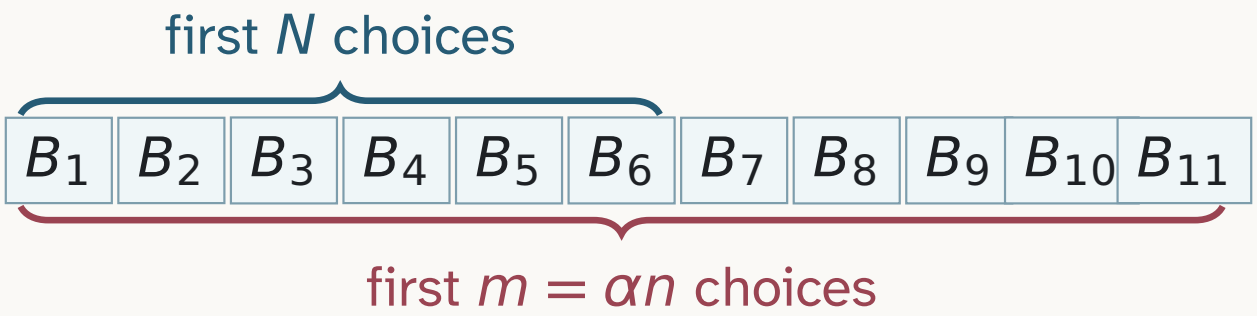
Couple back

5

Use the same bucket choices

The Poissonized model has a heavy bucket with high probability. Now connect it to the original hash table with exactly $m = \alpha n$ keys.
Generate one infinite sequence of independent uniform bucket choices:

$$B_1, B_2, B_3, \ldots.$$



Use the first $N \sim \text{Poisson}(\beta n)$ choices for the Poissonized experiment and the first m choices for the original experiment.

Adding balls cannot lower the maximum

On the event $N \le m$,

$$(B_1, \ldots, B_m) = (B_1, \ldots, B_N) + \text{additional placements.}$$

Adding balls cannot decrease any existing bucket load. Hence

$$\widetilde{L}_{\max} \le L_{\max} \qquad \text{whenever } N \le m.$$

Equivalently,

$$\{L_{\max} < k_-\} \subseteq \{\widetilde{L}_{\max} < k_-\} \cup \{N > m\}.$$

Therefore

$$\boxed{\Pr(L_{\max} < k_-) \le \Pr(\widetilde{L}_{\max} < k_-) + \Pr(N > m).}$$

The Poissonized process usually fits inside

Let

$$N \sim \text{Poisson}(\beta n), \qquad \mu = \mathbb{E}N = \beta n, \qquad m = \alpha n.$$

Since $\beta < \alpha$,

$$m = \alpha n = \frac{\alpha}{\beta}\mu = (1+\delta)\mu, \qquad \delta = \frac{\alpha}{\beta} - 1 > 0.$$

The Chernoff bound from the previous lecture gives

$$\Pr(N \geq (1+\delta)\mu) \leq \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}} \right)^{\mu}.$$

Substituting $\mu = \beta n$ and $1 + \delta = \alpha/\beta$,

$$\Pr(N \geq m) \leq \exp\left[-n \left(\alpha \log \frac{\alpha}{\beta} - \alpha + \beta \right) \right].$$

Finish the lower bound

The coefficient

$$\alpha \log \frac{\alpha}{\beta} - \alpha + \beta$$

is positive when $\beta < \alpha$, because $x \log x - x + 1 > 0$ for $x > 1$. Thus

$$\Pr(N > m) \to 0.$$

We also proved

$$\Pr(\widetilde{L}_{\max} < k_-) \to 0.$$

Using the coupling inequality,

$$\Pr(L_{\max} < k_-) \leq \Pr(\widetilde{L}_{\max} < k_-) + \Pr(N > m) \to 0.$$

Therefore

$$\boxed{\Pr(L_{\max} \geq k_-) \to 1.}$$

This is the lower half of the theorem.

PART 6

Conclusion

6

Maximum load at constant average load

For every fixed $\alpha > 0$ and fixed $\varepsilon > 0$,

$$\Pr\left((1-\varepsilon)\frac{\log n}{\log\log n} \le L_{\max} \le (1+\varepsilon)\frac{\log n}{\log\log n} \right) \to 1,$$

where $m = \alpha n$ balls are thrown independently into n bins.
Equivalently,

$$\boxed{L_{\max} = (1+o(1))\frac{\log n}{\log\log n} \quad \text{w.h.p.}}$$

average load $= \alpha$ but largest load $\asymp \dfrac{\log n}{\log\log n}$

What the proof taught us

The hashing theorem mattered, but so did the proof method.

Upper bound

One bucket first:

$$\Pr(L_i \geq k).$$

Then all buckets:

$$\mathbb{E}Z_k = n \Pr(L_i \geq k).$$

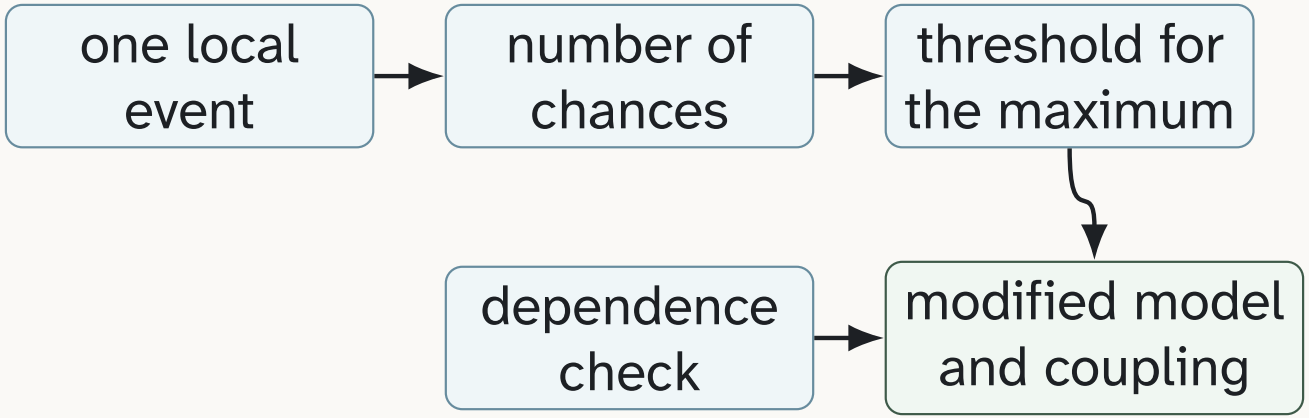
Above the threshold, Markov rules out heavy buckets.

Lower bound

Expectation alone was not enough. The bucket loads were dependent.
Poissonization gave an independent model, and coupling transferred the result back.

The reusable pattern

Now we can abstract the proof.



In this proof, the threshold came from

$$\mathbb{E}Z_k = n \Pr(L_i \geq k) \approx 1.$$

The upper bound followed directly. The lower bound required an additional move because the bucket loads were dependent.

Other places this issue appears

The same kind of question arises whenever many local random outcomes determine the worst case.

Examples

- maximum server load;
- largest collision class;
- randomized routing congestion;
- maximum local overload;
- any atypical coordinate, sample, or vertex.

Method

Analyze one location first. Count how many chances the event has to occur. Then check whether dependence changes the conclusion.

Next: two choices

Today we analyzed ordinary hashing with independent uniform bucket choices. The proof showed how to handle the dependence among bucket loads analytically.

Next time, the algorithm itself uses the current loads.

Two-choice placement

For each arriving key, sample two buckets i and j independently and uniformly. Place the key in the less loaded of the two buckets.

This changes the maximum load dramatically:

$$\text{one choice: } \Theta\left(\frac{\log n}{\log\log n}\right), \qquad \text{two choices: } \frac{\log\log n}{\log 2} + O(1).$$

The samples are independent, but the final placement is load-dependent.

Questions?