COMS 4995-3: Advanced Algorithms — Jan 25, 2017

Lecture 3 – Hashing: Power of 2 Choices

Instructor: *Alex Andoni* — Scribes: *Akshay Khatri, Kartikeya Upasani*

1 Hashing Continued

The hashing problem was introduced last time. In short, the problem is to design
$h : [U] \to [n]$
that solves the dictionary problem:

- given set S of size m
- query: given $x \in [U]$, output if $x \in S$

Parameters:

- space = ? ($O(|S| \log U)$ for a fully random hash function)
- time = ? ($O(1)$ for a fully random hash function)

h : fully random $\to$ takes too much space
h : Universal Hash Function $\to$ Cheaper and atleast as good as fully random (mostly)

Runtime: $|\left[h^{-1}\left[h(x)\right]\right] \cap S| \triangleq L_x$

C = #collisions = #pairs x,y $\in$ S that fall in the same bucket i.e. $h(x) = h(y)$

Claim 1. $\mathbb{E}[C] = \left[\frac{{}^mC_2}{n}\right] = \frac{m(m-1)}{2n}$

Proof. Proved last time

Want # collisions = 0?

Fix n = $\frac{4m(m-1)}{2} = O(m^2)$
$\Rightarrow \mathbb{E}[C] \leq \frac{1}{4}$

By Markov:

$Pr[C > 3\,\mathbb{E}[\mathbb{C}]] \leq \frac{1}{3}$

$\Rightarrow Pr\left[C > \frac{3}{4}\right] \leq \frac{1}{3}$

With probability $\geq \frac{2}{3}$, we have $C \leq \frac{3}{4}$

$\Rightarrow C \, = \, 0$

Conclusion:

fix $n = \frac{4m(m-1)}{2} = O(m^2)$

Then no collisions with probability $\frac{3}{4}$

To provide a different probability value,

$n \, = \, \frac{11m(m-1)}{2}$

$\Rightarrow \mathbb{E}[C] \leq \frac{1}{11}$

$P\left[C > 10\,\mathbb{E}[C]\,\right] \leq \frac{1}{10}$

$\Rightarrow$ With prob $\frac{9}{10}$, $C \leq \frac{10}{11}$, implies $C = 0$

If we set $n = O(m^2)$, suffices for no collisions
Space $= O(n) = O(m^2) = O(|S|^2)$

Better Space = ?

Fix n = m

Claim 2. *For fixed x, $\mathbb{E}\left[L_x\right] \leq \left[1 + \frac{m-1}{n}\right]$ where L_x is the size of the bucket containing x if $n = m$, $\Rightarrow \mathbb{E}[L_x] \leq O(1)$*

Proof.

$$\begin{aligned}
\mathbb{E}[L_x] &= 1 + \mathbb{E}\left[\sum_{y \neq x, y \in S} \mathbb{1}_{h(y)=h(x)}\right] \\
&= 1 + \sum_{y \in S} \mathbb{E}\left[\mathbb{1}_{h(y)=h(x)}\right] \\
&= 1 + \frac{m-1}{n}
\end{aligned}$$

2 Perfect Hashing

The goal of perfect hashing is to have zero collisions. A 2-level hashing scheme is used.
First level: $h : [U] \to [n]$
Second level: for each $i \in [n]$, $h_i : [U] \to [n_i]$
Fix $n = m = |S|$ and

$$n_i = 4 * \frac{S_i(S_i - 1)}{2}$$

.

where $S_i = \#$ items from S that map to bucket i.

Assuming no collisions in second level hash lookup, the time taken to run query is $O(1)$. The space taken can be computed as follows:

$$\begin{aligned}
space &= first\ level + second\ level \\
&= O(n) + \sum_{i=1}^{n} n_i \\
&= O(n) + O(1) * \left(n + \sum_{i=1}^{n} S_i^2 \right) \\
&= O(n) + O(1) * (n + O(n))
\end{aligned}$$

Claim 3. $\mathbb{E}\left[\sum_{i=1}^{n} S_i^2\right] = 2m$

Proof.

$$\begin{aligned}
\sum_{i=1}^{n} S_i^2 &= \mathbb{E}\left[\sum_{i=1}^{n} S_i * (S_i - 1)\right] + \mathbb{E}\left[\sum_{i=1}^{n} S_i\right] \\
&\left(where \ \left[\sum_{i=1}^{n} S_i * (S_i - 1)\right] = 2 * \# \ collisions \ in \ bucket \ i \right) \\
&= 2 * \frac{m(m-1)}{2} * \frac{1}{n} + m \\
&\leq m + \frac{m^2}{n} \\
&= 2m && as\ n \triangleq m
\end{aligned}$$

Hence it can be said that space required is $O(m)$ in expectation. For $\mathbb{E}[space] \leq 6m$, from Markov bound, we have $Pr[space > 10 \cdot 6m] \leq \frac{1}{10}$.

Full Algorithm:

sample h and check if space $\leq 60m$

if not, then resample h

for each i, sample h_i

check that there are no collisions (probability $\geq \frac{2}{3}$

if collision in h_i, resample it

Time complexity $= O(m) + O(m) + \sum_{i=1}^{n} S_i = O(m)$.

3 Power of 2 choices

Using one hash function: $\mathbb{E}[query\ time] = \mathbb{E}[bucket\ size] = O(1)$ if $n = m$.

Claim 4. *Max load in any bucket is $\Theta(\log n/\log\log n)$ with probability 50%.*

Proof. For proving the upper bound, fix the bucket size as S_i for the i^{th} bucket.

$$\begin{aligned} Pr[S_i \geq k] &\leq \sum_{T \subset S,\ |T|=k} Pr[all\ x \in T in\ bucket\ i] \\ &= {}^nC_k \cdot \frac{1}{n^k} \\ &\leq \left(\frac{en}{k}\right)^k \cdot \frac{1}{n^k} \\ &= \left(\frac{e}{k}\right)^k \end{aligned}$$

We want

$$Pr[S_i \geq k] \leq \frac{1}{4n}$$

so that

$$\begin{aligned} \mathbb{E}[\#\ buckets\ of\ size \geq k] &\leq n \cdot Pr[bucket\ size \geq k] \\ &\leq \frac{1}{4} \\ &\leq 1\ with\ prob \geq 1/2 && from\ Markov\ bound \end{aligned}$$

Hence we can infer about k

$$\left(\frac{e}{k}\right)^k < \frac{1}{4n}$$

hence for a large enough constant c,

$$k = c \cdot \frac{\log n}{\log\log n}$$

Using two hash functions: Consider $h_1, h_2 : [U] \to [n]$. For any x, compute $h_1(x)$ and $h_2(x)$ and put x into the lesser loaded bucket. Such load balancing ensures max load for any bucket is $O(\log\log n)$ with probability $\geq 50\%$.