Lecture 5 (09-12) - Perfect Hashing

The central idea of **Perfect Hashing** is to design hash functions H such that look-ups are guaranteed to be in constant time in the static setting for S.

Of course, on top of this, we want to use as low of space as possible (ideally $O(n)$) and the hash function itself to be fast ($O(1)$).

An $O(N^2)$ Space Solution

Here, we will utilize our earlier universal hashing functions:

Let H be a universal hashing function that maps to an array A with size $M = N^2$. If H detects a collision, it will generate another deterministic hash function pseudo-randomly.
Our claim is that by default, if we hash into an array of size N^2, then H is only expected to choose 2 hash functions. In other words: $P(\text{no collision}) \geq \frac{1}{2}$.

Consider the number of pairs in S: $\binom{N}{2}$, then for each pair, $P(\text{collision}) \leq \frac{1}{M} = \frac{1}{N^2}$.
Therefore, we sum the $P(\text{collision})$ over all pairs with a uniform probability distribution:

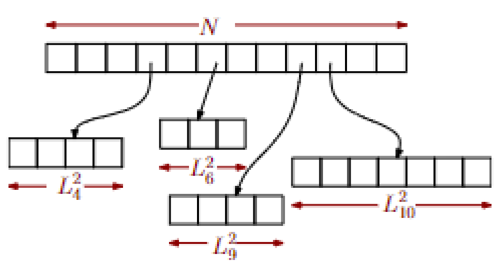
$$\sum_{x \neq y \in S} P(\text{collison}) \leq \frac{\binom{n}{2}}{N^2} = \frac{N(N-1)}{2N^2} \leq \frac{1}{2}$$

Therefore, $P(\text{no collision}) = 1 - P(\text{collision}) \geq \frac{1}{2}$.
Expected number of collisions is therefore at most 2 (we can just retry and we can do so in an expected constant number of times).

In actuality, we can show that *with high probability*, meaning that if we repeat the experiment $c\log(n) \in O(\log(n))$ times, then the probability of a collision is $\frac{1}{N^c}$.

An O(N) Space Solution

It turns out we can do better by extending our previous solution.
Instead of hashing to $M = N^2$, we can instead hash to $M = N$. Then, at each element $A[i]$ in our array, we can store a list L_i of length L_i^2 and hash our elements using the $O(N^2)$ solution:



As we have a static dictionary S, we can perform lookups in constant time.

Space Correctness

Now we should prove that the total space is indeed $O(N)$. First, we know that the space used is equal to

$$\sum_{i=1}^{M} L_i^2$$

Specifically, we can show that:

$$P\left(\sum_{i=1}^{M} L_i^2 > 4N\right) \leq \frac{1}{2}$$

By Markov's inequality, $\frac{E[x]}{a} \geq P(x \geq a)$, we can assert that

$$E[\sum_{i=1}^{M} L_i^2] \leq 2N$$

is a sufficient condition for the above case.

Now, we can show:

$$\begin{aligned} E\left[\sum_{i=1}^{M} L_i^2\right] &= E\left[\sum_{x}\sum_{y} C_{xy}\right] \text{ for random variable } C_{xy} = 1 \text{ for collision, 0 otherwise} \\ &= N + \sum_{x}\sum_{y \neq x} E[C_{xy}] \text{ by removal of collisions of x = y} \\ &= N + \frac{\binom{N}{2}}{M} \text{ by definition of universal hash} \\ &= N + \frac{N(N-1)}{2M} \end{aligned}$$

Given $M = N$, then we get:

$$= N + \frac{N-1}{2} \leq 2N$$

which is what we wanted to show. Once again, we can retry if the probability fails. In this case, we once again only have to retry a constant number of times.

#hashing #algo #universal-hashing #perfect-hashing