Hashing 2

Two-level hashing ("perfect hashing")

Let \( C(i) \) = num. of elements that gets mapped to location \( i \) (in the first level)

If Level 2 \( H_2 \) is \( C(i)^2 \) \( \rightarrow \) collision free for location \( i \)
Total table space for level 2

\[ \Sigma_{i=1}^{M} c(i)^2 \]

We know

\[ E[C] = \binom{N}{2} \frac{1}{M} \]

\[ E \left[ \Sigma_{i=1}^{M} c(i)^2 \right] = \binom{N}{2} \frac{1}{M} \]

\[ E \left[ \Sigma c(i)^2 - \Sigma c(i) \right] = O(N) \quad \text{(since } M = O(N)) \]

\[ \Rightarrow E \left[ \Sigma c(i)^2 \right] = O(N) \]

Collision-free and \( O(N) \) table space!
More stronger properties:

**k-wise independent hash functions**

**Defn:** $H : U \rightarrow [M]$ k-wise indep. if

for any $k$ distinct keys $x_1, \ldots, x_k$ and values $\alpha_1, \alpha_2, \ldots, \alpha_k$

$P \left( h(x_1) = \alpha_1 \cap h(x_2) = \alpha_2 \cap \ldots \cap h(x_k) = \alpha_k \right) \leq \frac{1}{M^k}$

$k = 2$, pair-wise indep.
Properties: \( H \) is \( k \)-wise indip for \( k \geq 2 \). Then

1. \( H \) is also \( (k-1) \)-wise indip.

2. For any \( x \in U \) and \( a \in [M] \), \( P [h(x) = a] \leq \frac{1}{M} \)

3. \( H \) is universal

Pairwise indip vs. universal?

\( h(x) = Ax \)

Consider \( x = 0 \)

\( h(0) = 0 \)

To turn this into pairwise indip: \( h(x) = Ax + b \)
Open addressing
- Single array
- No separate D.S.
- Linear probing
- Use step-size

Quadratic

Cuckoo Hashing

Two tables $T_1$ & $T_2$
both of size $m = O(N)$

Two hash functions $h_1, h_2 \in H$
assume fully random
Insertion:
- \( x \) goes into either \( T_1[h_1(x)] \)
  or \( T_2[h_2(x)] \)

  Stop when no more bumps
  or if more than
  \( \geq \log N \) bumps & rehash

Query:
\( O(1) \) only 2 locations.
**Thm.** The expected time to perform insert is $O(1)$ if $M \geq 4N$

**Proof Sketch:**

- **Cuckoo graph** $G_L$.
  - $M$ vertices: hash table locations.
  - Edges: correspond to the items to be inserted.

  If $x \in S$, $e_x = (h_1(x), h_2(x))$ for all
"y" can get bumped out when inserting a new element x if y falls in a path in the cuckoo graph starting from h₁(x) or h₂(x).

**Def:** Bucket of x  
\[ B(x) = \text{set of nodes of } G \text{ reachable from } h₁(x) \text{ or } h₂(x) \]

*Random*  
\[ = \text{connected component of } G \text{ with edge } ex \]

\[ E[\text{insertion time of } x] = E[|B(x)|] \]

*Size = 1* \[ B(x) \]
To show: \( E[|B(x)|] \leq O(1) \)

\[
E[|B(x)|] = \sum_{y \neq x} p[ e_y \in B(x) ]
\]

\[
\leq N \sum_{y \neq x} p[ e_y \in B(x) ]
\]

Sufficient to show

\[
p[ e_y \in B(x) ] \leq O(\frac{1}{M}) \quad \text{if} \quad M \geq 4N.
\]
Lemma: For any $i, j \in \{1, M\}$

\[ P[\text{there exists a path of length } l \text{ between } i \text{ and } j \text{ in the graph}] \leq \frac{1}{2^l M} \]

Proof: (exercise)

For $l = 1$ ...

Then use induction.
To show: \( P(\{y \in B(x)\}) \leq \Theta\left(\frac{1}{M}\right) \)

Proof: Using the lemma

\[
P(\{y \in B(x)\}) \leq \frac{1}{2^r M}
\]

\[
= O\left(\frac{1}{M}\right)
\]

\( M > 4N \)
Application: Bloom Filter

Membership query

Room for mistakes:

Only false positives

but no false negatives

Useful for ‘filter operations’ - typically elements not in the set

Space efficient data structure for approximate membership queries
. Array $T$ of $M$ bits

. $k$ hash functions $h_1, h_2, \ldots, h_k : V \rightarrow [M]$

  (Assume completely random)

Adding a key:

$x \in S$ set bits $T[h_1(x)], T[h_2(x)] \ldots T[h_k(x)]$ to 1

Membership query: check locations

$x : T[h_1(x)] \ldots T[h_k(x)]$
Let $p = \text{prob. that a bit in } T \text{ is not set}

$$p = \left(1 - \frac{1}{3^{1/n}}\right)^k = (1-p)^k$$

Assume $k \geq \frac{1}{\log(1/p)}$ for all $k$ bits set.
\[
\frac{d}{dh}(\_\_\_) = 0 \quad \text{(Exercise)}
\]

False positive prob. minimized

\[
k = \frac{M \ln 2}{N}
\]

\[
\varepsilon = \left(\frac{1}{2}\right)^{\frac{M \ln 2}{N}}
\]

\[
M = 1.44 \log \left(\frac{1}{\varepsilon}\right)
\]
For 1% false positive:

\[ M = 10N \text{ bits} \]
\[ k = 7 \]