Analysis of Algorithms I: Universal Hashing

Xi Chen

Columbia University
Goal: Let $U$ denote a (very large) universe set. Need a data structure to handle any sequence of $n$ dictionary operations:

$$\text{OP}_1(k_1), \text{OP}_2(k_2), \ldots, \text{OP}_n(k_n)$$

where $k_1, \ldots, k_n \in U$ are keys and $\text{OP}_i \in \{\text{Search, Insert, Delete}\}$. 

Introduction
Given a sequence of \( n \) operations, we let \( S_0 = \emptyset \) and let

\[
S_i = \text{subset of } U \text{ we get after the first } i \text{ operations}
\]

It is clear that \( |S_i| \leq n \) for any \( i \). The sets \( S_i \)'s are completely determined by the sequence of operations and do not depend on the data structure (or the hash function we use as in a hash table).
Suppose we use a hash table $T[0 \ldots m − 1]$ of size $m$ to handle a sequence of $n$ operations. Let $h : U \rightarrow \{0, 1, \ldots, m − 1\}$ be the hash function we use, then the $i$th operation $OP_i(k_i)$ takes time:

$$\text{time needed to compute } h(k_i) + O(\text{COL}_h(k_i, S_{i−1}))$$

where we use $\text{COL}_h(k, S)$ to denote the number of collisions between key $k$ and keys in $S$, with respect to $h$:

$$\text{COL}_h(k, S) = \left| \{ y \in S : h(k) = h(y) \} \right|$$

So $\text{COL}_h(k_i, S_{i−1})$ is the length of the list at slot $h(k_i)$ before $OP_i$. 

Introduction
As a result, if the evaluation of $h$ can always be done in constant many steps, the total running time is

$$O(n) + O\left(\sum_{i=1}^{n} \text{COL}_h(k_i, S_{i-1})\right)$$

In the last class we showed that no matter which hash function $h$ is used, there always exists a sequence of $n$ operations that leads to $\Omega(n^2)$ total running time when $|U|$ is large enough (e.g., $\geq nm$). This is unavoidable if we try to fix a hash function and use it to handle all possible sequences of dictionary operations.
Instead, we show how to randomly and properly pick (or build) a hash function so that for any sequence, the total running time is $O(n)$ in expectation. This method is usually referred to as Universal Hashing.
**Definition**

Let $H$ be a collection of hash functions from $U$ to $\{0, \ldots, m-1\}$. We say it is universal if for any two distinct keys $x$ and $y$ from $U$:

\[
\text{the number of functions } h \in H \text{ such that } h(x) = h(y) \leq |H|/m.
\]

A corollary from the definition: If we pick a hash function $h$ from $H$ uniformly at random (each with probability $1/|H|$), then

\[
\Pr[h(x) = h(y)] \leq 1/m, \quad \text{for all } x \neq y \in U
\]

That is, for any two keys $x$ and $y$, the probability that there is collision between them (with respect to $h$) is bounded by $1/m$. 
Theorem

Assume there is a universal collection $H$ in which every function $h$ can be evaluated in $O(1)$ steps. Then given any sequence of $n$ operations, if we pick a hash function $h$ from $H$ uniformly at random, then the total running time is

$$O(n + (n^2/m))$$

in expectation.
By the linearity of expectations, the expected total running time is

\[ O(n) + O \left( \sum_{i=1}^{n} E[\text{COL}_h(k_i, S_{i-1})] \right) \]

it suffices to show that for every \( i \in [n] \), we have

\[ E[\text{COL}_h(k_i, S_{i-1})] < \left( \frac{n}{m} \right) + 1 \]  \hspace{1cm} (1)
To prove (1), we first consider the case when \( k_i \in S_{i-1} \). Because \( |S_{i-1}| \leq n \), there are at most \((n - 1)\) keys \( y \in S_{i-1} \) other than \( k_i \). For each such \( y \), we use \( X_y \) to denote the indicator \( \{0, 1\} \) random variable which is 1 if \( h(y) = h(k_i) \) and is 0 otherwise. Then by the definition of \( \text{COL}_h \) and the linearity of expectations, we have

\[
E[\text{COL}_h(k_i, S_{i-1})] = E \left[ 1 + \sum_{y \in S_{i-1} - \{k_i\}} X_y \right] \\
= 1 + \sum_{y} E[X_y] = 1 + \sum_{y} \Pr[X_y = 1] \\
= 1 + \frac{(n - 1)}{m} < \frac{n}{m} + 1
\]
Here the last equation uses the fact that

$$\Pr[X_y = 1] = 1/m$$

This comes from the assumption that $H$ is universal (and this is the only place we use the assumption that $H$ is universal). The other case when $k_i \not\in S_{i-1}$ can be proved similarly.
But does such a universal collection $H$ exist? Next we present a construction of $H$ when $p = |U|$ is a prime. (What if $|U|$ is not a prime? Either find a prime $p$ that is a little larger than $|U|$ and use \{0,1,\ldots,p-1\} as the universe set instead; or use a construction that does not need this assumption. Google for other constructions of universal collections if interested.)
Assume $p$ is a prime. Let

$$\mathbb{Z}_p = \{0, 1, 2, \ldots, p - 1\} \quad \text{and} \quad \mathbb{Z}^*_p = \{1, 2, \ldots, p - 1\}$$

So $U = \mathbb{Z}_p$. For every pair $(a, b)$ where $a \in \mathbb{Z}^*_p$ and $b \in \mathbb{Z}_p$, let

$$h_{ab}(k) = (ak + b \mod p) \mod m$$

be a hash function from $U$ to $\{0, 1, \ldots, m\}$. Set

$$H = \{h_{ab} : a \in \mathbb{Z}^*_p \text{ and } b \in \mathbb{Z}_p\}$$

so $H$ contains $(p - 1)p$ functions.
This collection $H$ has all the properties we need: It is very easy to pick a hash function $h$ from $H$ randomly: just pick $a$ from $\mathbb{Z}_p^*$ and $b$ from $\mathbb{Z}_p$ uniformly at random and set $h = h_{ab}$. Evaluation of each $h \in H$ only takes $O(1)$ steps. Most importantly, $H$ is universal:

**Theorem**

*When $p$ is a prime, $H$ is a universal collection of hash functions.*
Let \( k \neq \ell \) be two different keys from \( U = \mathbb{Z}_p \). We need to count the number of pairs \((a, b)\), where \( a \in \mathbb{Z}_p^* \) and \( b \in \mathbb{Z}_p \), such that

\[ h_{ab}(k) = h_{ab}(\ell) \]

and show that it is no more than

\[ \frac{|H|}{m} = \frac{p(p - 1)}{m} \]
To this end we construct the following function:

\[ g : \mathbb{Z}_p^* \times \mathbb{Z}_p \rightarrow \mathbb{Z}_p \times \mathbb{Z}_p \]

where \((r, s) = g(a, b)\) if

\[ r = ak + b \mod p \quad \text{and} \quad s = a\ell + b \mod p \]

Using \(g\), we now need to count the number of pairs \((a, b)\) such that \((r, s) = g(a, b)\) satisfies

\[ r \mod m = s \mod m \quad (2) \]
Next we prove that the map $g$ defined in the last slide is indeed a one-to-one correspondence between $\mathbb{Z}_p^* \times \mathbb{Z}_p$ and

$$\{(r, s) \in \mathbb{Z}_p \times \mathbb{Z}_p : r \neq s\}$$

To prove this, we need to show that

1. When $r = s$, there exists no $(a, b) \in \mathbb{Z}_p^* \times \mathbb{Z}_p$ such that $g(a, b) = (r, s)$; and

2. When $r \neq s$, there exists exactly one $(a, b) \in \mathbb{Z}_p^* \times \mathbb{Z}_p$ such that $g(a, b) = (r, s)$.

Both can be proved using the assumption that $p$ is prime.
Once we know that $g$ is a one-to-one correspondence between $(a, b) \in \mathbb{Z}_p^* \times \mathbb{Z}_p$ and $(r, s) \in \mathbb{Z}_p \times \mathbb{Z}_p$ with $r \neq s$, we have

number of $(a, b) \in \mathbb{Z}_p^* \times \mathbb{Z}_p$ s.t. $(r, s) = g(a, b)$ satisfies (2)

is exactly the same as

number of $(r, s) \in \mathbb{Z}_p \times \mathbb{Z}_p$ that satisfies $r \neq s$ and (2)
It is much simpler to count the number of \((r, s)\) such that \(r \neq s\) and (2) is satisfied. Fix \(r\) to be any number from \(\{0, 1, \ldots, p - 1\}\). Then to satisfy both conditions, \(s\) can only be

\[
\ldots, r - 2m, r - m, r + m, r + 2m, \ldots
\]

Assume there are \(q_1\) many possible \(s\)'s smaller than \(r\): \(r - q_1 m, \ldots, r - m\) and \(q_2\) many possible \(s\)'s larger than \(r\): \(r + m, \ldots, r + q_2 m\). Because \(r - q_1 m \geq 0\) and \(r + q_2 m \leq p - 1\), we have

\[
(r + q_2 m) - (r - q_1 m) \leq p - 1
\]

and thus, the total number of possible \(s\)'s is \(q_1 + q_2 \leq (p - 1)/m\).
Therefore, the total number of \((r, s)\) that satisfies \(r \neq s\) and (2) is

\[ \leq p \cdot \frac{p - 1}{m} \]

Since the total number of hash functions in \(H\) is \(p(p - 1)\), we can finally conclude that \(H\) is universal.