Amortised Analysis and Dynamic Tables

Jie Wang

University of Massachusetts Lowell
Department of Computer Science

Hash Tables

- A hash table T is a data structure consisting of
  1. An n element array (table), where each element is called a *hash bucket*.
  2. A hash function $h : U \to \{0, 1, \ldots, n-1\}$, where each element of U is called a *key*.
- Assumption: The runtime of computing $h(x)$ is $O(1)$, regardless of x.
- A hash table supports two operations :
  1. $\textsc{Insert}(k, v)$: Insert key k with value v into table T at location $h(k)$; i.e., $T[h(k)] = v$.
  2. $\textsc{Lookup}(k)$: Return the value associated with key k.

Collision Handling

Three methods to handle collisions:

1. Failing.
2. Separate chaining. That is, $T[h(k)]$ heads a linked list of pairs (k, v).
   - Insertion runtime: $O(1)$.
   - Lookup runtime: $O(1 + \alpha(T))$, where $\alpha(T) = m/n$, called the *load factor*, with m being the number of insertions so far.
3. Open addressing.

Open Addressing

- Every bucket holds one element.
  - A sequence of locations is used to find an empty slot for insertion or the correct entry looked for.
  - This sequence is called a *probe sequence*.
  - To bootstrap probe sequences, modify hash function to include a probe number.
  - The probe number is the number of insertion collisions so far.
- $\textsc{Insert}(k, v)$ stores the key-value pair in the first cell $T[h(k, i)]$ that is empty.
  - The runtime for insertion is proportional to the length of the probe sequence.
  - This is related to the load factor.
- $\textsc{Lookup}(k)$ searches location $T[h(k, i)]$ for all i until the key-value pair is found or until an empty slot is discovered.
  - Incur the same running time as $\textsc{Insert}$.

Probing Strategies

Three common probing strategies for open addressing, with an initial probe number $i = 0$:

1. Linear Probing
   - Given a base hash function $h' : U \to \{0, 1, \ldots, n-1\}$, construct the following hash function:

$$h(k, i) = (h'(k) + i) \bmod n.$$

2. Quadratic Probing
   - Given a base hash function $h' : U \to \{0, 1, \ldots, n-1\}$, construct the following hash function:

$$h(k, i) = (h'(k) + c_1 i + c_2 i^2) \bmod n,$$

   where c_1 and c_2 are positive constants.

3. Double Hashing
   - Given two different base hash functions $h_1 : U \to \{0, 1, \ldots, n-1\}$ and $h_2 : U \to \{0, 1, \ldots, n-1\}$, construct the following hash function:

$$h(k, i) = (h_1(k) + ih_2(k)) \bmod n.$$

Table doubling or table halving is a technique for *growing* or *contracting* a table. First look at hash-table doubling.

- When the number of insertions m into the table is the same as the table size n, double the size of the table.
- To double the size of a hash table:
  1. Allocate a new table T' of size $2n$ initializing every entry to $\textsc{Nil}$.
  2. Insert all m entries in T into T' using a *new* hash function.
- Table doubling takes m insertions.
- Table doubling doesn't happen too often.
  - After the size of table is doubled to m', it takes $\frac{m'}{2}$ insertions before doubling must happen again.
  - Starting with an empty table, table doubling incurs $\lfloor \log m \rfloor$ times.

Potential Amortized Analysis

- When deletion is allowed, tables can also be contracted when the load factor becomes too small.
- Want to figure out the cost of a sequence of m operations starting from an empty table.
- Two observations:
  1. Insertions and deletions form potential energy for the next possible expansion or contraction.
  2. When $\alpha(T) = 1/2$, the table is half full; no need to double or contract.
- Using $\alpha(T) = 1/2$ as a threshold value to define a potential function:

$$\Phi(T) = \begin{cases} 2\ell - n, & \text{if } \alpha(T) \geq 1/2, \\ n/2 - \ell, & \text{if } \alpha(T) < 1/2, \end{cases}$$

where $\alpha(T) = \ell/n$.

Φ Is A Legitimate Potential Function

- If $T = \emptyset$, then $\ell = 0$ and $n = 0$. Thus, $\Phi(T) = 0$.
- If $\alpha(T) = 1/2$, then $\ell = n/2$. Thus, $\Phi(T) = 2(n/2) - n = n - n = 0$.
- If $\alpha(T) = 1$ (i.e., T is full), then $\ell = n$. Thus, $\Phi(T) = 2n - n = n$.
  - With potential n it is sufficient to pay for expansion when an item is inserted.
- If $\alpha(T) = 1/4$, then shrink the table.
  - This means that $\ell/n = 1/4$, which implies $4\ell = n$. Thus,

    $$\Phi(T) = 4\ell/2 - \ell = \ell,$$

    which is enough to pay for contraction.
- The potential is nonnegative for the following reasons:
  - The number n cannot be greater than 2ℓ when $\alpha(T) \geq 1/2$.
  - The number ℓ cannot be greater than $n/2$ when $\alpha(T) < 1/2$.

Analysis on m Consecutive Operations

Start with an empty table. That is,

$$\Phi(T_0) = 0, n = 0, \ell = 0, \alpha(T_0) = 1.$$

Suppose that the i^{th} operation is INSERT.

- **Case 1.** $\alpha(T_{i-1}) < 1/2$ and $\alpha(T_i) < 1/2$; no expansion.

$$\begin{aligned}
\hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\
&= 1 + \left(\frac{n_i}{2} - \ell_i\right) - \left(\frac{n_{i-1}}{2} - \ell_{i-1}\right) \\
&= 1 + \left(\frac{n_{i-1}}{2} - (\ell_{i-1} + 1)\right) - \left(\frac{n_{i-1}}{2} - \ell_{i-1}\right) \\
&= 0.
\end{aligned}$$

Analysis Continued

- **Case 2.** $\alpha(T_{i-1}) < 1/2$ and $\alpha(T_i) \geq 1/2$ and the table does not expand.

$$
\begin{aligned}
\hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\
&= 1 + (2\ell_i - n_i) - \left(\frac{n_{i-1}}{2} - \ell_{i-1}\right) \\
&= 1 + (2(\ell_{i-1} + 1) - n_{i-1}) - \left(\frac{n_{i-1}}{2} - \ell_{i-1}\right) \\
&= 3 + 3\ell_{i-1} - \frac{3n_{i-1}}{2} \\
&= 3 + 3\alpha(T_{i-1})n_{i-1} - \frac{3n_{i-1}}{2} \\
&< 3 + \frac{3n_{i-1}}{2} - \frac{3n_{i-1}}{2} \\
&= 3.
\end{aligned}
$$

Analysis Continued

- **Case 3.** $\alpha(T_{i-1}) \geq 1/2$ and $1 > \alpha(T_i) \geq 1/2$ and the table does not expand.

$$\begin{aligned}
\hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\
&= 1 + (2\ell_i - n_i) - (2\ell_{i-1} - n_{i-1}) \\
&= 1 + (2(\ell_{i-1} + 1) - n_{i-1}) - (2\ell_{i-1} - n_{i-1}) \\
&= 3.
\end{aligned}$$

- **Case 4.** $\alpha(T_{i-1}) \geq 1/2$ and the table expands. That is, $n_i = 2n_{i-1}$, $\ell_i = n_{i-1}$, and $c_i = \ell_i$ (the total number of insertion).
  Thus, $\alpha(T_i) = 1/2$.

$$\begin{aligned}
\hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\
&= \ell_i + (2\ell_i - n_i) - (2\ell_{i-1} - n_{i-1}) \\
&= \ell_i + (2n_{i-1} - 2n_{i-1}) - (2\ell_{i-1} - (\ell_{i-1} + 1)) \\
&= \ell_i - \ell_{i-1} + 1 \\
&= 2.
\end{aligned}$$

Analysis Continued

Suppose that the i^{th} operation is DELETE.

- **Case 1.** $\alpha(T_{i-1}) < 1/2$ and the table is not contracted.

$$
\begin{aligned}
\hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\
&= 1 + \left(\frac{n_i}{2} - \ell_i\right) - \left(\frac{n_{i-1}}{2} - \ell_{i-1}\right) \\
&= 1 + \left(\frac{n_{i-1}}{2} - (\ell_{i-1} - 1)\right) - \left(\frac{n_{i-1}}{2} - \ell_{i-1}\right) \\
&= 2.
\end{aligned}
$$

Analysis Continued

- **Case 2.** $\alpha(T_{i-1}) < 1/2$ and the table is contracted. That is, $n_{i-1} = 4\ell_{i-1}$ and $c_i = \ell_{i-1}$.

$$
\begin{aligned}
\hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\
&= \ell_{i-1} + \left(\frac{n_i}{2} - \ell_i\right) - \left(\frac{n_{i-1}}{2} - \ell_{i-1}\right) \\
&= \ell_{i-1} + \left(\frac{n_{i-1}}{4} - (\ell_{i-1} - 1)\right) - \left(\frac{n_{i-1}}{2} - \ell_{i-1}\right) \\
&= 1 - \frac{n_{i-1}}{4} + \ell_{i-1} \\
&= 1.
\end{aligned}
$$

Analysis Continued

- **Case 3.** $\alpha(T_{i-1}) \geq 1/2$ and $\alpha(T_i) < 1/2$; no contraction. Then

$$
\begin{aligned}
\hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\
&= 1 + \left(\frac{n_i}{2} - \ell_i\right) - (2\ell_{i-1} - n_{i-1}) \\
&= 1 + \left(\frac{n_{i-1}}{2} - \ell_{i-1} + 1\right) - (2\ell_{i-1} - n_{i-1}) \\
&= 2 + \frac{3n_{i-1}}{2} - 3\ell_{i-1} \\
&= 2 + \frac{3n_{i-1}}{2} - 3\alpha(T_{i-1})n_{i-1} \\
&\leq 2 + \frac{3n_{i-1}}{2} - 3n_{i-1}/2 \\
&= 2.
\end{aligned}
$$

- **Case 4.** $\alpha(T_{i-1}) \geq 1/2$ and $\alpha(T_i) \geq 1/2$. Thus, no contraction. Then

$$\begin{aligned}
\hat{c}_i &= c_i + \Phi(T_i) - \Phi(T_{i-1}) \\
&= 1 + (2\ell_i - n_i) - (2\ell_{i-1} - n_{i-1}) \\
&= 1 + (2(\ell_{i-1} - 1) - n_{i-1}) - (2\ell_{i-1} - n_{i-1}) \\
&= -1.
\end{aligned}$$

- Thus, the runtime of a sequence of n insertions and deletions on T is $\Theta(n)$.