浅谈一类哈希表的复杂度分析

陈于思

中国人民大学附属中学

2022 年 1 月

哈希函数

约定 $[s]=\{0,1,\ldots,s-1\}$。
哈希表通过哈希函数将键值集合 $\mathcal{U}\subseteq[u]$ 映射到 $[m]$ 上。实际问题中一般 $u\gg n$，且通常我们选取的 m 满足 $m=\Theta(n),m>n$ 。

定义

称 $\mathcal{U}$ 到 $[m]$ 的随机映射为哈希函数。

定义

称从 $|\mathcal{U}|^m$ 个 $\mathcal{U}$ 到 $[m]$ 的映射中等概率随机选取的哈希函数 h 为全真随机哈希函数（truly random hash function）。

定义

称一组随机变量 $X_0, X_1, \ldots, X_{n-1} \in S$ 是 $k-$**独立** ($k-$`wise independent`) 的，如果对任意互不相同的 k 个下标 $i_0, i_1, \ldots, i_{k-1} \in [n]$ 与 k 个定值 $y_0, y_1, \ldots, y_{k-1} \in S$，都有

$$\Pr\left[\bigwedge_{j\in[k]} X_{i_j} = y_j\right] = \frac{1}{m^k}$$

定义

设 $\mathcal{U} = \{x_0, x_1, \ldots, x_{|\mathcal{U}|-1}\}$ 。称一个哈希函数 $h : \mathcal{U} \to [m]$ 是 $k-$ 独立 的，如果随机变量 $h(x_0), h(x_1), \ldots, h(x_{|\mathcal{U}|-1})$ 是 $k-$ 独立 的。

性质

$(k+1)-$ 独立 哈希函数都是 $k-$ 独立 哈希函数。
全真随机哈希函数都是 $k-$ 独立 哈希函数。

例

取质数 $p > u$ 。等概率随机选择 $a_0, a_1, \ldots, a_{k-1} \in [p]$ ，令 $h : \mathcal{U} \to [p]$ 为:

$$h(x) = (a_{k-1}x^{k-1} + \cdots + a_1 x + a_0) \bmod p$$

则 $h'(x) = h(x) \bmod m$ 可以看作是 $k-$ 独立 的。

证明.

$$\Pr[h(x_0) = y_0 \wedge \cdots \wedge h(x_{k-1}) = y_{k-1}] = \frac{1}{p^k}$$

□

定义

称 $h:\mathcal{U}\to[m]$ 是一个全域哈希函数 (universal hash function)，如果 $\forall x_1\neq x_2$ ，都有 $\Pr[h(x_1)=h(x_2)]=\frac{1}{m}$ 。

性质

$2-$ 独立 哈希函数都是全域哈希函数。

独立链法

- 独立链法 (separate chaining) 是一种用于解决哈希冲突的哈希表实现方法。

独立链法

- 独立链法 (separate chaining) 是一种用于解决哈希冲突的哈希表实现方法。
- 独立链法对每个哈希值建立一个初始为空的链表。执行一次有关键值 $x \in \mathcal{U}$ 的操作时，对 $y = h(x)$ 位置的链表进行操作。记 y 对应的链表的大小为 c_y ，那么这次操作的访问的链表元素个数不会超过 c_y。

独立链法

- 独立链法 (separate chaining) 是一种用于解决哈希冲突的哈希表实现方法。
- 独立链法对每个哈希值建立一个初始为空的链表。执行一次有关键值 $x \in \mathcal{U}$ 的操作时，对 $y = h(x)$ 位置的链表进行操作。记 y 对应的链表的大小为 c_y，那么这次操作的访问的链表元素个数不会超过 c_y。
- 我们考察一次操作访问的链表元素个数的最大值，即

$$\max_{t=0}^{m-1} c_t$$

定理

对全域哈希函数 h，独立链法单次操作复杂度为 $O\left(\frac{\log n}{\log\log n}\right)$ 的概率足够高。事实上，如果 $\alpha = \frac{n}{m}$ 是一个定值，则对任意 $b > 0$ 均存在 a 使得下式恒成立：

$$\Pr\left[\max_{y\in[m]} c_y > \frac{a\ln n}{\ln\ln n}\right] < n^{-b}$$

证明.

- 考察 $t \in [m]$ 对应的链表长度 c_t。

证明.

- 考察 $t \in [m]$ 对应的链表长度 c_t。
- 记 $\mu = \mathrm{E}[c_t] = \frac{n}{m}$，则由 Chernoff bound 有

$$\Pr[c_t > (1+\delta)\mu] < \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\mu}$$

$\square$

- 取 $\lambda = \frac{b+1+\epsilon}{\mu}, \delta = \frac{\lambda \ln n}{\ln \ln n} - 1$，则

$$
\begin{aligned}
\Pr[c_t > (1+\delta)\mu] &< \left(\frac{e^{\frac{\lambda \ln n}{\ln \ln n} - 1}}{e^{\frac{\lambda \ln n}{\ln \ln n} \ln\left(\frac{\lambda \ln n}{\ln \ln n}\right)}} \right)^{\mu} \\
&< \left(\frac{n^{\frac{\lambda}{\ln \ln n}}}{e^{\lambda \ln n \left(1 + \frac{\ln \lambda - \ln \ln \ln n}{\ln \ln n}\right)}} \right)^{\mu} \\
&= n^{\lambda\mu\left(\frac{1 + \ln \ln \ln n - \ln \lambda}{\ln \ln n} - 1\right)} \\
&= O\left(n^{-b-1}\right)
\end{aligned}
$$

- 于是

$$\Pr\left[\max_{y\in[m]} c_y > (1+\delta)\mu\right] < \sum_{y\in[m]} \Pr[c_y > (1+\delta)\mu] = O\left(n^{-b}\right)$$

线性探查法

- 线性探查法 (linear probing) 是一种哈希表实现方法。

线性探查法

- 线性探查法 (linear probing) 是一种哈希表实现方法。
- 线性探查法首先建立一个长为 m 的数组 T，每个位置存储一对 (x, v)。

线性探查法

- 线性探查法 (linear probing) 是一种哈希表实现方法。
- 线性探查法首先建立一个长为 m 的数组 T，每个位置存储一对 (x, v)。
- 执行 `Insert(x,v)` 时，找到 x 对应的哈希值 $y = h(x)$，如果 T_y 是空的，就将 (x, v) 存储在 T_y 位置；如果 T_y 已经被占用了，扫描 $T_{y+1}, T_{y+2}, \ldots, T_{m-1}, T_0, T_1, \ldots$，找到第一个未被占用的位置，将 (x, v) 存储在这个位置。

线性探查法

- 线性探查法 (linear probing) 是一种哈希表实现方法。
- 线性探查法首先建立一个长为 m 的数组 T，每个位置存储一对 (x, v)。
- 执行 `Insert(x,v)` 时，找到 x 对应的哈希值 $y = h(x)$ ，如果 T_y 是空的，就将 (x, v) 存储在 T_y 位置；如果 T_y 已经被占用了，扫描 $T_{y+1}, T_{y+2}, \ldots, T_{m-1}, T_0, T_1, \ldots$ ，找到第一个未被占用的位置，将 (x, v) 存储在这个位置。
- 执行修改、删除和查询操作时，从 T_y 开始向后扫描，直到找到一个位置的键值等于 x，对这个位置进行操作。

- 下面我们说明对于 $5-$ 独立 的哈希函数，线性探查法单次操作的最坏复杂度期望是 $O(1)$ 的。

- 下面我们说明对于 $5-$ 独立 的哈希函数，线性探查法单次操作的最坏复杂度期望是 $O(1)$ 的。
- 显然，对键值 $x \in [u]$，x 将会被存储在 $h(x)$ 及以后的第一个空位置。执行一次关于 x 的操作所用的时间不会超过 $h(x)$ 到下一个的空位置的距离。

- 下面我们说明对于 $5-$ 独立 的哈希函数，线性探查法单次操作的最坏复杂度期望是 $O(1)$ 的。
- 显然，对键值 $x \in [u]$，x 将会被存储在 $h(x)$ 及以后的第一个空位置。执行一次关于 x 的操作所用的时间不会超过 $h(x)$ 到下一个的空位置的距离。
- 为方便起见，下文中我们认为 $m \geq \frac{3}{2}n$ 且 m 是 2 的幂。

- 下面我们说明对于 $5-$ 独立 的哈希函数，线性探查法单次操作的最坏复杂度期望是 $O(1)$ 的。
- 显然，对键值 $x \in [u]$，x 将会被存储在 $h(x)$ 及以后的第一个空位置。执行一次关于 x 的操作所用的时间不会超过 $h(x)$ 到下一个的空位置的距离。
- 为方便起见，下文中我们认为 $m \geq \frac{3}{2}n$ 且 m 是 2 的幂。
- 设目前哈希表中已经存储了一个大小为 n 的键值集合 $S \subseteq \mathcal{U}$ ，现在要执行一次 `Query(q)`。

定义

称一个区间 $R = [l, r]$ 是一个极大连续段，如果 R 中每个位置都被占用了，且 $l - 1$ 与 $r + 1$ 均是空的。

性质

包含 $h(q)$ 的极大连续段 R 唯一，且 $\texttt{Query}(q)$ 访问的位置数不会超过 $|R| + 1$ 。

定义

称一个形如 $[i2^l,(i+1)2^l)$ 的区间是一个 $l-$ 区间，其中 $i \in \left[\frac{m}{2^l}\right]$。

定义

称一个 $l-$ 区间 I 是危险的，如果 S 中至少有 $\frac{3}{4}2^l$ 个元素对应的哈希值属于 I。

引理

对于一个长为 $r \geq 2^{l+2}$ 的极大连续段 R，与 R 有交的前 4 个 $l-$ 区间 中至少有一个是危险的。

证明.

设与 R 有交的前 4 个 $l-$ 区间 为 I_0, I_1, I_2, I_3，则 I_0 的最后一个元素属于 R，且 $I_1, I_2, I_3 \subset R$ ，于是 $L = R \cap \left(\bigcup_{i=0}^{3} I_i\right)$ 满足 $|L| \geq 3 \times 2^l + 1$ 。 $\square$

而 L 是 R 的一个前缀，从而如果 x 被存储在 L 中，那么 $h(x) \in L$。从而 $|\{x \in S \mid h(x) \in L\}| \geq |L| \geq 3 \times 2^l + 1$，因此

$$\sum_{i=0}^{3} |\{x \in S \mid h(x) \in I_i\}| \geq 3 \times 2^l + 1$$

从而存在 $i \in [4]$ 使 $|\{x \in S \mid h(x) \in I_i\}| \geq \frac{3}{4}2^l$，即 I_i 是危险的。

引理

如果包含 q 的极大连续段 R 的长度 r 满足 $2^{l+2} \le r < 2^{l+3}$，则以下 12 个 $l-$区间 中至少有一个区间是危险的：包含 $h(q)$ 的 $l-$区间，它左侧的 8 个 $l-$区间 以及它右侧的 3 个 $l-$区间。

证明.

考虑前述引理中的四个区间 I_0, I_1, I_2, I_3。由于 $r < 8 \times 2^l$，故 I_0 一定是包含 $h(q)$ 的 $l-$区间 和它左侧的 8 个 $l-$区间 之一。于是这四个区间包含于引理所述的 12 个区间，从而这 12 个区间中至少有一个是危险的。 □

显然，每个 $l-$ 区间 危险的概率是相等的。记 P_l 为一个 $l-$ 区间 危险的概率。

定理

$Query(q)$ 的期望复杂度为

$$O\left(1+\sum_{l=0}^{\log_2 m} 2^l P_l\right)$$

证明.

- 考虑包含 q 的极大连续段 R，记 $r = |R|$。

证明.

- 考虑包含 q 的极大连续段 R，记 $r = |R|$。
- 若 $r \geq 4$，设引理中的 12 个区间分别为 $J_0, J_1, \ldots, J_{11}$，则

$$
\begin{aligned}
&\Pr\left[r \in \left[2^{l+2}, 2^{l+3}\right)\right] \\
\leq &\Pr\left[J_0, J_1, \ldots, J_{11}\text{中至少有一个是危险的}\right] \\
\leq &\sum_{i=0}^{11} \Pr\left[J_i\text{是危险的}\right] = 12P_l
\end{aligned}
$$

证明.

- 考虑包含 q 的极大连续段 R，记 $r = |R|$。
- 若 $r \geq 4$，设引理中的 12 个区间分别为 $J_0, J_1, \ldots, J_{11}$，则

$$\begin{aligned}
&\Pr\left[r \in \left[2^{l+2}, 2^{l+3}\right)\right] \\
\leq &\Pr\left[J_0, J_1, \ldots, J_{11}\text{中至少有一个是危险的}\right] \\
\leq &\sum_{i=0}^{11} \Pr\left[J_i\text{是危险的}\right] = 12P_l
\end{aligned}$$

- 于是

$$\mathrm{E}[r] < \sum_{l=0}^{\log_2 m} 2^{l+1} \Pr\left[r \in \left[2^l, 2^{l+1}\right)\right] = O\left(1 + \sum_{l=0}^{\log_2 m} 2^l P_l\right)$$

引理

如果随机变量 $X_0, X_1, \ldots, X_{n-1} \in \{0,1\}$ 是 $4-$ 独立 的，$X = \sum_{i=0}^{n-1} X_i$， $\mu = \mathrm{E}[X] \geq 1$，则

$$\Pr\left[|X-\mu| \geq d\sqrt{\mu}\right] \leq \frac{4}{d^4}$$

定理

$m \geq \frac{3}{2}n$ 且 m 是 2 的幂时，如果选取一个 $5-$ 独立 的哈希函数 $h : \mathcal{U} \to [m]$，那么线性探查法单次操作的最坏复杂度期望是 $O(1)$。

证明.

- 考虑固定 q 对应的哈希值 $h(q) = y_q$ ，那么对于不同的四个键值 $x_0, x_1, x_2, x_3 \in \mathcal{U} \setminus \{q\}$ 与四个哈希值 $y_0, y_1, y_2, y_3 \in [m]$，有 $\Pr\left[\bigwedge_{i=0}^{3} h(x_i) = y_i \mid h(q) = y_q\right] = m^{-4}$
- 从而 S 中所有键值对应的哈希值是 $4-$ 独立 的。

$\square$

- 对于一个 $l-$ 区间 I，考虑它危险的概率。对 $x \in S$，记 $X_x = [h(x) \in I]$。于是，$X = \sum_{x \in S} X_x$ 即为 S 中哈希值落在 I 中的元素个数，I 危险当且仅当 $X \geq \frac{3}{4} 2^l$。

- 对于一个 $l-$ 区间 I，考虑它危险的概率。对 $x \in S$，记 $X_x = [h(x) \in I]$。于是，$X = \sum_{x \in S} X_x$ 即为 S 中哈希值落在 I 中的元素个数，I 危险当且仅当 $X \geq \frac{3}{4} 2^l$。
- 记 $\mu = \mathrm{E}[X]$，则 $\mu = \frac{n 2^l}{m} \leq \frac{2}{3} 2^l$，于是

$$X \geq \frac{3}{4} 2^l \implies X - \mu > \frac{1}{12} 2^l > \frac{1}{10} \sqrt{2^l \mu}$$

- 对于一个 $l-$ 区间 I，考虑它危险的概率。对 $x \in S$，记 $X_x = [h(x) \in I]$。于是，$X = \sum_{x \in S} X_x$ 即为 S 中哈希值落在 I 中的元素个数，I 危险当且仅当 $X \geq \frac{3}{4}2^l$。
- 记 $\mu = \mathrm{E}[X]$，则 $\mu = \frac{n2^l}{m} \leq \frac{2}{3}2^l$，于是

$$X \geq \frac{3}{4}2^l \implies X - \mu > \frac{1}{12}2^l > \frac{1}{10}\sqrt{2^l \mu}$$

- $$\Pr\left[X \geq \frac{3}{4}2^l\right] \leq \Pr\left[X - \mu \geq \frac{1}{10}\sqrt{2^l \mu}\right] \leq \frac{40000}{2^{2l}}$$

- 从而单次操作的最坏复杂度期望是

$$O\left(1+\sum_{l=0}^{\log_2 m} 2^l P_l\right) = O\left(1+\sum_{l=0}^{\log_2 m} 2^l \times 2^{-2l}\right) = O(1)$$

推论

$m \geq 3n$ 时，选取一个 $5-$ 独立 的哈希函数 $h : \mathcal{U} \to [m]$，线性探查法单次操作的最坏复杂度期望是 $O(1)$。

谢谢大家！