Optimal Bounds for Open Addressing Without Reordering

Martin Farach-Colton , Andrew Krapivin , William Kuszmaul

Presented by - Amit Roy

March 27, 2025

Hashing

Collision Resoutions

- Chaining - Each slot points to pointer to a linked list which stores the keys
- Open Addressing - All elements occupy the hash table itself

Definitions

Probe Complexity

The number of slots in the array

Results

1. Greedy
   - Worst-case expected probe complexity $\mathcal{O}(\log^2 \delta^{-1})$
   - High-probability worst-case probe complexity $\mathcal{O}(\log^2 \delta^{-1} + \log \log n)$
     - ★ Matching lower bound

2. Non-Greedy
   - Amortized probe complexity $\mathcal{O}(1)$
   - Worst-case expected probe complexity $\mathcal{O}(\log \delta^{-1})$
     - ★ Matching lower bound

Theorem - Non-greedy Open-Addressing

Let $n \in \mathbb{N}$ and $\delta \in (0, 1)$ be parameters such that $\delta > \mathcal{O}(1/n)$. There exists an open-addressing hash table that supports $n - \lfloor \delta n \rfloor$ insertions in an array of size n, that does not reorder items after they are inserted, and that offers

- amortized expected probe complexity $O(1)$
- worst-case expected probe complexity $\mathcal{O}(log \delta^{-1})$, and
- worst-case expected insertion time $\mathcal{O}(log \delta^{-1})$.

Theorem - Greedy Open-Addressing

Let $n \in \mathbb{N}$ and $\delta \in (0, 1)$ be parameters such that $\delta > \mathcal{O}(1/n^{o(1)})$. There exists a greedy open-addressing strategy that supports $n - \lfloor \delta n \rfloor$ insertions that has

- worst-case expected probe complexity (and insertion time) - $\mathcal{O}(\log^2 \delta^{-1})$
- worst-case probe complexity over all insertions - $\mathcal{O}(\log^2 \delta^{-1} + \log \log n)$, with prob $1 - 1/poly(n)$,
- amortized expected probe complexity - $\mathcal{O}(\log \delta^{-1})$

Funnel Hashing

Lemma

For a given $i \in \alpha$, we have with probability $1 - \frac{1}{n^{\omega(1)}}$ that, after $2|A_i|$ insertion attempts have been made in A_i, fewer than $\frac{\delta}{64}|A_i|$ slots in A_i remain unfilled.

Lemma

The number of keys inserted into $A_{\alpha+1}$ is fewer than $\frac{\delta}{8}n$, with probability $1 - \frac{1}{n^{\omega(1)}}$.

Lemma: Power of two choices

If m balls are placed into n bins by choosing two bins uniformly at random for each ball and placing the ball into the emptier of the two bins, then the maximum load of any bin is $m/n + \log\log n + \mathcal{O}(1)$ with high probability in n.

Algorithm for special array $A_{\alpha+1}$

1. Split $A_{\alpha+1}$ into two subarrays B and C of equal size.
2. First, try to insert in B. Upon failure insert into C (insertion to C is guaranteed to succeed with high probability)
3. B is implemented as a uniform probing table, and we give up searching through B after $\log \log n$ attempts.
4. C is implemented as a two-choice table with buckets of size $2 \log \log n$.

Probe Complexity of $A_{\alpha+1}$

Complexity of inserting into B -

1. B has size $A_{\alpha+1}/2 \geq \delta n/4$, so load factor never exceeds $1/2$.
2. Each insertion makes $\log\log n$, each of which has success probability of $1/2$.
3. Thus, expected number of probles is $\mathcal{O}(1)$
4. Probability that insertion fails after all attempts is $1/2^{\log\log n} \leq 1/\log n$.

Probe Complexity of $A_{\alpha+1}$

Complexity of inserting into C -

1. Recall, C is implemented as a two choice table with buckets of size $2 \log \log n$
2. From Lemma we have that, with high probability, no bucket in C overflows.
3. Expected time of each insertion in C is at most $o(1)$.

Algorithm

Input: $n \geq 0$
Output: $y = x^n$
$y \leftarrow 1$
$X \leftarrow x$
$N \leftarrow n$
while $N \neq 0$ **do**
  if N is even **then**
    $X \leftarrow X \times X$
    $N \leftarrow \frac{N}{2}$
  else
    $y \leftarrow y \times X$
    $N \leftarrow N - 1$
  end if
end while

List of Publications

1. Amit Roy and Jayalal Sarma. On Alternation, VC-dimension and k-fold Union of Sets. Proceedings in 11th *European Conference on Combinatorics, Graph Theory and Applications* (EUROCOMB 2021), Sep 6-10, 2021, Barcelona

Thoughts and Questions ?

Thanks