

CS121 Lab2

Lei Huang

Jan 2023

1 Introduction

In this lab, I implemented the Cuckoo hashing.

2 Hardware and software configuration

I run the benchmark on *OS: Ubuntu 20.04.4 LTS x8664*, with *Intel Xeon Gold 6342 (96t/48c) @ 3.500GHz* CPU. I ran my benchmark on NVIDIA RTX 3090.

3 Experiment

3.1 Inserion Test

See Fig 1 and 2. Using 3 hash functions is significantly better than that using 2 hash functions.

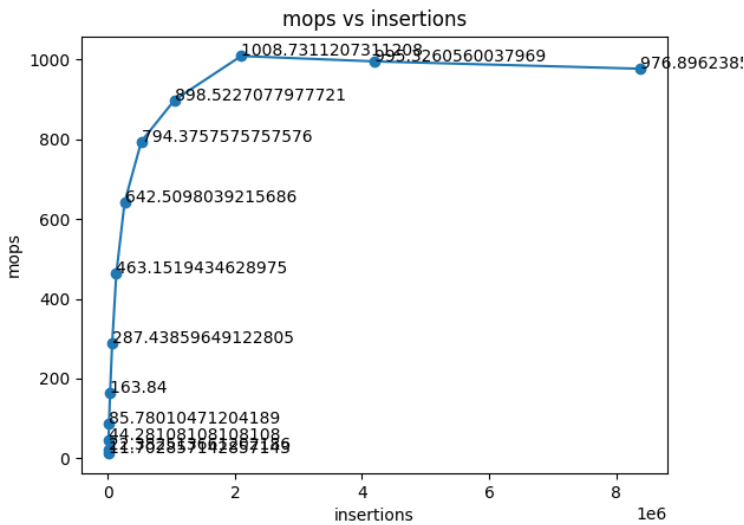


Figure 1: inserion performance $t = 2$

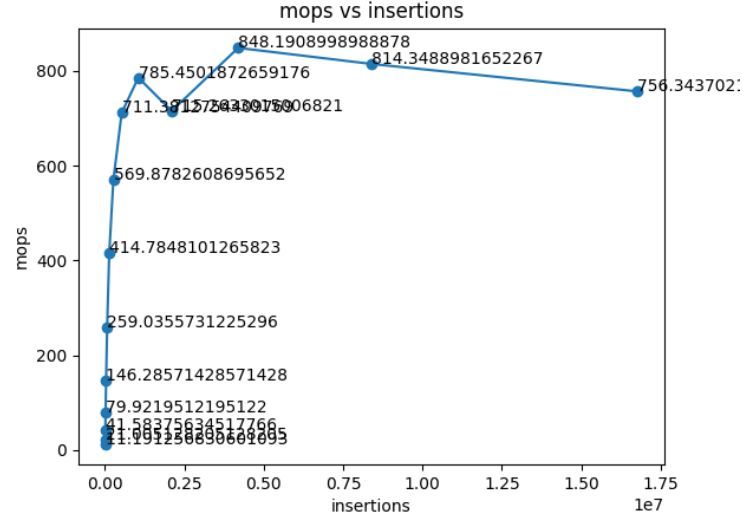


Figure 2: inserion performance $t = 3$

3.2 Lookup Test

This section evaluates the efficiency of lookup operations using 2 and 3 hash functions. It's seen that using only 2 hash functions will improve the lookup efficiency.

In Figs. 3 and 4, the performance metrics of lookup operations with 2 and 3 hash functions are presented, respectively.

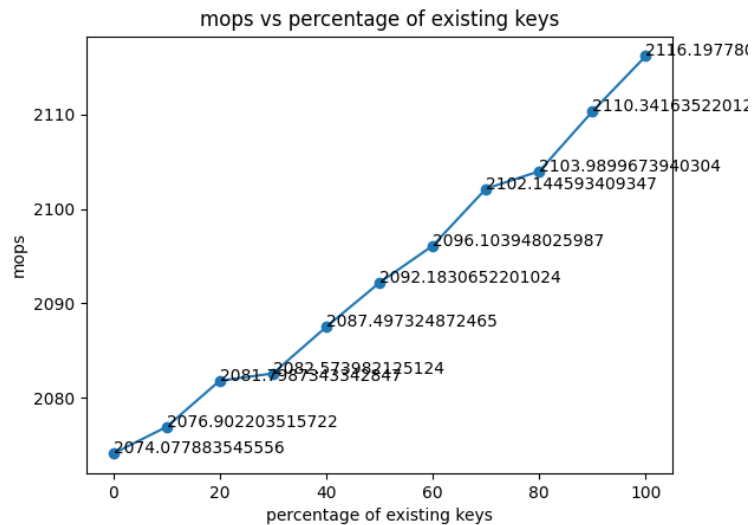


Figure 3: Lookup performance $t = 2$

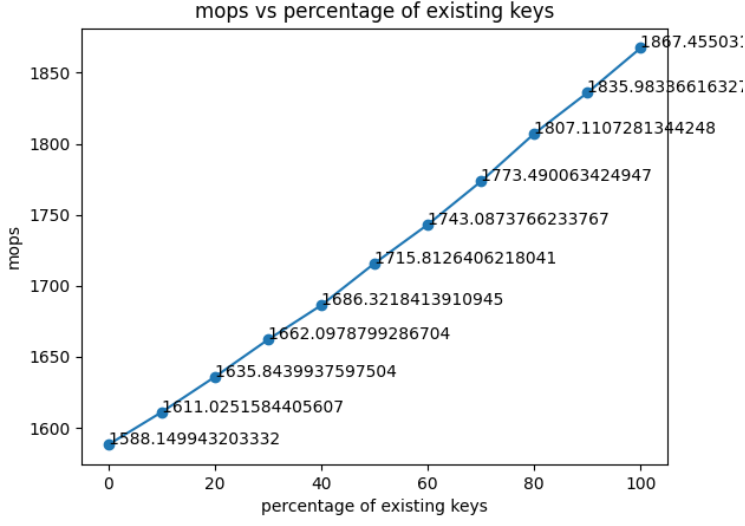


Figure 4: Lookup performance $t = 3$

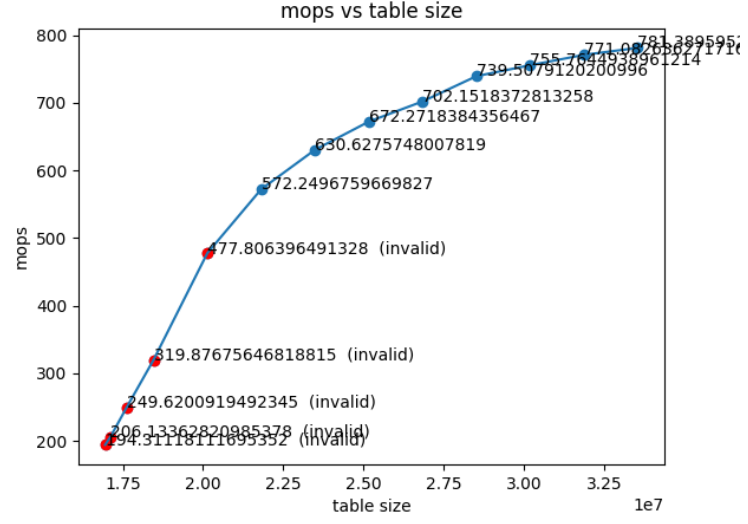


Figure 6: Capacity performance with $t = 3$

3.4 Chain Size Test

Since the construction result is invalid for $t = 2$ when the table size is small. Here we only test the $t = 3$ situation. From the graph we can recognize two major changes of the chain size: 11 and 79 where the time dropped.

3.3 Capacity Test

The Capacity Test evaluates the ability of the hash structure to handle varying sizes of table sizes. However, the $t = 2$ function Cuckoo Hashing is poor in the capacity. This results in all invalid results. The time of the invalid data points equals to the time to construct the table.

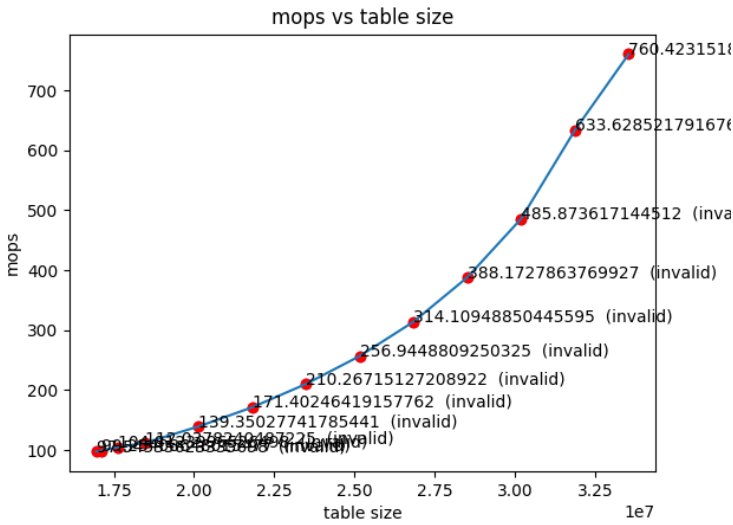


Figure 5: Capacity performance with $t = 2$

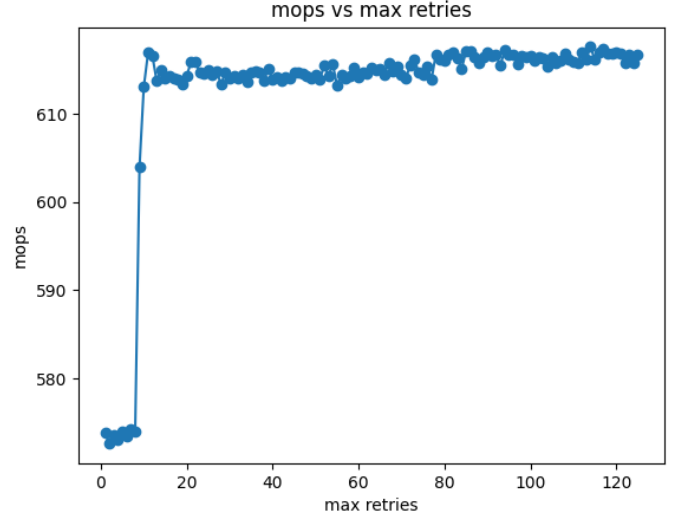


Figure 7: Construction Performance with different max chain size