

My design methodology involved increasing the table size so that there would be fewer collisions and faster results from getting the history value stored. I tested many different combinations of increasing and decreasing both the table_bit length and the history size, and the most optimal was to have the history size be 15 while the table_bit length be 30.

With a higher amount of histories being able to be saved, it becomes more accurate at predicting. Any table bit length greater than 30 causes bit errors since the int sizes are 32 bits. The design storage overhead is significantly higher due to the increase in the table bit size from 15 to 30, increasing by over 30,000 times. A design decision could have been to only slightly increase from 15, but to obtain the lowest MPKI, 30 is the best despite the storage overhead. I opted to keep the basic algorithm and just optimize the parameters to achieve the below 6.305 MPKI average. The result is an average MPKI of 5.290.