

Scheduling Algorithms

Philip Nelson

2017 April 22

Hypothesis

Different cache algorithms will produce better results in terms of Efficiency, Latency, and Response Time because they will make better or worse choices of which pages to keep in the cache. We will look at fifo, most recently used, least recently used and second chance.

Process

Different cache algorithms were implemented and tested on a simulation framework that had each algorithm choose pages to load into the cache.

Findings

Interestingly all algorithms were more efficient on a single core. As expected shortest job first was the most efficient on 4 cpus. Shortest job first also produced the lowest latency and fastest response time. Approximate shortest job first followed closely behind shorted job first.

Conclusion

It is clear looking at the data that as the cache size increased, the algorithms were more efficient. Most recently used performed significantly worse while the other three were pretty evenly matched. Oddly latency and response time suffered with increased cache sizes however second chance seemed to perform slightly better than fifo or least recently used. Most recently used was still terrible.

Data

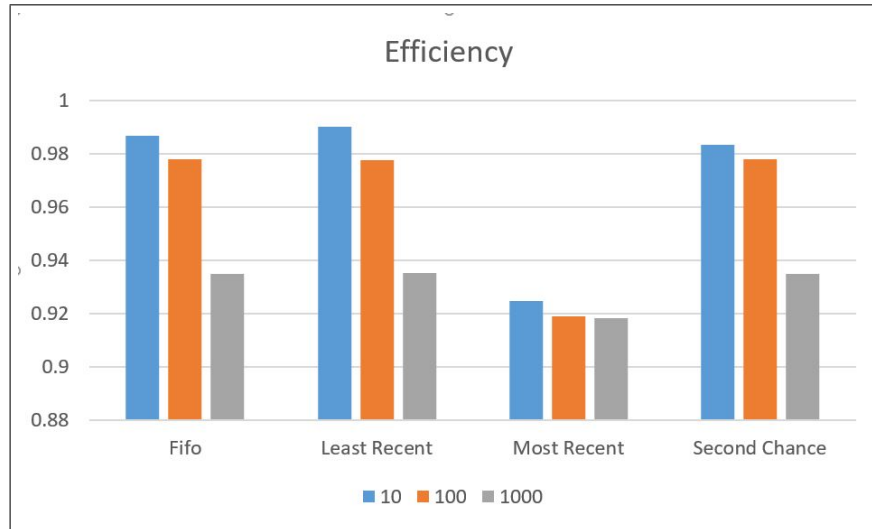


Figure 1: Efficiency

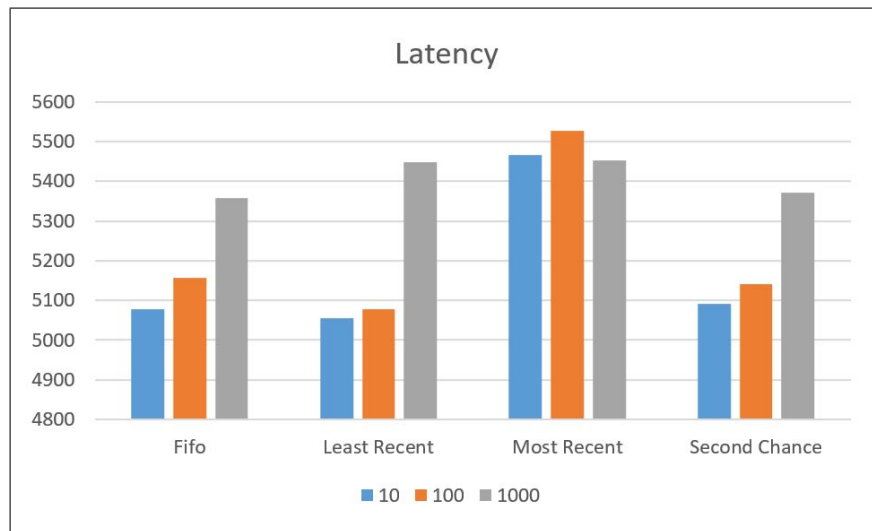


Figure 2: Latency

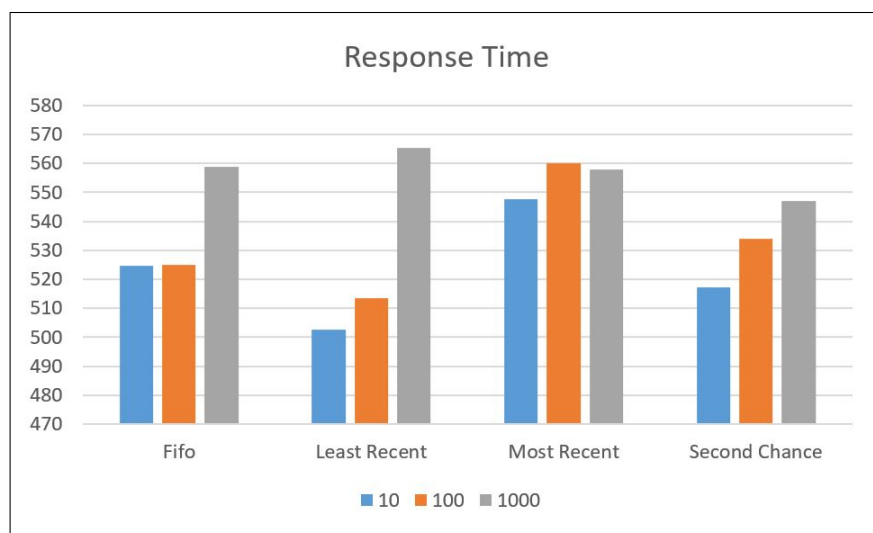


Figure 3: Response Time