

Experiment Documentation

Group members:

1. Balaji Udayakumar (balaji@bu.edu)
2. Luke Staib (ljstaib@bu.edu)
3. Duruvan Saravanan (durusarv@bu.edu)
4. Syahrial Dahler (dahler@bu.edu)

Introduction:

This experiment aims to demonstrate the benefits of caching. First, we will compare the performance of clients with a web cache with the performance of clients with no web cache. Second, we intend to compare the performance of clients using various caches running different caching algorithms.

Also, we will compare our web retrieval performance by modifying the loss rate of the connection between the clients and the server. We will track various metrics such as RTT and throughput in order to elaborate on our results.

Terminology:

- LRU
 - Shorthand for Least Recently Used. This is the caching algorithm that we have implemented in our first created cache.
- FIFO
 - Shorthand for First In, First Out. This is the caching algorithm that we have implemented in our second created cache.
- Average memory access time
 - A measurement of a cache's performance.
 - Equal to the hit time of the cache plus the miss rate of the cache times the miss penalty of the cache.
- Loss
 - Simulated loss rate for the clients accessing web objects.
- Nginx (<https://www.nginx.com/>)
 - Existing cache that uses LRU. In the second part of our experiment, we will be comparing this cache to our created FIFO and LRU caches.

Part 1: Retrieving Objects with and without Caching

Design:

Our design will consist of a client with a cache server on one GENI rack and a remote server on another GENI rack.

The purpose of the remote server is to hold web objects for the clients to download. The cache server can be disconnected and reconnected in order to test our client with a cache and without a cache.

We have configured our setup so that a client will access their cache before downloading a web object. There is no extra setup required by the client to configure the cache server.

Methodology:

1. Set-up
 - a. Configure our client, server, and cache server with their respective programs
 - Also, make sure that helper programs and files are configured as well
 - b. Clear the cache server (start with a fresh cache)

For Steps 2-4:

- Have our client run client.py
- Have our server run server.py
- Have the script run in its entirety
- At the end, the script will output average throughput, average RTT, and average memory access time for the clients over the minute

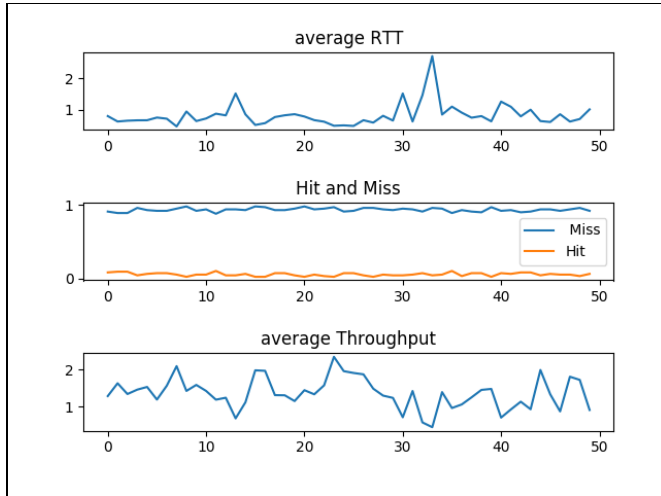
2. Test retrieving web objects without a cache and without loss

3. Test retrieving web objects without a cache and with loss

4. Repeat Steps 2-3 with our LRU cache

Analysis:

Without a cache and without loss:



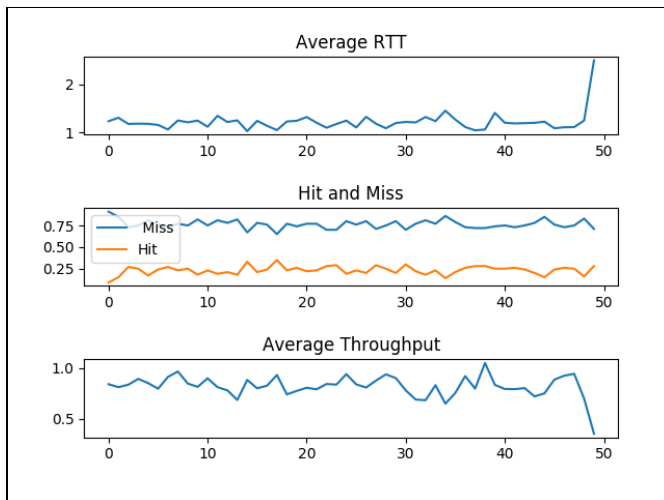
Without a cache and with loss:

(Missing)

With our LRU cache and without loss:

(Missing)

With our LRU cache and with a loss rate of 0.1:



For a client using our LRU cache, there is lower average RTT and higher average throughput than for a client not using any cache. A client using our LRU cache is able to access popular web objects much faster than a client not using our LRU cache. Therefore, conditions with the LRU cache improve upon conditions without any cache. In our case, we were able to see that the average round-trip time and throughput are improved.

Part 2: Comparing Caches:

Design:

Similarly to part 1, our design will consist of several clients with three different cache servers on one GENI rack and a remote server on another GENI rack.

However, in this part, we will switch between our three different caches (our FIFO cache, our LRU cache, and the Nginx cache) in order to compare their performance.

Methodology:

1. Set-up

1. Configure our client, server, and cache server with their respective programs
 - Also, make sure that helper programs and files are configured as well
2. Clear the cache server (start with a fresh cache)

For steps 2-5:

- Have our client run client.py
- Have our server run server.py
- Have the script run in its entirety
- At the end, the script will output average throughput, average RTT, and average memory access time for the clients over the minute

2. Test our FIFO cache without loss

3. Test our FIFO cache with loss

4. Repeat Steps 2-3 with the LRU cache

5. Repeat Steps 2-3 with the Nginx cache

Analysis:

FIFO Cache without loss:

(Missing)

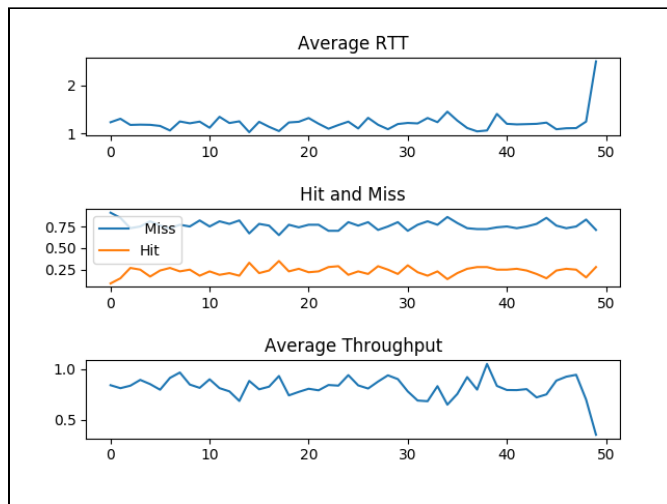
FIFO Cache with loss:

(Missing)

LRU Cache without loss:

(Missing)

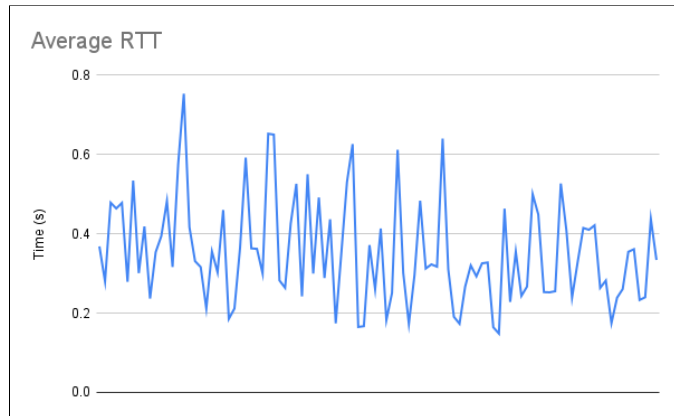
LRU Cache with a loss rate of 0.1:

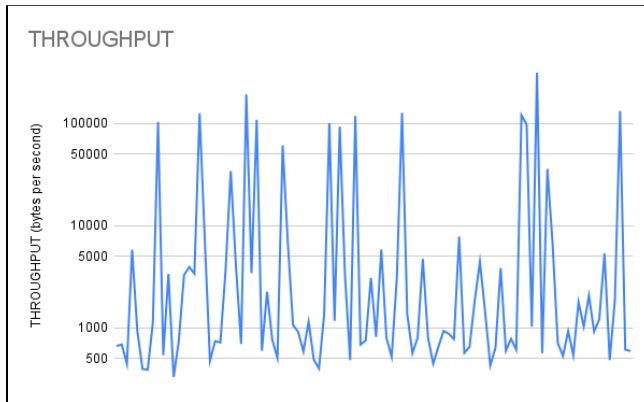
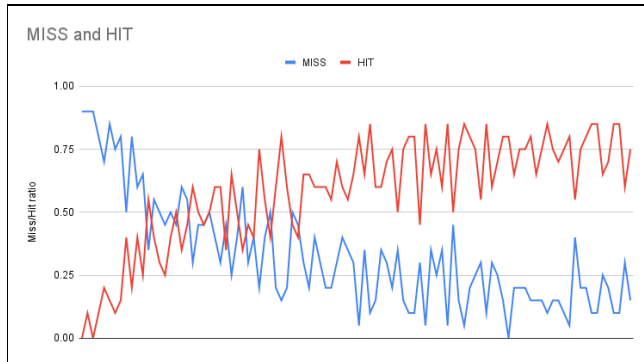


Nginx Cache without loss:

(Missing)

Nginx Cache with a loss rate of 0.1:





(Need the rest of the graphs to finish my analysis)

Conclusion:

Here we'd discuss how caching is better than not caching and show our evidence. We'd also discuss the differences between our created cache running FIFO and LRU respectively and Nginx. We would use the charts that we created to describe what we found in our experiments.

(Need to finish the analyses before I can complete this)