CS5331: Mobile Data Management and Privacy

Spring 2023

Project #3: A Pull-based Cache Invalidation - Progress Report

* Name only: \_\_\_\_\_\_\_\_\_\_\_\_\_Sayed Erfan Arefin\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* Release date: Apr 7th, 2023 (Friday)
* Due date: Apr 12th, 2023 (Wednesday), submit through the Blackboard before midnight, 11:59 PM (hard deadline)
* It should be done individually; No handwritten code or report; No email submission; No late progress report will be accepted.
* Total 10 points 1. Submit the followings through the Blackboard.
* Working source code(s) – The TA \*will not\* compile and run your code(s).
* Your source code name should be “project number” + “last name” + “first name”, e.g., “pro3\_bond\_james.c”.
* Project progress report (e.g., MS word file) - answer the following questions and be specific.

1. How much work has been done in general? Are you in the right track to complete the project within the time deadline based on your development plan? If not, why?   
   I have so far completed most of the work. I still have to debug the code to fix the cache hit ratio and number of queries to be reinitialized to 0 when removing the cache cold state.

I should be able to finish the work within the given time frame.

1. What functionalities have been done?  
   I have completed these:  
     
   a. Server has two processes. The update item process updates hot and cold items based on the requirements after T\_UPDATE period of time.   
   The second process runs and listens to the server mailbox for incoming messages. Which is basically listening for incoming requests. If a request is received, the server process replies with the corresponding message. If the process receives a MSG\_REQUEST, it returns with the corresponding data with MSG\_DATA type. If it receives a validation check with MSG\_CHECK type message, it returns either a confirm message (MSG\_CONFIRM) or data message (MSG\_DATA) based on the last updated time.

b. The client process generates a query every T\_QUERY time.   
c. Upon receiving MSG\_DATA the client process saves that on the cache and updates last access time. If the cache is full and the DATA is new to the cache, LRU is executed. If the data was present on the cache but got invalid, the process replaces the cache with the new data since the data item id are same.

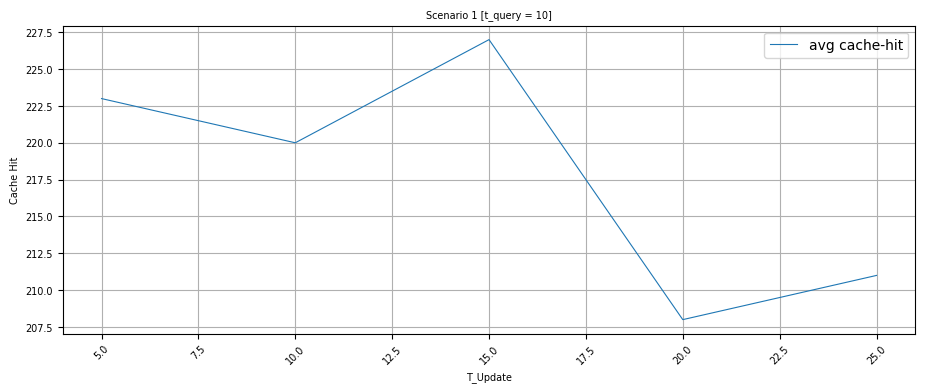
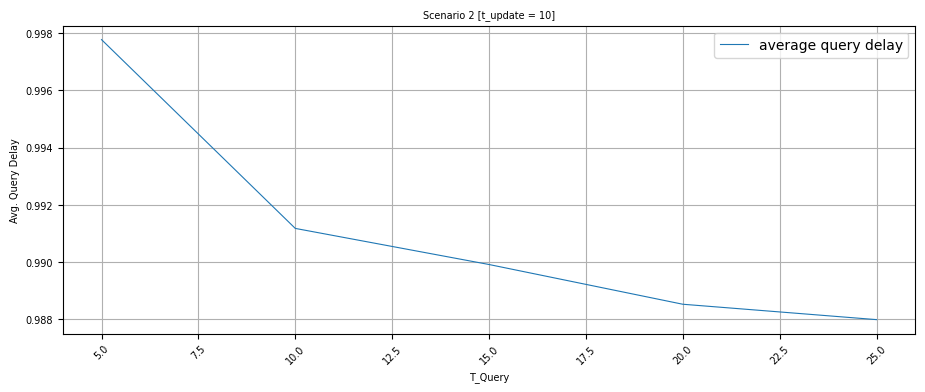
c. The cache replacement policy (LRU) works.

1. What part of work is currently on going?

Trying to debug the client process.

1. What functionalities remain to be done?

There may be a bug on the simulation which is producing incorrect results. The current plots looks like this:



I am working on the project to fix this.

1. Any implementation issues.

No implementation issues.