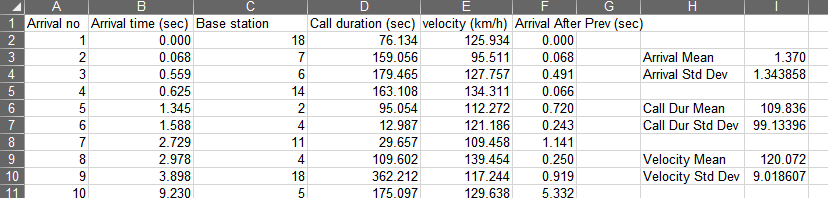
CZ4015 Simulation & Modeling

Assignment 1 – Part 2

Isaac Yang Lin Ramal

U1820798H

**Input Analysis**



Based on the data provided in “PCS\_TEST\_DETERMINISTIC”, the following data can be obtained:

* Time between arrivals
  + Obtained by subtracting the previous arrival time with the current one. The data is demonstrated in Column F
* Arrival Time
  + Mean
    - Obtained from the average of time between arrivals (Col F)
    - Calculated to be 1.370
  + Standard Deviation
    - Obtained from calculating the time between arrivals (Col F)
    - Calculated to be 1.343858
* Call Duration
  + Mean
    - Obtained from the average of call durations (Col D)
    - Calculated to be 109.836
  + Standard Deviation
    - Obtained from calculating the call durations (Col D)
    - Calculated to be 99.13396
* Velocity
  + Mean
    - Obtained from the average of velocity (Col E)
    - Calculated to be 120.072
  + Standard Deviation
    - Obtained from calculating the velocity (Col E)
    - Calculated to be 9.018607

These values are used to generate the random variables that are used to perform the simulation through the formula:

| (Standard Deviation) \* (Random Number) + Mean |

**Code**

The code was originally prepared as per the submitted pseudocode, however, throughout the implementation, various changes were made.

The simulation program consists of 3 parts.

* Main Program
* Caller class
* Constants

The Constant class contains specific information such as the values obtained through input analysis, as well as the warmup period, number of calls to simulate and number of iterations to perform.

The Caller class contains all the information and functions required for each entity.

These are:

* Speed
* Direction
* Current Cell
* Remaining Duration of call
* Time to next cell
* Next event time
* Whether the call has started
* Whether the next event is the last

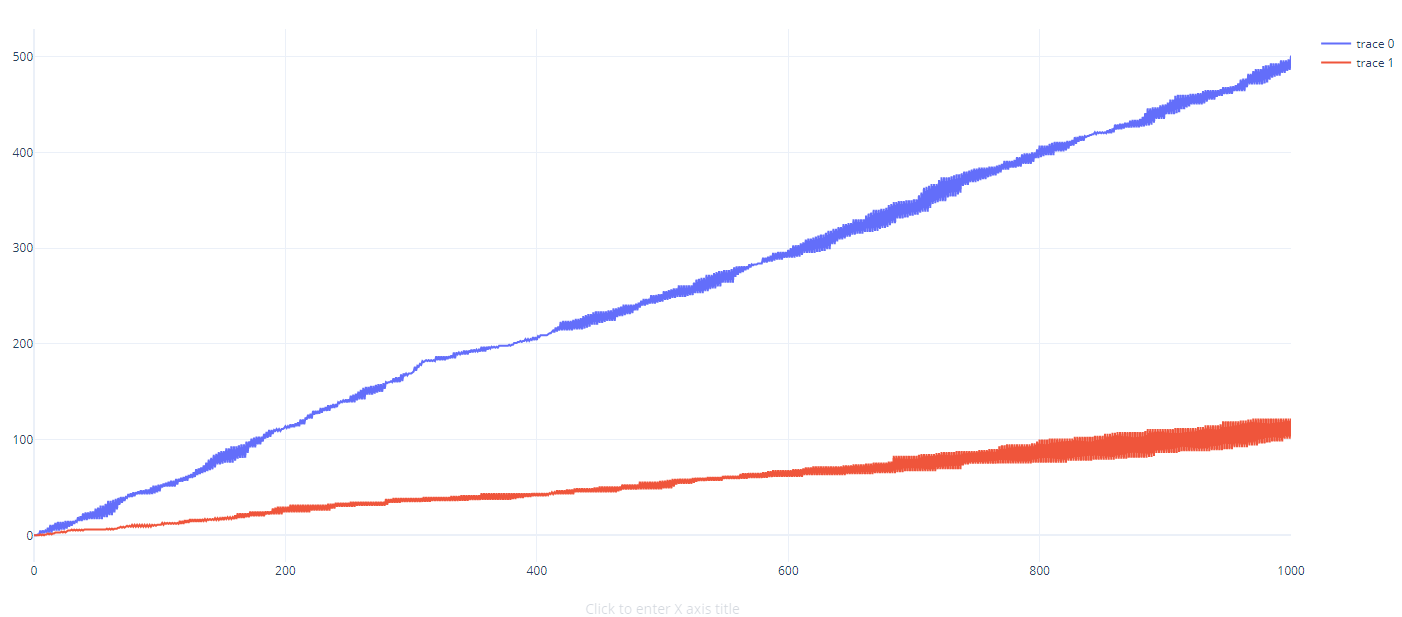
This allows the main program to simply have a list of Caller classes sorted by their next event time.

Any events that may occur is easily handled using the information available in each entity and once they have completed their final event, be it leaving the highway, completing the call, blocking or dropping, the entity is then removed.

Lastly, the main program consists of arrays to store the data obtained through the multiple iterations. Based on the specified number of iterations and simulations in the constants, the main program simulates and controls the entities and finally outputs the data once it has completed.

**Warmup Period**

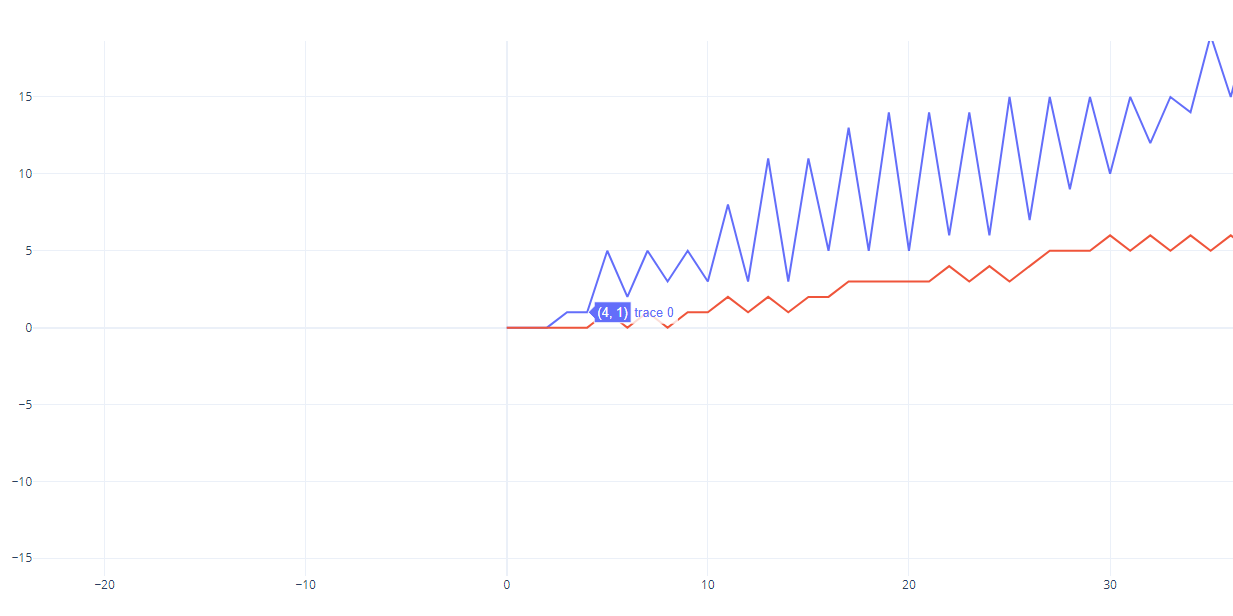
The warmup period was obtained by running a set number of simulations to plot the data and observe the timeframe in which the simulation starts to collect meaningful data.



Trace 0 (Blue): Shows the number of blocked calls per 100 calls against the total amount of calls simulated

Trace 1 (Red): Shows the number of dropped calls per 100 calls against the total amount of call simulated

This data was obtain with 100000 simulated calls and a Reserved channel value of 1

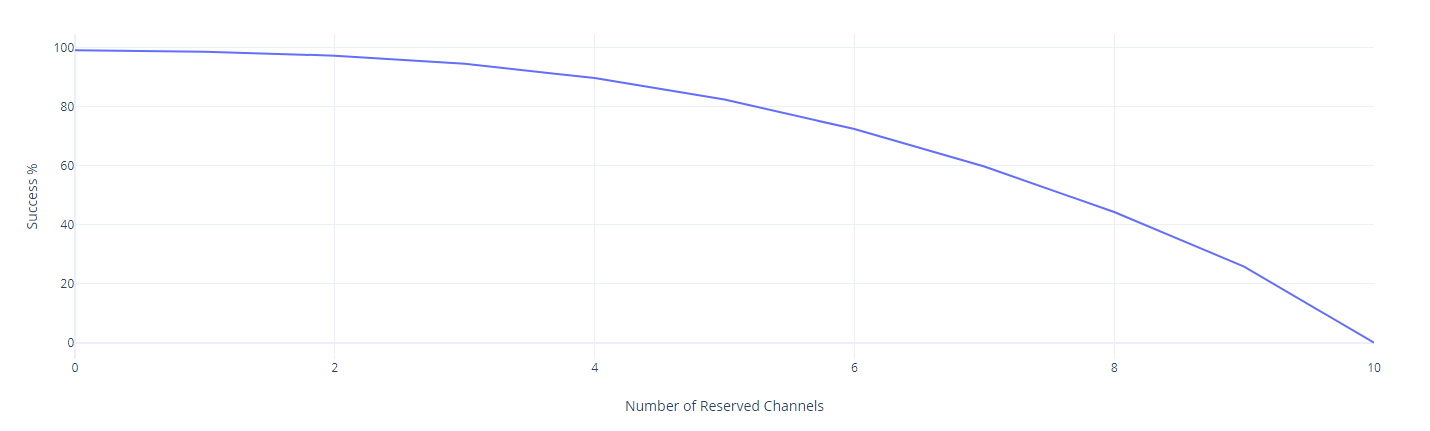


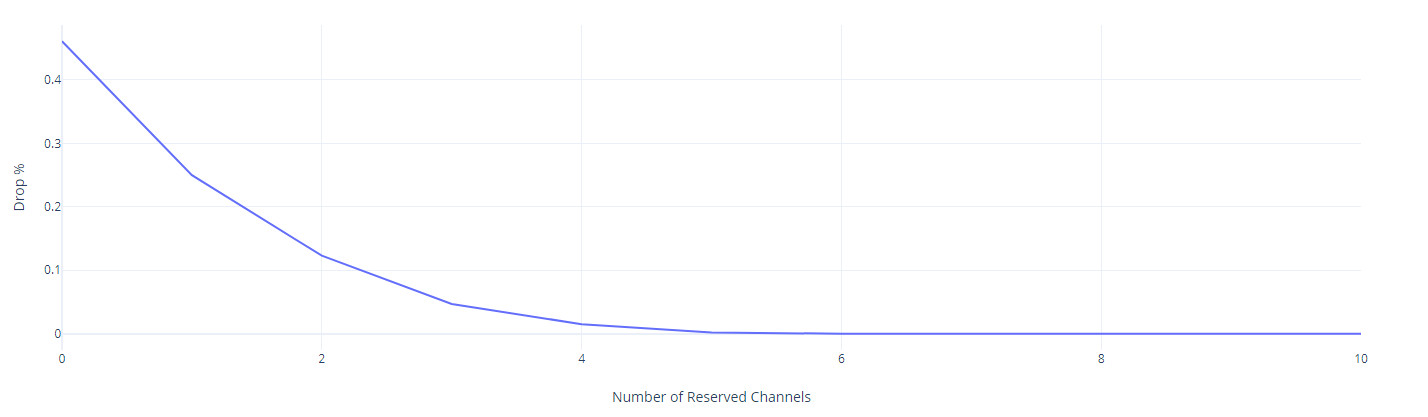
Zooming in towards the start of the graph, we can see that at approximately 400 calls, the amount of dropped and blocked calls starts to follow a trend of alternating spikes. Whereas before this amount, the dropped and blocked calls are not available.

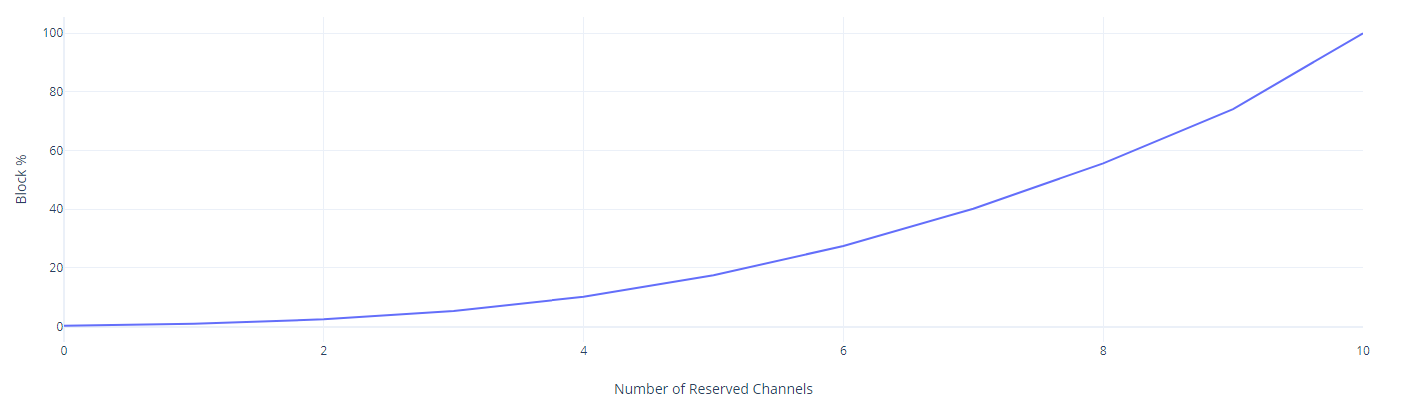
This can be explained as during the initial phase of the simulation, the number of available channels is much greater than the incoming number of calls. As such, most if not all calls are able to be assigned a channel.

As such, we can deduce that 400 is a suitable warmup period for this simulation

**Summary of Results**

****Fig 1

**** Fig 2

**** Fig 3

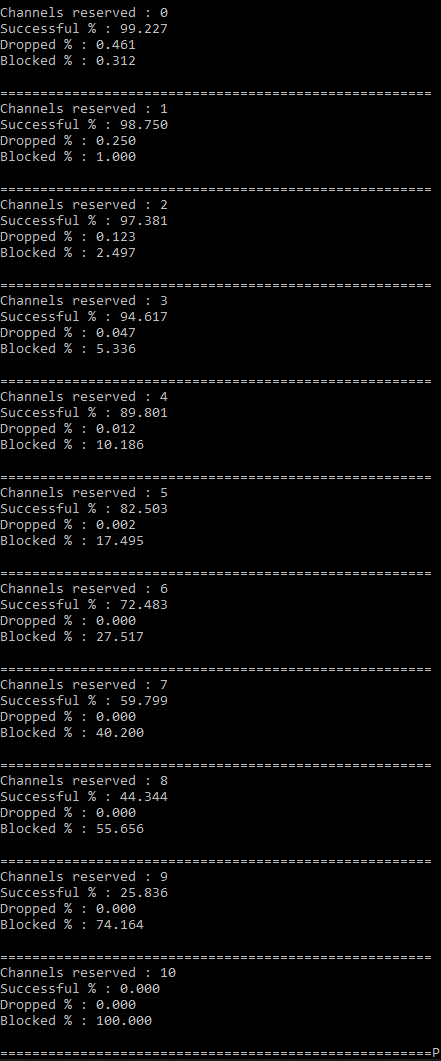
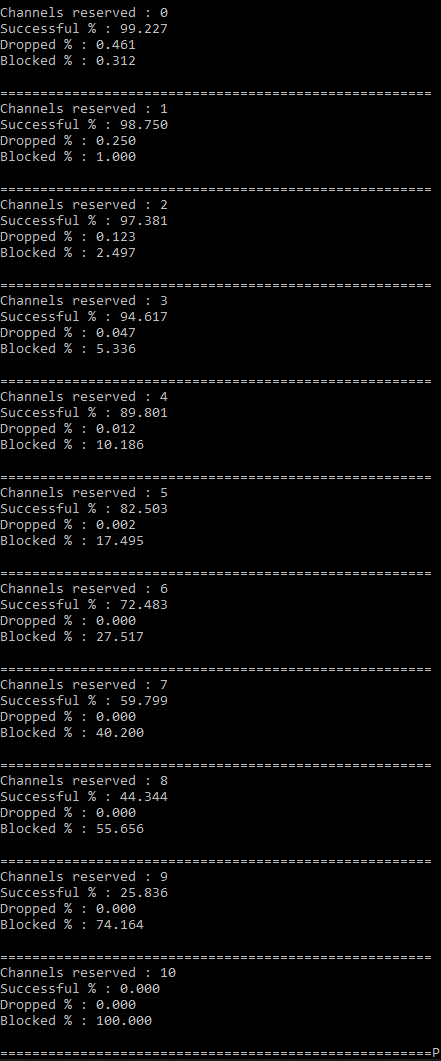


Fig 4

Fig 1 Shows the success rate of called as the amount of reserved channels increase. As indicated, as the reserved channels approach the limit of 10, the success rate drops exponentially. Eventually reaching 0 as the reserved channels reach 10.

Fig 2 Shows the percentage of calls dopped as the amount of reserved channels approach 10. The curve indicates that as the channels increase initially, the amount of dropped called get significantly reduced. At 6 reserved channels, almost no more calls are ever dropped.

Fig 3 Shows the percentage of calls blocked as the amount of reserved channels approach 10. The curve has an exponential increase and reaches 100% as the amount of reserved channels reach 10. This is as all channels have been used for transferring that no user is able to start any calls at all.

Fig 4 Displays the results of 100 Iterations of 10,000 calls. This is done for each value of reserved channel from 0 to 10.

**Conclusion & Recommendation**

Keeping in mind the original QoS Requirements of

* Blocked Calls < 2%
* Dropped Calls < 1%

We can see that only a 0 or 1 channel reservation system would fulfill the criteria specified.

The reason the larder amount of reserved channels are not satisfactory is due to the fact that as the amount of reserved channels grow, any ongoing call would be extremely easy to transfer, whereas any calls that wish to begin will have a difficult time obtaining an available channel.

To compare between 0 and 1, the amount of dropped calls in 1 reserved channel is approximately half of not having any reserved channels. This however comes at the cost of tripling the amount of blocked calls. While this still falls under the QoS requirements, the overall success rate of calls actually decreases by approximately 0.5%

Based on the requirements, there is a greater importance of lesser dropped calls as compared to blocked calls, as shown in the 2% to 1% requirement.

As such, while the current system is perfectly suitable with regards to the QoS requirements, it can be said that having 1 channel reserved may be more suitable due to the significantly lesser amount of dropped calls while taking a very small reduction in the overall success rate due to larger amounts of blocked calls

**Appendix:**

Source Code: https://github.com/oisaacylro/CZ4015\_Assignment

.