I summarize the comparison performance of SC and RL algorithm for various inter-arrival rates with different distribution.

We find out the average time response of RL algorithm and SC algorithm for same requests and compare them. We also find out Cummulative Distribution Histogram to find out the probability with which the optimal configuration is selected in different settings. We then plot the delay performance of RL after it converges under the figure name Accumulated Reqeust time.

Working with **Inter-arrival Rate with Uniform Distribution:**

We use the function **Uniform(least value, upper value)** available at rvgs.cpp file (available at standard library) to generate inter-arrival rates with uniform distribution.

When I say we have uniform inter-arrival rate of 100, it means the random values are distributed uniformly in the range 100-20 to 100+20. Number 20 is chosen randomly. Thus, when we simulate for uniform inter-arrival rate of 100, we can expect values between (80-120) only. Similarly, for inter-arrival rate of 50, we can expect inter-arrival time between (30,70).

For SC Baseline Paper, for Uniform Distribution, the optimal configuration for different arrival rates do not vary much. It follows the following table:

|  |  |
| --- | --- |
| Inter-arrival rates | Optimal Configuration |
| 20 | R7t1 (7 requests processed; 1 thread for 1 request) |
| 30 | R2t4 (2 requests processed; 4 threads for 1 request) |
| >40 | R1t8 (1 requests processed, 8 threads for 1 request) |

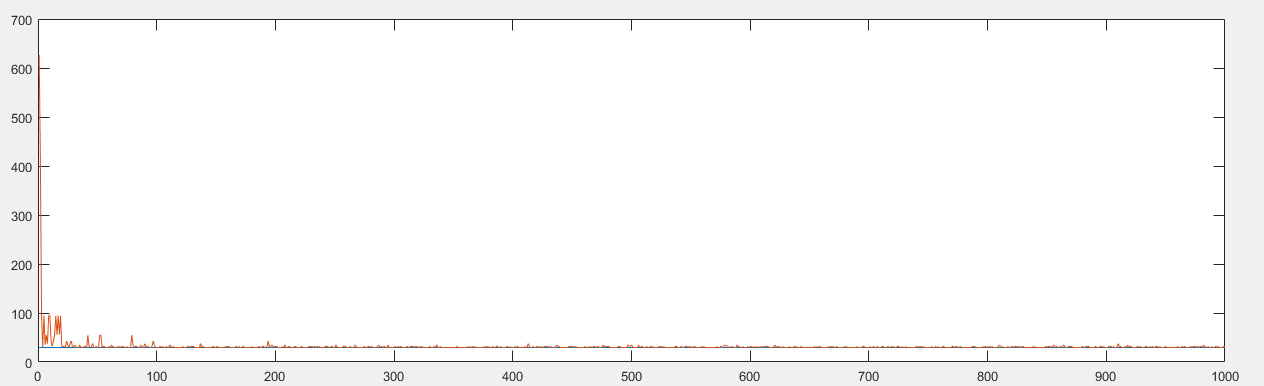


Figure 1 Average Time Response

This is the figure of comparison of average latency between RL and SC approach. Since for inter-arrival rate >40, the optimal configuration for SC is always r1t8, and is always executed with same average arrival rate. For SC, plotted in blue color, it is a straight line seen just above the zero line. For RL, which is plotted in orange color, it starts with high fluctuation but later converges to the optimal average time. However, we still can see that RL algorithm is fluctuating (which is normal).

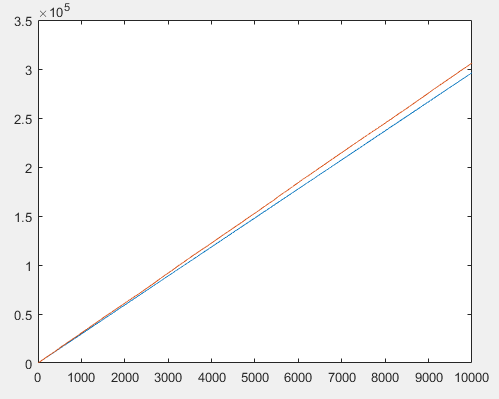
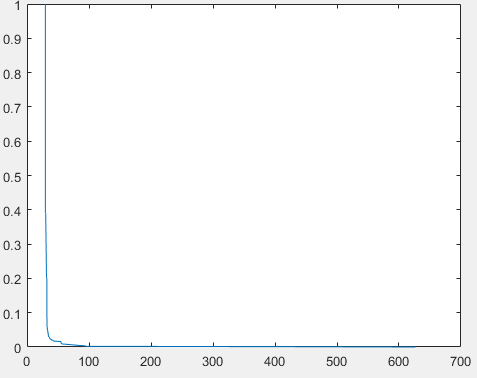
 When we plot the CDH for the same, we observe the following figure: 

Fig 2(CDH Diagram before convergence Fig 3 Accumulated Request time

From fig 2, we can see that large number of selection is made at the optimal configuration, which is good. . X axis the the time response, and y axis it the probability of running with that average response time. This plot uses average response time from the start of our simulation.

Fig 3 plots the accumulated time response for two different algorithms after convergence. We can see that even after RL converges, the overall time of execution for RL (Orange Color) is slightly greater than SC (Blue Color). This is as expected.

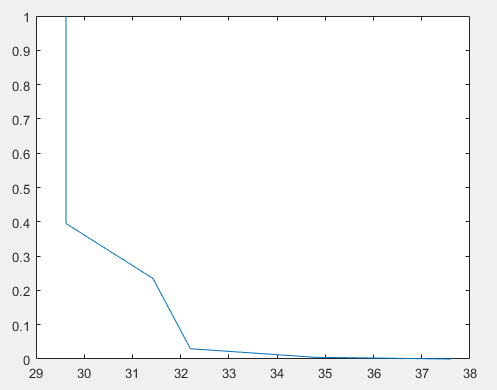


Figure 4 plots the CDH diagram for the same experiment after RL converges. X axis the time response, and y axis it the probability of running with that average response time. We can see that the optimal configuration is selected more than any other configuration combined.

Figure 4 CDH after convergence

For inter-arrival rates of 200:

The figures are preety much the same. They are corresponding figure for this arrival rate.

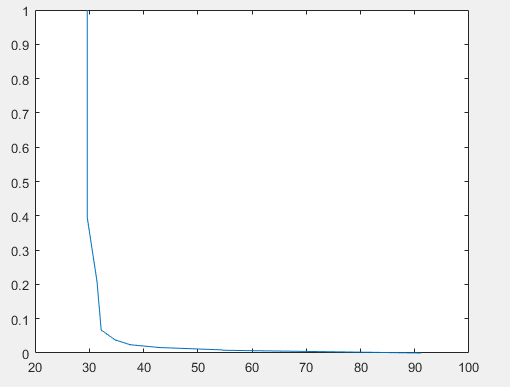
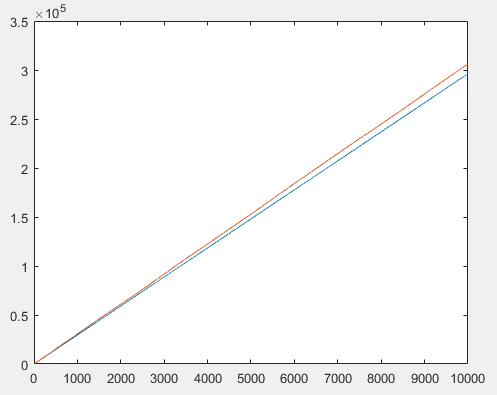


Figure 1.1 Figure 2.1

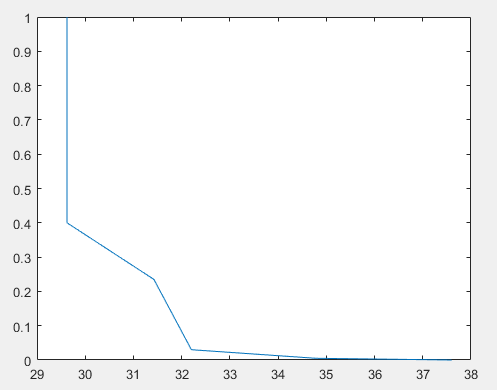


Figure 3.1

For Erlang Distribution of Inter-arrival rate:

For Erlang distribution, optimal configuration for all stable inter-arrival rates is r1t8.

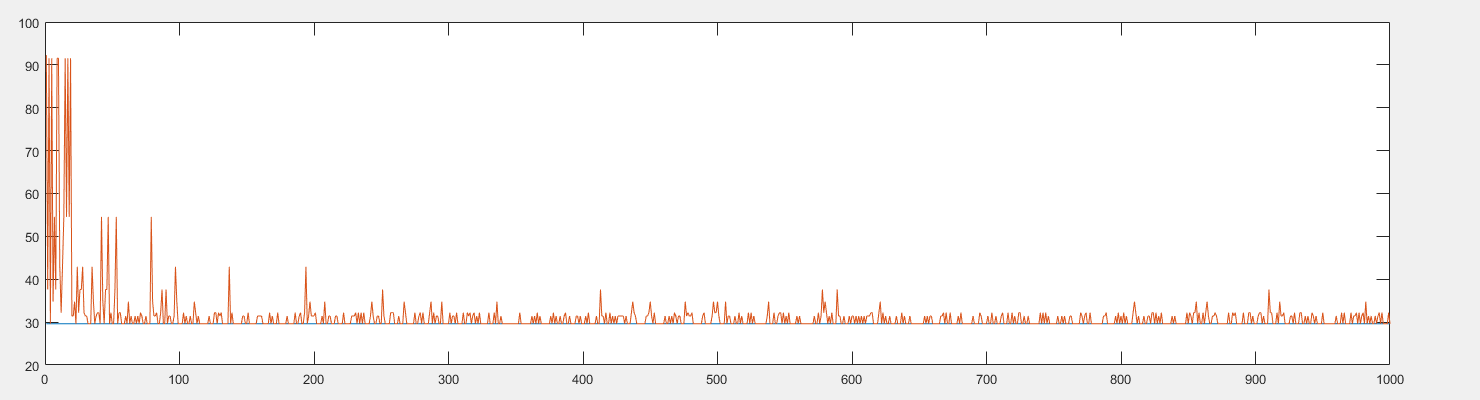


Fig 5

This figure also plot the average response of RL and SC algorithm. In the case of Erlang Distribution, we see that the the convergence of RL algorithm is not so good as compared to Uniform Distribution. This Erlang distribution has scale parameter 1.

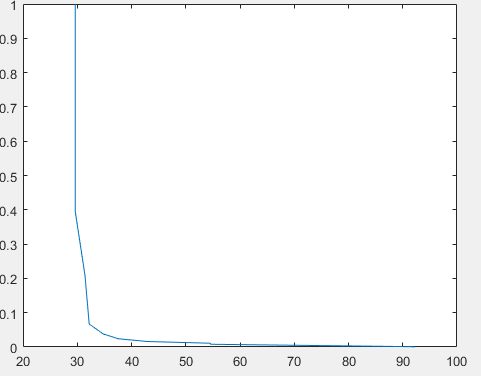
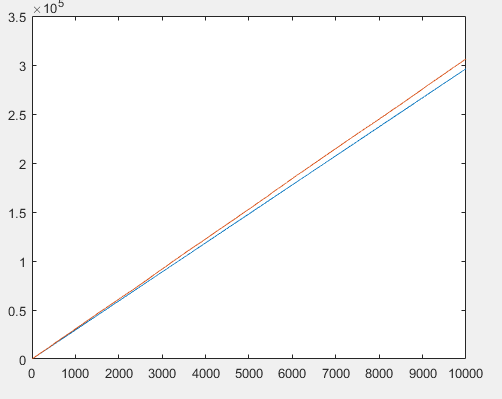
 Fig 6. CDH Plot before convergence

Fig 7. Accumulated Response Time

Figure 6 plot the cumulative distributive histogram for RL algorithm before convergence, and we see very high variability in the average response time.

We compare how the delay adds up after convergence in figure 7.

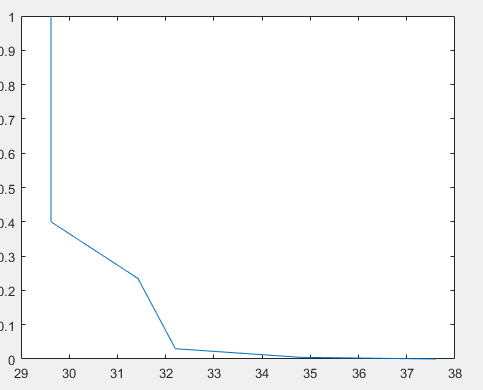


Fig 8. CDH after convergence

Here we see that after convergence, little variance is seen after convergence.

For scale parameter 2,

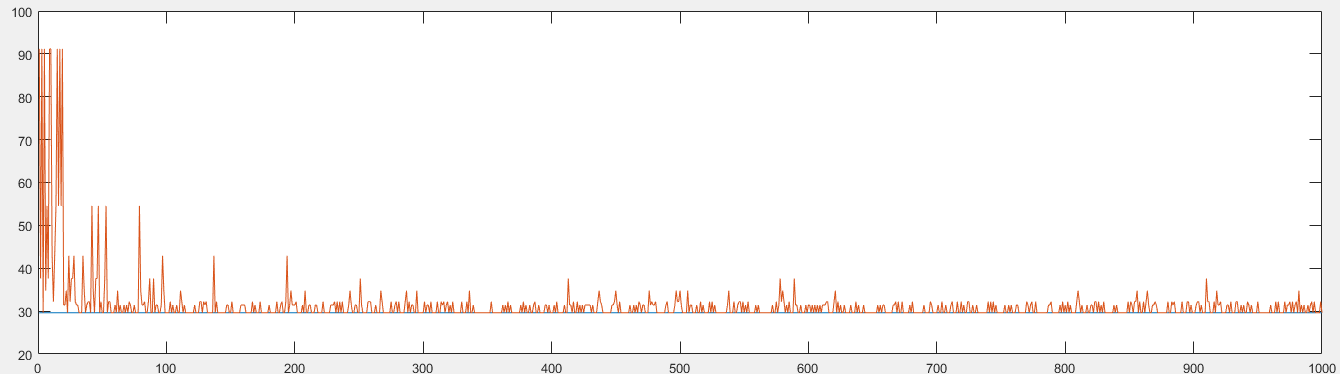
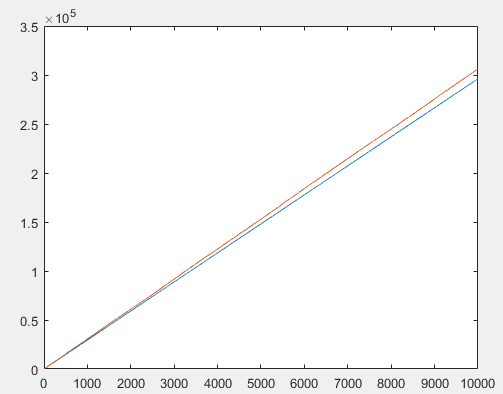
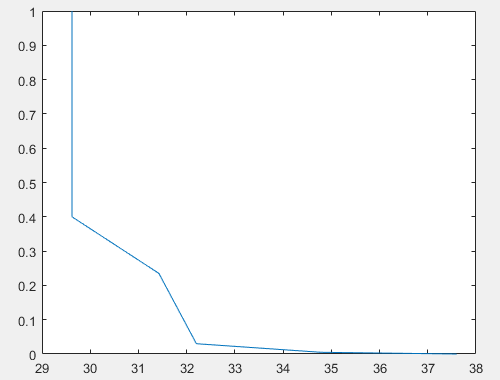


Fig 9 Average Time Response

  
 Fig 10. Accumulated Time Response Fig11. CDH after convergence

The latter two figures looks almost similar for both the above distributions because the optimal configurations are the same for both distribution.

For Normal Distribution:

For normal distribution, arrival rate of 100 means, it has mean parameter=100 and variance=1.

|  |  |
| --- | --- |
| Inter-arrival rates | Optimal Configuration |
| 20 | R4t2 (4 requests processed; 2 thread for 1 request) |
| 30 | R2t4 (2 requests processed; 4 threads for 1 request) |
| >40 | R1t8 (1 requests processed, 8 threads for 1 request) |

For stable region analysis, SC algorithm only runs at a constant optimal configuration, when the optimal time response is 29.6 units.

Comparison at Inter-Arrival Rate of 100:

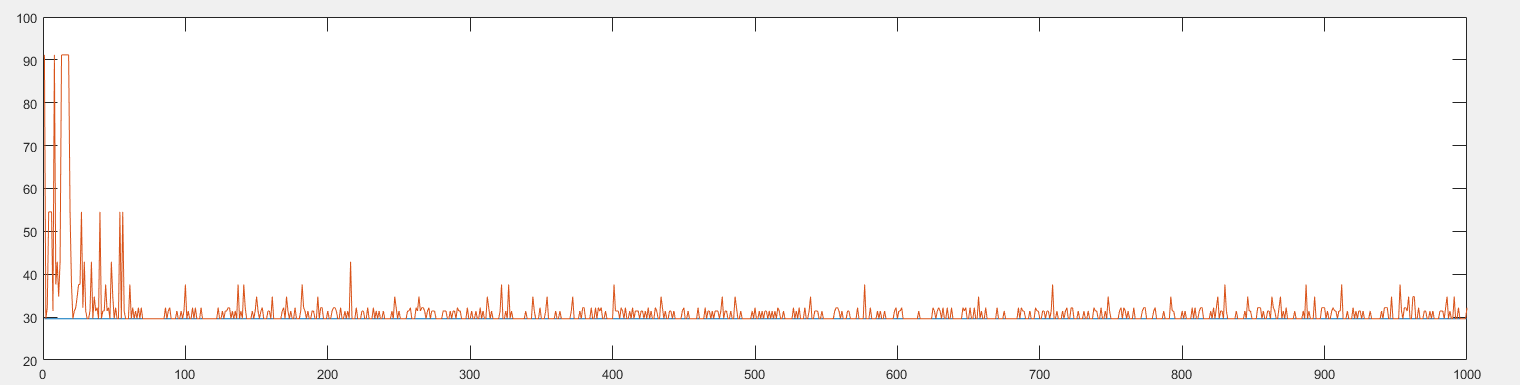


Fig12. Time Response Chart

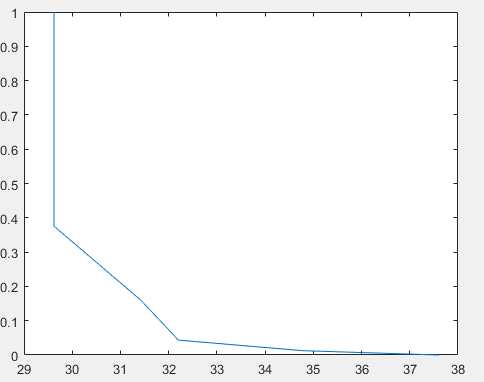
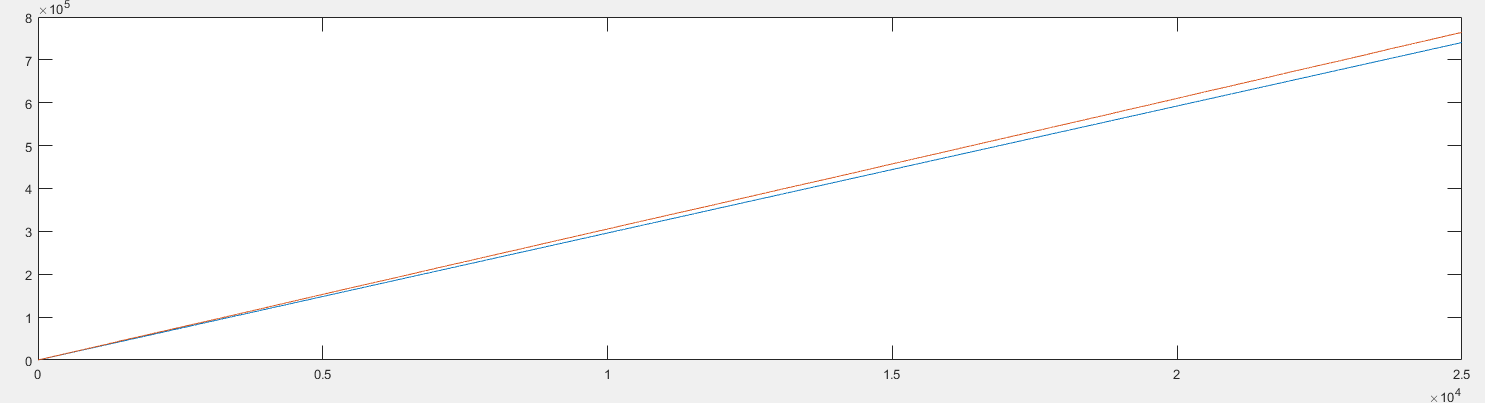


Fig: 13 Accumulated Time Response Fig 14. CDH after convergence

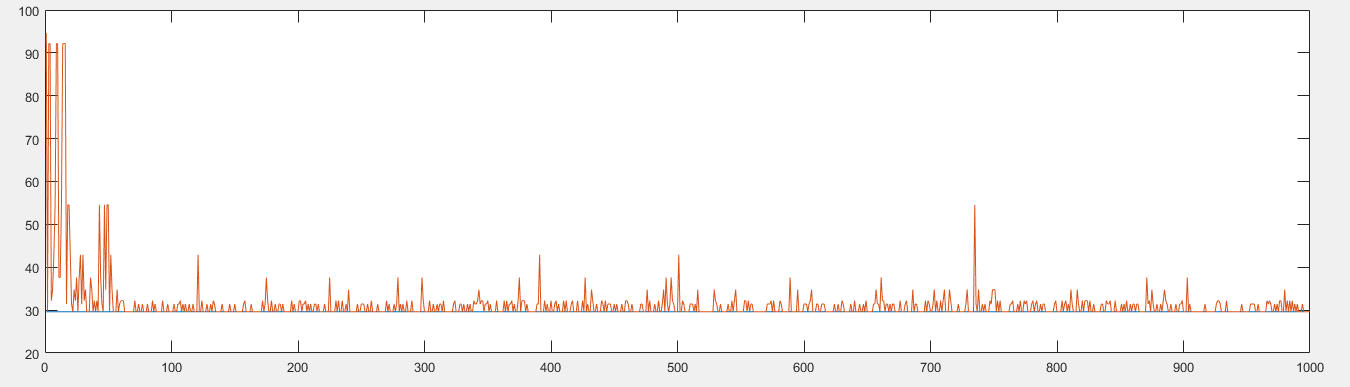
Figure 13 shows comparison between SC and RL algorithm (red being the RL).

Figure 14 shows the CDH after convergence.

ChiSquare Distribution:

SC algorithm produces the optimal configuration for different arrival rates as follow:

|  |  |
| --- | --- |
| Inter-arrival rates | Optimal Configuration |
| 20 | R4t2 (4 requests processed; 2 thread for 1 request) |
| 30 | R2t4 (2 requests processed; 4 threads for 1 request) |
| >40 | R1t8 (1 requests processed, 8 threads for 1 request) |



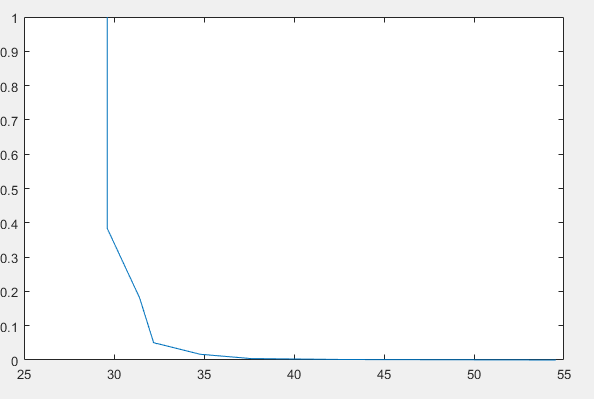
 Fig 15. Average Latencies

Fig. CDH after convergence

Observations:

Upon observation of different inter-arrival rates distribution, we can see that:

1. SC algorithm has almost same optimal configuration (except for Exponential Arrival rates), and we don’t have to change the optimal configuration. Although RL algorithm learns the optimal configuration efficiently for all different distributions, SC has better performance than RL.
2. In all of the CDH diagrams, we see that the optimal configuration is selected with maximal probability, and other configuration that are selected also are good selections.