ET2594 project

AIRPORT-Adaptive Queuing

Rajnandini Bhutada Tejaswi Dunuka Parthasarathy Singh Samanta

Goal:

To implement adaptive queuing in an Airport model.

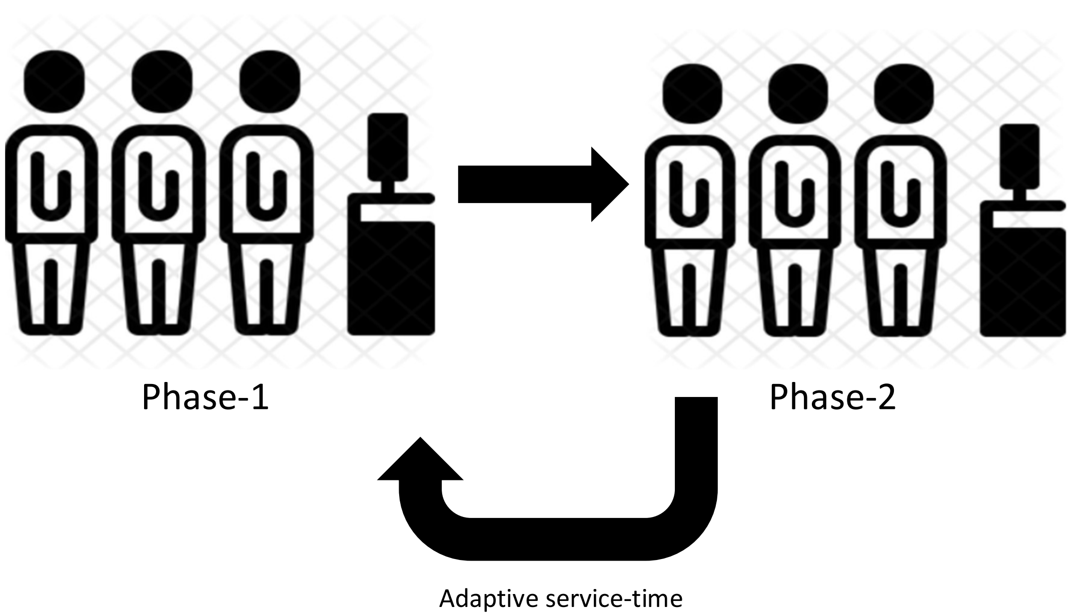
Scenario:

To implement adaptive queuing in an Airport model where the queue-1 representing the number of people arrived in check-In counter (phase-1) is modelled by M/M/1 queuing system and queue-2 represents people departing from check-In counter to Security check Counter(phase-2) which is again modelled in M/M/1 queuing system.

In phase-1 the initial service rate($\mu 1$) is considered = 10 in such a way that check-in gets completed quickly while in the phase-2 the service rate($\mu 2$) = 9 is considered and there is general delay in the service of the phase-2 due to security check.

So, queue of phase-2 is monitored continuously to control the service time of phase-1 such that no customer entering queue of phase-2 is rejected when there is a big queue in phase-2.

Modelling:



Implementation:

We have implemented the adaptive service-rate model in MATLAB. In the present Scenario we have used two M/M/1 queuing model combined one after another indicating the continuity of the system.

We have modified the code such that it produces 4 events:

event(1) = time for arrival in Phase-1
event(2) = service-time in Phase-1
event(3) = Service-time in Phase-2
event(4) = time for measurement

All the parameters for the simulation are:

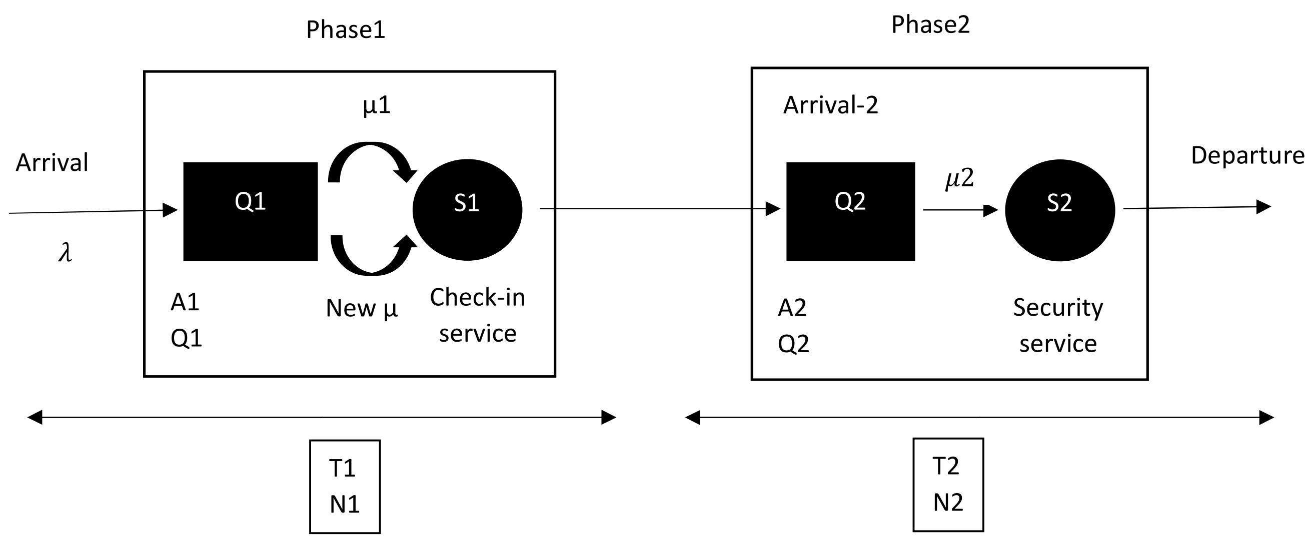
```
lambda=7      % arrival rate
mu=1          % modified service rate for PHASE 1
mu1=10        % service rate of PHASE 1
mu2=8         % service rate of PHASE 2
Threshold=7  % Threshold of QUEUE 2
endtime=100 %simulation length (seconds)
```

We are initially generating arrival for customer using ‘exprnd’ function with rate ‘lambda’. Later, the number of customers in Phase-1 are updated, queue1 is also updated and the stamp-time is noted for that customer. Later we are predicting the arrival for the next customer and simultaneously generating departure time for that customer using ‘exprnd’ function with service-rate ‘mu1’, ‘mu1’ is always selected in an adaptive where queue2 is always monitored.

```
if queue2 < Threshold
        mu = mu1;
else
        mu = mu1*(1/log(currcustomers2));
end
```

In the above Threshold are the buffer value of queue2. When queue2 exceeds the threshold value a new ‘mu’ is selected such that there is less inflow of people arriving in phase2 thereby decreasing the service-rate in phase1.

As the customer exits phase-1 it is assumed that they have arrived in phase-2, later numberOfCustomers2 are updated, queue2 is also updated and the stamp-time is noted. The departure time for customer in queue2 is also generated using exprnd' function with service-rate 'mu2'. Hence, when the customer departs from phase-2 it is assumed they have left the system.



Result:

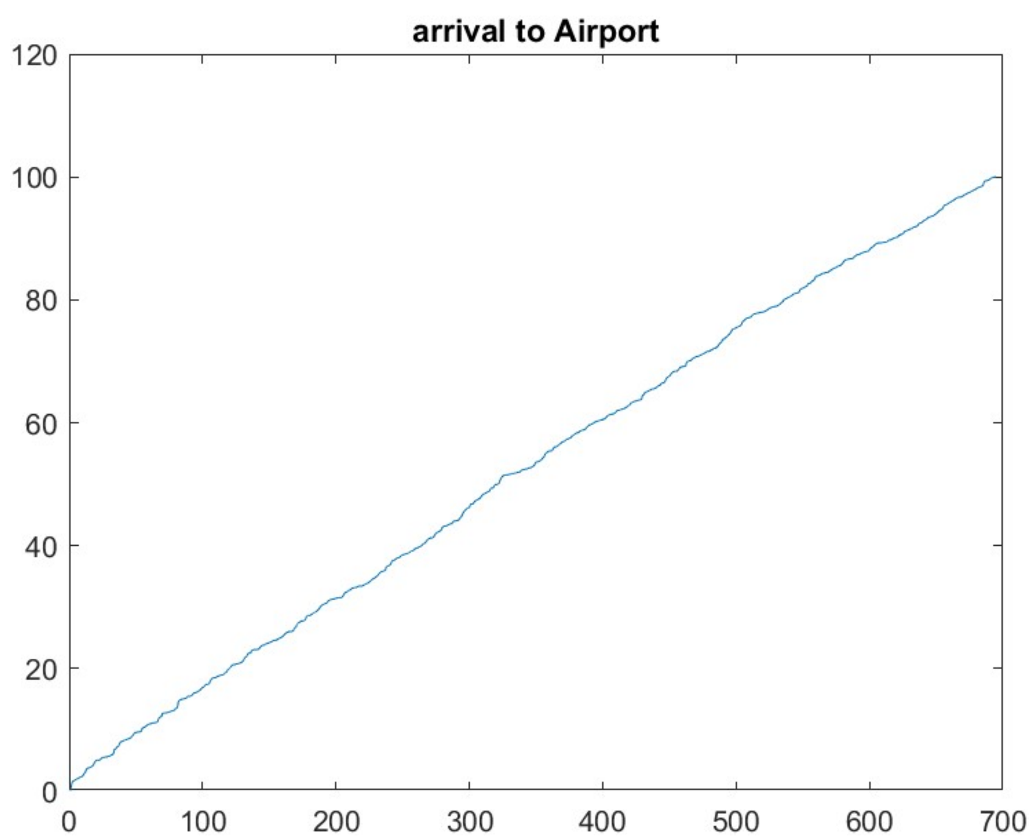


Figure 1

Figure 1 represents the arrival of the customers with mean inter arrival time of 'lambda' i.e., 7 to phase-1 for the simulation time of 100.

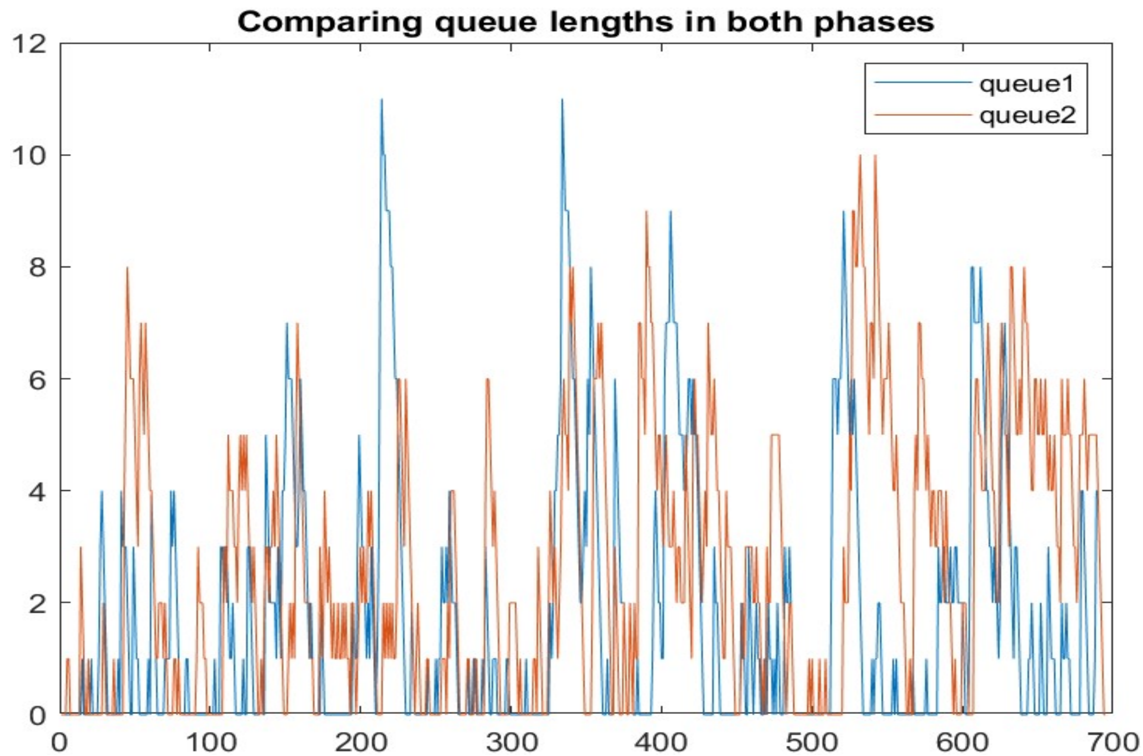


Figure 2

Figure 2 represents the queue of both phases, when queue2 of phase-2 exceeds the minimum queue length then the service rate of phase-1 is decreased and service-time increases, such that people spend more time in phase-1 for service.

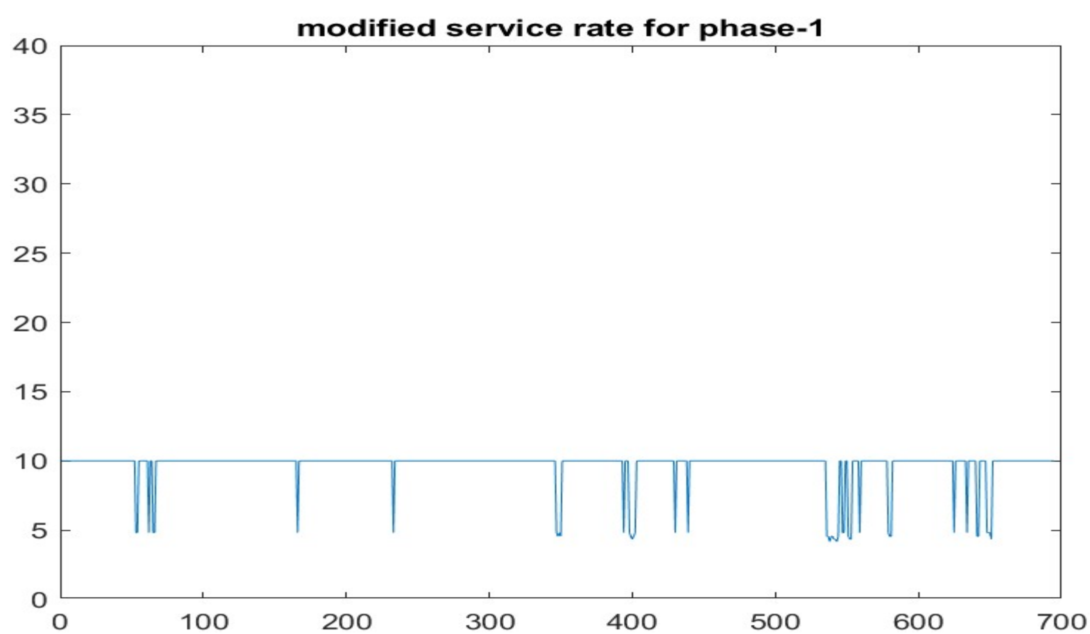


Figure 3

Figure 3 represents the change in the service rate of phase-1 with respect to queue2 as seen in the Figure 2.

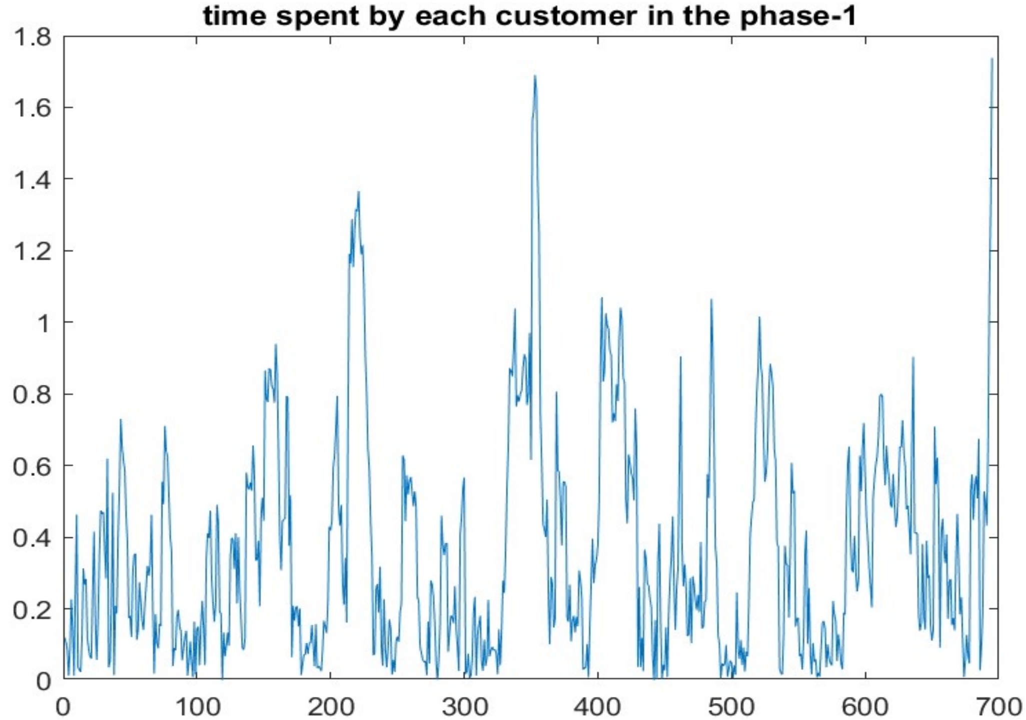


Figure 4

Figure 4 represents the sojourn time spent by the customers in phase-1 and we can see that the average time spent by customers in phase-1 increases as the service rate is decreased and by comparing the Figure 3 we can relate with these fluctuations.

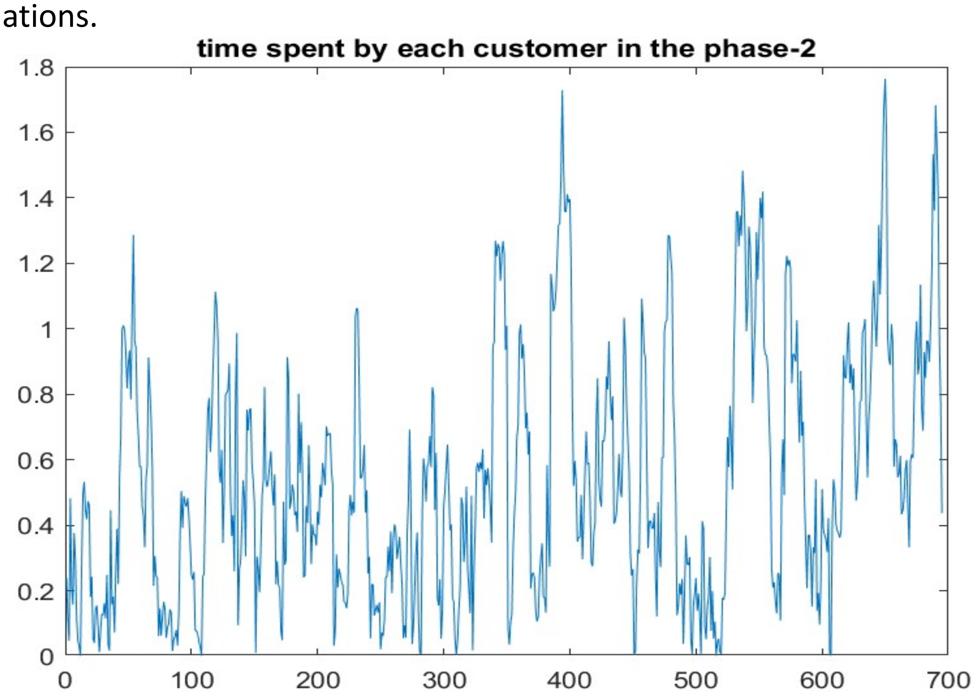


Figure 5

Figure 5 represents the sojourn time spent by the customer in phase-2, where the service rate is always same.

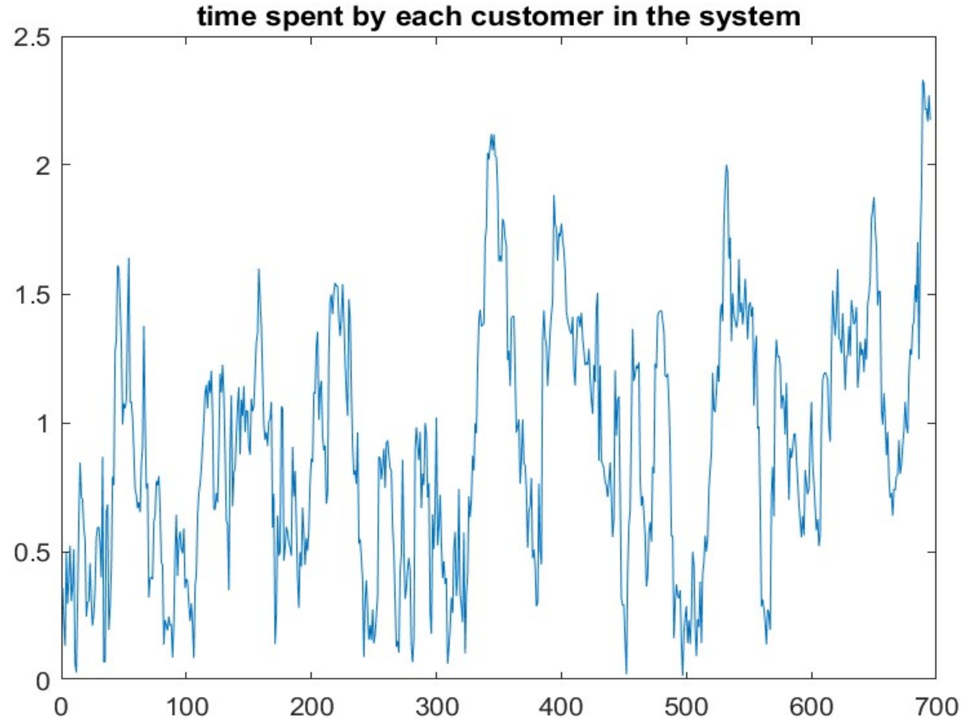


Figure 6

Figure 6 represents the total sojourn time spent by the customer in the system i.e., both in phase-1 and phase-2.

Conclusion:

Hence, from the simulated results we can relate to a real-life scenario, where more number of customers are in queue and are waiting for security check at an airport. Hence, this can be tackled by implementing the adaptive queuing for different service counters present in the airport.