M/M/S Queueing Model

This hypothetical example describes a simple M/M/S queueing model in which we adjust queue length and number of servers to handle customers during high-traffic periods. [Note: if you are not familiar with Kendall Notation for queueing models, then before continuing you should read our introductory queueing optimization page.]

The initial model settings are intended to handle modest off-peak traffic. But then we will adjust the parameters to try to handle peak traffic loads. Here are the starting values:

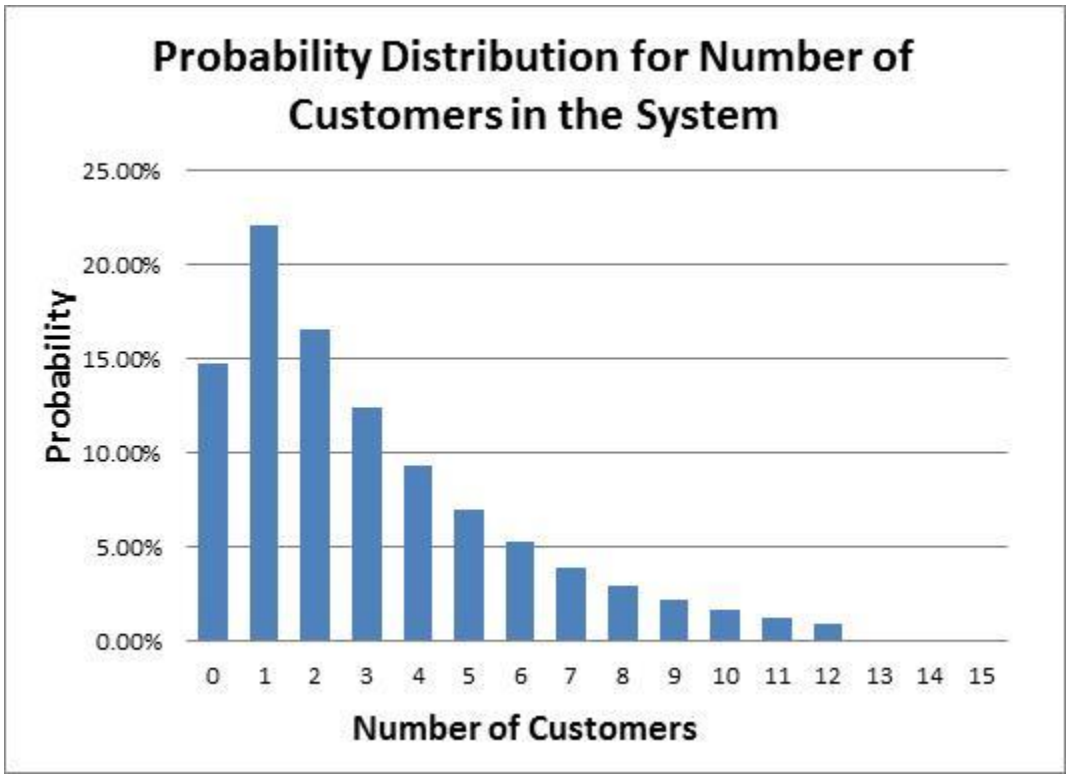
- Arrival rate: 30 customers per hour
- Average service time: 3 minutes per customer
- Number of servers: 2
- Service rate: 20 customers per hour
- Queue length: 10

Note: This model could apply to many qualitatively different types of situation, including ones where there is no actual spatial queue where customers stand. It could, for example, apply to a telephone help desk or inbound telemarketing call center just as well as a physical setting such as a retail store. In those former cases, the time on hold on the phone would represent the time in queue; and the queue length would be the number of calls that the system will accept and put on hold before giving a busy signal on the caller's phone or playing a recorded message asking the caller to hang up and try again later. These starting values give us the following results:

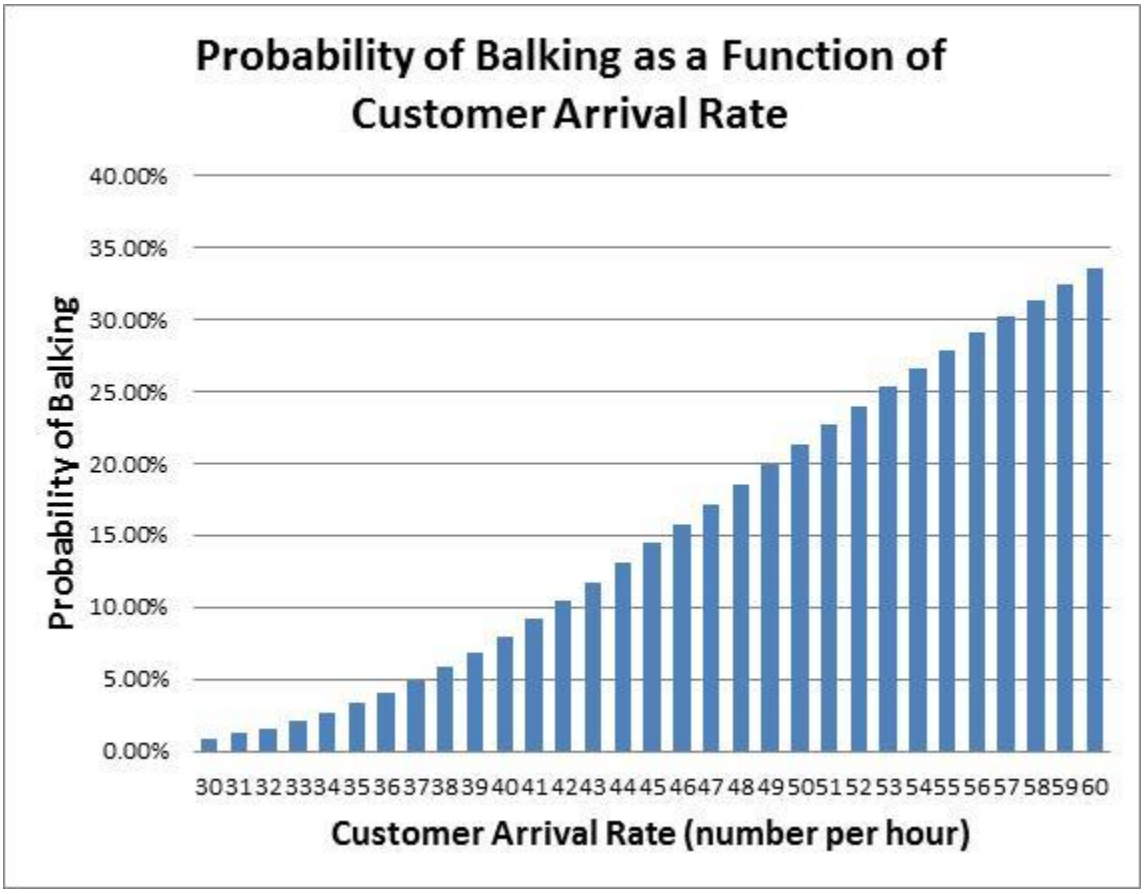
Measure	Value	Time Unit
Arrival rate	30	per hour
Service rate	20	per hour
Servers	2	
Queue capacity	10	
Utilization	74%	
Traffic Intensity	1.50	
Average number of customers in queue	1.59	
Average number of customers in system	3.08	
Average time in queue	3.21	minutes
Average time in system	6.21	minutes
Probability of having to wait	63.29%	
Probability of full system	0.93%	

As the table indicates, under these model conditions the system is able to handle the traffic quite well, and is utilizing 74% of capacity. Traffic intensity shows that the arrival rate of 30 customers per hour is 1.5 times the service rate of 20 customers per hour. On average, at any given time there will be about three customers in the system and about 1.6 customers waiting in the queue. Although just over 60% of customers will have to wait in the queue, wait times are relatively brief. There is less than a 1% chance that at any given moment the system will be full and someone will balk, or refuse to wait in the queue, and thus not enter the system at all.

The following graph shows the probability distribution for number of customers in the system:



And here is the graph of the expected balking rate under various traffic conditions ranging from 30 to 60 customers per hour, assuming only two servers on duty and a queue length of 10:



As we saw in the previous table, the current balking probability (probability of a full system) is less than 1%, so things are moving smoothly. But if traffic picks up, then more customers will balk. We would like to make sure that we have sufficient capacity to keep the balking rate relatively low, thus ensuring an acceptable level of customer satisfaction.

Now let's look at what happens when traffic starts to pick up, and we have an average of 45 customers per hour arriving at the system:

Measure	Value	Time Unit
Arrival rate	45	per hour
Service rate	20	per hour
Servers	2	
Queue capacity	10	
Utilization	96%	
Traffic Intensity	2.25	
Average number of customers in queue	5.80	
Average number of customers in system	7.72	
Average time in queue	9.03	minutes
Average time in system	12.03	minutes
Probability of having to wait	94.30%	
Probability of full system	14.43%	

Now we see that the system is definitely being taxed. We are at 96% of capacity; traffic is arriving at a rate that is 2.25 times the service rate; there are an average of about 8 customers in the system and about 6 customers waiting in the queue. So about 94% of customers are forced to wait, and the balk rate has risen to more than 14%, which is unacceptable. We need to make an adjustment to the system.

First let's look at what happens if we increase the queue length from 10 to 15 to encourage more people to wait in line instead of leaving:

Measure	Value	Time Unit
Arrival rate	45	per hour
Service rate	20	per hour
Servers	2	
Queue capacity	15	
Utilization	98%	
Traffic Intensity	2.25	
Average number of customers in queue	9.59	
Average number of customers in system	11.55	
Average time in queue	14.65	minutes
Average time in system	17.65	minutes
Probability of having to wait	97.21%	
Probability of full system	12.74%	

This hasn't helped much: the balk rate fell only slightly, from 14.43% to 12.74%; and now the system is 98% full, and wait times have increased dramatically. We've run out of physical space, so we can't increase the queue length beyond 15; but even if we could increase it more, that wouldn't really solve our problem. So it's time to add another server.

Here's what happens when we increase the number of servers from two to three under the current traffic conditions:

Measure	Value	Time Unit
Arrival rate	45	per hour
Service rate	20	per hour
Servers	3	
Queue capacity	15	
Utilization	75%	
Traffic Intensity	2.25	
Average number of customers in queue	1.60	
Average number of customers in system	3.85	
Average time in queue	2.14	minutes
Average time in system	5.14	minutes
Probability of having to wait	56.53%	
Probability of full system	0.19%	

Capacity utilization has now dropped back down to 75%, and the balk rate is back down to a very low level. We could even reduce the queue length back down to 10. Here is the result of reducing the queue length when we also have three servers:

Measure	Value	Time Unit
Arrival rate	45	per hour
Service rate	20	per hour
Servers	3	
Queue capacity	10	
Utilization	74%	
Traffic Intensity	2.25	
Average number of customers in queue	1.40	
Average number of customers in system	3.63	
Average time in queue	1.88	minutes
Average time in system	4.88	minutes
Probability of having to wait	55.71%	
Probability of full system	0.82%	

Capacity utilization remains good, and the balk rate is still under 1%, which is fine. So it is clear that maximizing queue length doesn't matter nearly as much as adding a server. In terms of sensitivity analysis, this means that the system is much more sensitive to a 50%change in the number of servers than to a 50% change in queue length.

But we're not out of the woods yet. Traffic continues to increase until it reaches a peak of 70 customers per hour. With three servers and a queue length of 10, let's see what happens:

Measure	Value	Time Unit
Arrival rate	70	per hour
Service rate	20	per hour
Servers	3	
Queue capacity	10	
Utilization	97%	
Traffic Intensity	3.50	
Average number of customers in queue	6.13	
Average number of customers in system	9.05	
Average time in queue	6.30	minutes
Average time in system	9.30	minutes
Probability of having to wait	94.73%	
Probability of full system	16.57%	

Now we're back up to 97% capacity utilization, wait times have increased, and the balk rate is now a whopping 16.57%. So it's time to add a fourth server, which has the following result:

Measure	Value	Time Unit
Arrival rate	70	per hour
Service rate	20	per hour
Servers	4	
Queue capacity	10	
Utilization	85%	
Traffic Intensity	3.50	
Average number of customers in queue	2.54	
Average number of customers in system	5.94	
Average time in queue	2.24	minutes
Average time in system	5.24	minutes
Probability of having to wait	68.42%	
Probability of full system	2.92%	

Utilization has dropped to 85% and wait times are acceptable. The probability of balking is just under 3%, which is pretty good. We can increase the queue size back up to its maximum of 15 to try to reduce the balk rate a bit more. Here is the result:

Measure	Value	Time Unit
Arrival rate	70	per hour
Service rate	20	per hour
Servers	4	
Queue capacity	15	
Utilization	86%	
Traffic Intensity	3.50	
Average number of customers in queue	3.46	
Average number of customers in system	6.92	
Average time in queue	3.01	minutes
Average time in system	6.01	minutes
Probability of having to wait	71.28%	
Probability of full system	1.36%	

This works out well: utilization stays about the same, wait times are reasonable, and the balk rate is now down to 1.36%. By keeping track of historical data over a sufficiently long period of time, we can continue the modeling process so that in the future we will be able to anticipate

traffic intensity at various times of day and make the appropriate adjustments before the balk rate gets out of hand.