

# BA252 Exercise 4--Queuing and Simulation

The book makes this topic more complicated than it needs to be for our purposes. the topic can be simplified substantially by using tables rather than calculations, and even more by using POM-Win.

## Characteristics of Queues

To simplify, we will only consider

One-step process

FIFO discipline

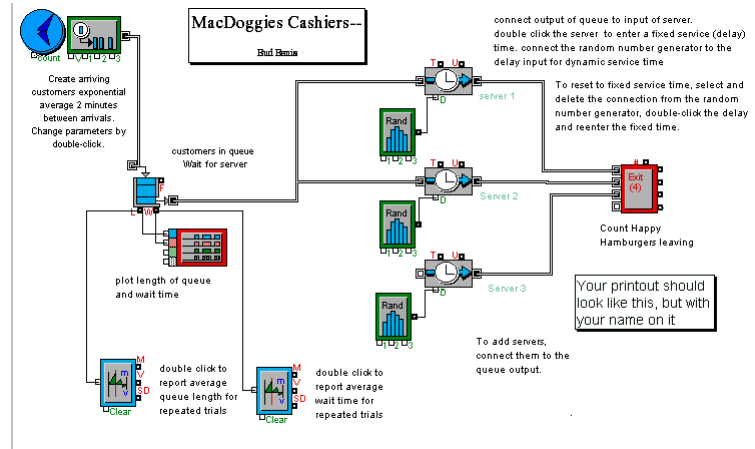
Poisson arrival distribution

Exponential Service time distribution

Infinite source of customers

## Criteria /goals for a good queue design

- reduced waiting time
- reduced queue length
- less customer time in the system



## Application of EXTEND LT to a queuing problem

After playing with this a little, I'm not sure about the absolute results from the EXTEND queuing model. The one I've set up (Macdog.mox) has initially an arrival rate of 30 per hour for an average arrival interval of 2 minutes and a service rate of 30 per hour with one server, constant service (delay) time of 2 minutes. Theoretically, this should be an exploding queue, but that isn't what happens. This makes me think I'm overlooking something. Nevertheless, we can use this as an experimental model and may be able to draw some conclusions from the relative results.

## Tutorials

Install EXTEND-LT from the CD ROM that came with the book. in the start menu, you can open *getting started* and watch at least the tutorials on discrete event models and animation.

## Experiment 1: Constant vs. variable service time

Download the Macdog.mox model from the website by shift click or right click on the link. Start EXTEND and open the Macdog.mox model. The model is initially set up to give an arrival rate of 30 per hour. check this by double-clicking the generator and observe that it has an exponential distribution with an arrival interval of 2 minutes. The generator produces customers and feeds them into the FIFO queue block. The queue results are fed to a plotter and averaging blocks that keep track of average line length and waiting time over repeated trials.

click on run/ simulation setup to see that the time unit is set to minutes, each simulation goes for 120 minutes, and a simulation runs the 120 minute queuing model for 100 repetitions. Double-clicking the server (delay) block will allow you to see the activity delay is initially set for a constant 2 minutes. only one of the servers is connected to the queue output. Just to get a feel for how much variation there is in these, you should run the 100 rep experiments three times and record the results. Run the model 3 times and at the end of each run, double-click the mean and variance blocks and record results for queue length and waiting time for each of the three runs.

Notice there is a large variation in the results.

after recording these results, connect the random number generator to the Delay connector on the delay block and perform three more runs, recording the same results for average queue length and waiting time.

The random number generator will keep the same average service time (delay) of two minutes, but adds random variation that fits an Erlang distribution. You can see this by double-clicking on the random number generator.

## Experiment 2: Service increase to two servers

Connect the queue output to the second server as well. This one is already set up with the random number generator. rerun the model three more times and note how additional service capacity drastically reduces waiting lines and waiting times.

## Experiment 3: Effect of combining queues

Set up a second generator, queue and server so you can run two one-server systems in parallel. The easy way is to copy and paste to duplicate the objects, then hook them together in the required configurations. Gather data from both on line length and average wait times. Do this for 3X100X120 minutes and calculate averages.

Combine the two by using a combine block to feed customers from both of the generators into one queue, which then feeds two servers in parallel. This simulates a combination of two queues with arrival rates of 30 per hour. Now the combined arrival rate is one per minute or 60 per hour. Feeding this combined queue into two servers, depending on which is free. Compare to those for the two independent single-server queues with arrival rate of 30 that was studied in the second experiment.

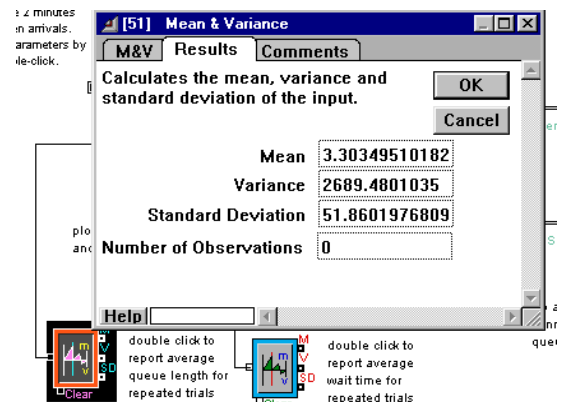
What you should see, if experimental variation allows, is a less-than doubling of the queue length and a substantial reduction in average waiting time for the customers. This effect is why serpentine combined queues have become so prevalent in places like banks and airports in the last 15 years. This is operations management at work improving our lives!

## For your amusement: Animation

- no need to report this. some minor animation effects can be observed by checking show animation under the run menu. This is cute, but the display drastically slows down the simulation, so if you want to do the studies with animation turned on, it would be best to put in a case of beer and a few movies to watch while the computer runs. In any case, be sure to cut back from 100 repetitions in the run /simulation setup menu. In the event your simulation runs amuck (or amok), you can stop it under the run menu.

## Report

Make sure you replace my name with yours. Print out the diagram of the model after modifying to two separate queues. the easiest way to do this is via the PrintScr button to capture the screen on the clipboard, then paste into WORD or EXCEL. Also print out some data after double-clicking the report average queue length module. This data must correspond to one of the results reported in your short report below.



Submit the two printouts along with a short 1-2 page report with tables of means (don't show variance or s.d.) showing queue length and waiting times for each of the triplicate experiments and write a few sentences commenting on the significance of the results of each experiment. **Everyone must report their own data and conclusions.**

I Hope I've set this up so you have to do a little fooling with it just so you know what you are doing, but it shouldn't be more than a few hours of what I hope will seem like play. This is not a complex or long exercise. Avoid MITDC.