Performance Engineering
Queueing Theory

Performance Engineering Laboratory

University College Dublin

Queueing is EVERYWHERE

- The same approach(es) can be applied to solve queueing problems in very different areas

Queueing Introduction

- When you have to wait in a bank, or a supermarket check-out line, or at the doctor's office, or to get into a nightclub...
- When you drive/cycle in the city, fly on a plane, take the bus...
- When you browse a website, send an email, make a phone call...
- ...then you are a customer in a queueing system

- Most real queueing systems cannot be solved exactly
- Usually only approximate solutions can be found...
- ...but even these can give vital and surprising insights

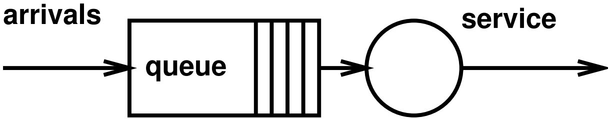
Queueing for Performance Analysis

- Stochastic Service Systems – queueing is waiting in line
- Customers, or units, which require service are usually identical in their properties
- Servers, or service capacity, performs the service operations
- There is an inability to schedule demands for service and hence there is unpredictable timing
- Also lack of predictability of the time duration of the service operation

Scheduling and Random Events

- If the customers and their requirements are known in advance then can schedule the capacity and work to match exactly. This can be a complex problem.
- However when the demand or service, or both, is random, then normally have to assign more capacity than is really needed.
- Even with this larger capacity the customers must queue for service.
- Queueing is the same for all applications, regardless of the customers or the service.

System Flow

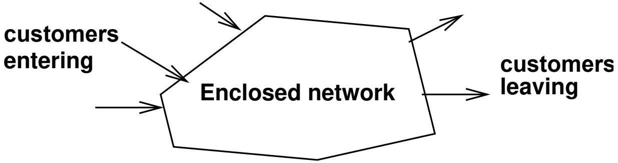


- Arrival process, λ average rate, distribution
- Queue process, method, size
- Service process, μ average rate, distribution
- Arrival and service processes are stochastic (random with probabilities)

Random Flows

- At any point in a system the times of arrival of the basic data unit, (procedure call, message, person, request) are random variables.
- The time to process this arrival in the system typically depends on some characteristic of the arrival. For example the data within the request could vary the workflow.
- Therefore the processing time for the arrival also becomes a random variable.
- We only deal with steady state flows.

Random Flow



- Consider a closed boundary that contains a system with one or more inputs and one or more outputs.
- No customers are created or destroyed in the system.
- Customers can flow in or out of the system across the boundary or be stored in the system.

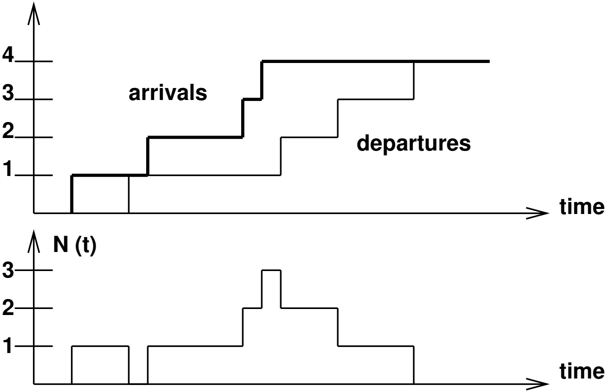
Stable System In Steady State

- If the average input rate across the boundary exceeds the average output rate, then the number of customers stored within the system constantly increases. This is an unstable queue and is out of scope here.
- If the average output rate exceeds the average input rate, the number of customers stored constantly decreases to zero, at which point the average output rate no longer exceeds the average input rate.
- For stable steady state :

$$\text{Average input rate} = \text{Average output rate}$$

- Assumptions for modelling: https://www.youtube.com/watch?v=hJ2CmtDvmv8 1:34-end

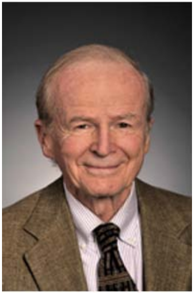
Number in the System - Mathematical Notation



Little's Law

- The average number of customers stored in a system is equal to the product of the average arrival rate to the system and the average time spent in the system.

$$N = \lambda T$$



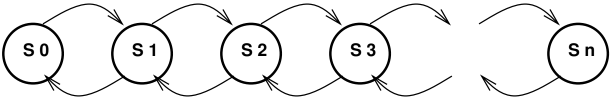
John Little *then at Case Western Reserve University*

- This is for any system arrival, service discipline and service distribution. If you observe a blackbox and see the average rate of arrivals and the duration spend inside, then you can estimate the number in the system.

State Of Queueing System

- A queueing systems can be thought of having a number of states, where the state is usually defined to be the number of customers in the system.
- State 0 is when the system is empty
- State 1 means that there is 1 customer in the system
- State *n* means that there are *n* customers in the system
- If we have a single server system then state 1 means that the queue is empty and the server is busy, and state 2 means that there is 1 customer in the queue and the server is busy.

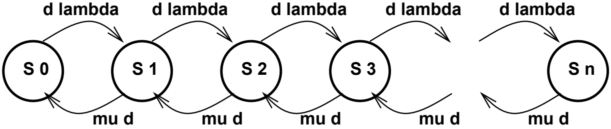
State Of Queueing System



- If we examine the system every *d* seconds, where *d* is very small, then there is likely to be only 1 arrival and/or 1 departure occuring (this type is called a birth-death system).
- Therefore we can get either an arrival on it's own and we move up a state, or we get a departure on it's own and move down a state.

State Of Finite Queueing System

- We could get both an arrival and departure where we stay in the state that we are in, or we get nothing happening and we remain in the same state also.
- If λ is the arrival rate then the probability of getting 1 arrival in a very short time interval d is λd. Also if the departure rate is μ then the probability of getting 1 departure in a very short time interval d is μd.



Probability Of A Finite Queueing System

- The long term probability of being in any state S_n is defined by P_n. If this is in steady state, then we can cut the chain at any point and balance the flows in each direction.

$$\begin{aligned}
\lambda d\ P_0 &= \ \mu d\ P_1 & P_1 &= \frac{\lambda}{\mu}\ P_0 \\
\lambda d\ P_1 &= \ \mu d\ P_2 & P_2 &= \frac{\lambda}{\mu}\ P_1 \\
\lambda d\ P_2 &= \ \mu d\ P_3 & P_3 &= \frac{\lambda}{\mu}\ P_2 \\
& & P_i &= \frac{\lambda}{\mu}\ P_{i-1} \\
& & P_n &= \frac{\lambda}{\mu}\ P_{n-1}
\end{aligned}$$

Solving The Flow Balance Equations

- We notice that the dependency on d is gone.
- We also see that the dependency on the actual values of λ and μ are gone and all that matters is the ratio of them, or $\rho = \frac{\lambda}{\mu}$.
- We need 1 more equation to solve this and we get it from the fact that we must be in a state at every time, or the sum of the probabilities must be 1.

$$1 = P_0 \; + \; P_1 \; + \; P_2 \; + \; P_3 \; + \; \cdots \; + \; P_{n-1} \; + \; P_n$$

Solving The Flow Balance Equations

$$1 = P_0 + P_1 + P_2 + \cdots + P_{n-1} + P_n$$

$$1 = P_0 + \rho P_0 + \rho^2 P_0 + \cdots + \rho^{n-1} P_0 + \rho^n P_0$$

$$P_0 = \frac{1}{1 + \rho + \rho^2 + \cdots + \rho^{n-1} + \rho^n}$$

$$\sum_{i=0}^{\infty} \rho^i = \frac{1}{1-\rho} ; \qquad \rho < 1$$

$$\sum_{i=0}^{n} \rho^i = \sum_{i=0}^{\infty} \rho^i - \sum_{i=n+1}^{\infty} \rho^i = \frac{1}{1-\rho} - \frac{\rho^{n+1}}{1-\rho} = \frac{1-\rho^{n+1}}{1-\rho}$$

Solving This Finite Queue

$$P_0 = \frac{1-\rho}{1-\rho^{n+1}} \qquad P_1 = \rho\ P_0 = \frac{\rho\ (1-\rho)}{1-\rho^{n+1}}$$

$$P_i = \rho^i\ P_0 = \frac{\rho^i\ (1-\rho)}{1-\rho^{n+1}}$$

$$P_n = \rho^n\ P_0 = \frac{\rho^n\ (1-\rho)}{1-\rho^{n+1}}$$

- For this finite queue the probability of blocking, P_B, is given by the probability of being in S_n, or $P_B = P_n$.
- For the infinite queue there is no loss and the equations are different.

Solving For The Infinite Queue

$$P_0 = \frac{1}{1 + \rho + \rho^2 + \cdots} = 1 - \rho$$

$$P_1 = \rho \, P_0 = \rho \, (1 - \rho)$$

$$P_i = \rho^i \, P_0 = \rho^i \, (1 - \rho) \qquad [\mathit{Finite} \, \frac{\rho^i \, (1 - \rho)}{1 - \rho^{n+1}}]$$

- There is no loss in this system and the queue could be unstable if $\rho \geq 1$. In fact for $\rho = 1$ the equations that we have derived do not hold.
- Once we have found the state probabilities we can then find out any information that we want from the system.

Calculation of Average Number and Wait Time

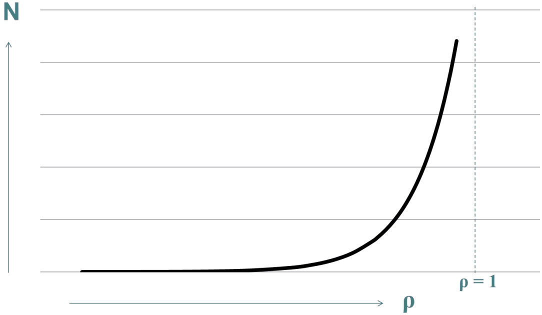
- We might want to find the average number in the system, N, the average number in the queue, N_q, the average waiting time in the system, T, and the average waiting time in the queue, W.
- To find out any of the above, we usually find N first and then use Little's law to find T and then subtract the service time to find W and then Little's law to find N_q.
- To find N we use the equation :

$$N = \sum_{i=0}^{\infty} i \; P_i = \frac{\rho}{1-\rho}$$

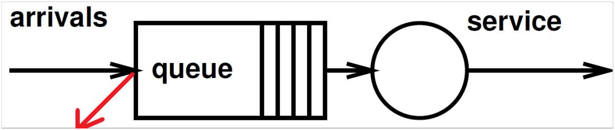
$$N = \sum_{i=0}^{n} i \; P_i = \frac{\rho + (\rho n - n - 1)\rho^{n+1}}{(1-\rho^{n+1})(1-\rho)}$$

Queueing results

- A non-linear curve
  not a linear relationship like you might (naively) expect...



Two Types of Arrivals In The Finite Queue



- Arrivals that enter the system
- Arrivals that are only offered but get rejected
- Define λ to be the offered arrival rate to be consistent with the other definitions
- Define γ to be the carried arrival rate or the throughput
- We did not have this problem with the infinite queue as everything that was offered, was also carried
- Can relate the two arrival rates by the following :

$$\gamma = \lambda (1 - P_B)$$

Careful With The Finite Queue

- When using Little's law for a finite queue must use the carried arrival rate rather than the offered arrival rate.
- The efficiency of this finite queue is defined to be the throughput divided by the service rate or :

$$\text{efficiency} \;=\; \frac{\gamma}{\mu} \;=\; \rho\,(1\;-\;P_B)$$

- Can work out the average waiting times other ways than using Little's law. For example can see that if we arrive when the system is in S_0 then we wait $1/\mu$, and if the system were in S_1 then we wait $2/\mu$.

Waiting In The Finite Queue

$$T = \frac{1/\mu\ P_0 + 2/\mu\ P_1 + \cdots + n/\mu\ P_{n-1} + 0\ P_n}{\sum_{i=0}^{n-1} P_i}$$

$$T = \frac{1/\mu \sum_{i=0}^{n-1} (i+1)\ P_i}{1 - P_B}$$

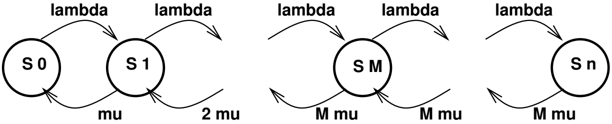
- Should be able to solve this and then using Little's law get the average number in the system to be the same as we got before. Remember that Little's law is now given by :

$$N = \gamma\ T$$

Congestion In Queues

- If we have to wait too long in the infinite case or if there is too much loss in the finite case what can be done?
- In the infinite case could make a finite queue and this would reduce the waiting time. In the finite case make the waiting room larger which would reduce the loss.
- However these are really only temporary solutions and what is probably needed is to increase the service rate, possibly by having more than 1 server. Suppose that we have M servers then how do we solve the system?

Multiple Servers In A Queue



- Here there is a maximum service rate of $M\mu$ but this will only occur when there are at least M customers in the system.
- If there are less than this number of servers in use, then the service rate is just that number of customers multiplied by the service rate of each.
- This is a state dependent queue and can be solved in a similar manner to that which we have just solved.

State Dependent Queues

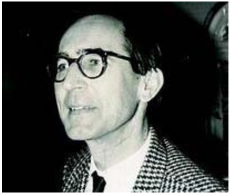
- Rather than having M servers it is always better to have 1 server at M times the rate of the previous one. However this is not always possible to do.
- Can also have state dependent queues that are due to the arrival rate depending on the number of customers in the queue.
- This would be similar to the customer seeing that there is a large queue and then maybe not going into the queue.
- Maybe then try to hide the queue from the customer.

General Information On Multiple Server Queues

- Can have multiple queues for multiple servers or can have a single queue for multiple servers.
- Does it matter which one we use?
- Always better to have a single queue rather than have a number of seperate queues.
- This is because there might be a customer waiting in one of the queues while a server in another queue is idle. This can be achieved in some cases but again can not always be done.
- Video on Car Parks: https://www.youtube.com/watch?v=Ww7oENHrRpA

Different Queueing Models

- In the previous part there was no explicit assumption on the statistical properties of the arriving or departing process.
- A notation has been developed for queueing models called the Kendell notation developed by D. G. Kendall in 1953.



- The most basic is the *arrival distribution / service distribution / number of servers*

Kendall Notation

- Three common types of distributions are
  - Memoryless, or Markovian (M) process, that is equivalent to one with Poisson statistics.
  - Deterministic (D) where there is no randomness, fixed service or inter-arrival times.
  - General (G) where it does not fit one of the other known process.
  - Others are E, H and other extensions
- Therefore we can have for example:
  - M/G/1 is Markov arrivals, General service and single server
  - D/M/2 is Deterministic arrivals, Markov service and 2 servers
- There are extensions to this. If the System is Finite with say k places, then we use M/G/1/k or D/M/2/k

Full Kendall Notation

- The full notation is as follows, where there is a default if empty
  - Arrival process
  - Service process
  - Number of servers
  - Queue System capacity (default infinite) = total number of customers which can be accommodated in the system
  - Population size (default infinite)
  - Queueing discipline (default FIFO)
- Therefore M/M/5/40/1200/LCFS-PR is Markov arrivals and service with 5 servers, 35 waiting places and a possible population of 1200, and Last Come First Served with Preemptive interrupt and Resume

Differences Between System And Queue

- N is the average number of customers stored in the complete system of the queue and the server. N_q is the number stored in the queue alone.
- Little's law can be used to relate N and N_q
  - $N = \lambda T$ for complete system
  - $N_q = \lambda W$ for buffer only
- W is the average wait in the queue, and T the average wait in the complete system.
- T and W can be related by $T = W + 1/\mu$ (single server)
- $N = \lambda T = \lambda W + \lambda/\mu$
- Average number of customers stored in the system $=$ average number of customers stored in the queue $+ \rho$

Review Of Queueing: Two basic types

- **Delay –** Infinite waiting room, arrival rate $=$ throughput, no loss at all, can be unstable so need to keep $\rho < 1$
- **Loss –** Finite waiting room with n places, arrival rate is greater than throughput, lose some arrivals, stable queues, have probability of blocking P_B, and ρ can be any value. Have $P_B \; = \; P_n$ and have the carried arrival rate given by

  $$\gamma \; = \; \lambda \, (1 - P_B)$$

  where $\gamma/\mu < 1$ always.

Review Of Queueing: Flow Balance

- Cut the state diagram and what flows left is equal to the flow right. (There is another way of looking to see how to get into and out of a state).
- Get a general iterative expression like :

  $$P_i \;=\; \rho^i \; P_0$$

  Then use the normalising equation :

  $$\sum_{i=0}^{\infty} P_i \;=\; 1$$

  to solve for P_0 and then solve for P_i.

Review Of Queueing: Little's Law

- Little's law says that : $N \; = \; \lambda \; T$
- System and queue : $T \; = \; W \; + \; \frac{1}{\mu}$
- Had for the infinite queue the following equations :

$$N \; = \; \frac{\rho}{1-\rho} \qquad T \; = \; \frac{1}{\mu} \, \frac{1}{1-\rho}$$

$$N_q \; = \; \rho \, \frac{\rho}{1-\rho} \qquad W \; = \; \rho \, \frac{1}{\mu} \, \frac{1}{1-\rho}$$

for the markov systems, or the M/M/1 system.

Review Of Queueing

- In a loss system like the M/M/1/n queue we have :

$$P_B \;=\; P_n \;=\; \frac{(1-\rho)\;\rho^n}{1-\rho^{n+1}}$$

- Suppose that the offered arrival rate is twice the service rate and $n = 5$, what efficiency do we have ?

$$\lambda = 2\mu \quad ; \quad \rho = 2 \quad ; \quad P_B = \frac{(1-2)2^5}{1-2^6} = \frac{32}{63}$$

$$\text{efficiency} \;= \frac{\gamma}{\mu} = \frac{\lambda(1-P_B)}{\mu} = 2(1-32/63) = 98.4\%$$

Average Number In System N

$$N \;=\; \sum_{i=0}^{\infty} \; i \; P_i \;=\; \sum_{i=0}^{\infty} \; i \; \rho^i \; P_0 \;=\; P_0 \sum_{i=0}^{\infty} \; i \; \rho^i$$

$$\sum_{i=0}^{\infty} \; i \; \rho^i \;=\; \sum_{i=0}^{\infty} \; \rho \; \frac{d \; \rho^i}{d \; \rho} \;=\; \rho \; \frac{d}{d \; \rho} \sum_{i=0}^{\infty} \; \rho^i$$

For the infinite case we have :

$$\sum_{i=0}^{\infty} \; i \; \rho^i \;=\; \rho \; \frac{d}{d \; \rho} \; \frac{1}{1-\rho} \;=\; \frac{\rho}{(1-\rho)^2}$$

$$N \;=\; P_0 \; \frac{\rho}{(1-\rho)^2} \;=\; (1-\rho) \; \frac{\rho}{(1-\rho)^2} \;=\; \frac{\rho}{1-\rho}$$

Average Number In The Finite Case

$$N = P_0 \, \rho \, \frac{d}{d\,\rho} \sum_{i=0}^{n} \rho^i$$

$$P_0 = \frac{1-\rho}{1-\rho^{n+1}} \qquad \sum_{i=0}^{n} \rho^i = \frac{1-\rho^{n+1}}{1-\rho}$$

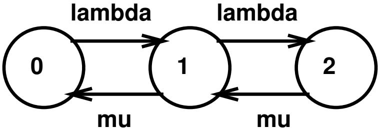
$$N = \frac{1-\rho}{1-\rho^{n+1}} \; \rho \; \frac{d}{d\,\rho} \frac{1-\rho^{n+1}}{1-\rho}$$

$$N = \frac{1-\rho}{1-\rho^{n+1}} \; \rho \; \frac{-(n+1)\rho^n(1-\rho) \; + \; (1-\rho^{n+1})}{(1-\rho)^2}$$

$$N = \frac{\rho \, (1 \; - \; (n+1)\rho^n \; + \; \rho n \rho^n \,)}{(1-\rho^{n+1})(1-\rho)} = \frac{\rho \; + \; (\rho n - n - 1)\rho^{n+1}}{(1-\rho^{n+1})(1-\rho)}$$

Example before $\rho = 2, n = 5$ work out 4.1

Example Of Two Place Finite Queue



$$P_1 \; = \; \rho \; P_0 \qquad P_2 \; = \; \rho^2 \; P_0 \qquad P_0 \; = \; \frac{1}{1+\rho+\rho^2}$$

$$N \; = \; 1 \; P_1 \; + \; 2 \; P_2 \; = \; \frac{\rho \; + \; 2 \; \rho^2}{1+\rho+\rho^2}$$

$$T \; = \; \frac{\frac{1}{\mu} \; (P_0 \; + \; 2 \; P_1)}{1 - P_B} \; = \; \frac{1}{\mu} \; \frac{1+2\rho}{1+\rho+\rho^2} \Bigg/ \left[1 - \frac{\rho^2}{1+\rho+\rho^2}\right]$$

Example Of Two Place Finite Queue

$$T \;=\; \frac{1}{\mu}\,\frac{1+2\rho}{1+\rho+\rho^2} \Bigg/ \left[1-\frac{\rho^2}{1+\rho+\rho^2}\right] \;=\; \frac{1}{\mu}\,\frac{1+2\rho}{1+\rho}$$

- Now use Little's law to go back to the number in the system N, except must use γ instead of λ. So Little say $N=\gamma T$.

$$\begin{aligned}\gamma\;T \;&=\; \lambda\,(1-P_B)\;T \;=\; \rho\,\frac{1+2\rho}{1+\rho}\,(1-P_B)\\ &=\; \rho\,\frac{1+2\rho}{1+\rho}\left(1-\frac{\rho^2}{1+\rho+\rho^2}\right) \;=\; \rho\,\frac{1+2\rho}{1+\rho+\rho^2}\end{aligned}$$

- Which is the same as we had for N before!

Why Poisson Arrival Input Process?

- Assume that each customer has a small probability of arriving in any short time interval, and all arrivals are independent.
- The average number of customers arriving in time *d* is λd
- What is the probability that over *d* seconds there are going to be *k* arrivals?
- We call this $p(k)$ and we have $\sum_{k=0}^{\infty} p(k) \ = \ 1$
- Divide the interval into *n* pieces and we are going to use the "law of small numbers"
- Expected number in one piece is now given by $(\lambda d)/n$.
- Ignore multiple events occuring as we can make *n* as large as we need to so that 2 events can not happen in one interval.

Pushing The Internals To Be Small

- The expected number also equals the probability of one event in this piece.
- Probability of 1 customer is : $(\lambda d)/n$
- Probability of 0 customers is : $1 - (\lambda d)/n$
- Probability of k customers is :

$$\binom{n}{k} \left(\lambda d/n\right)^{k} \left(1 - \lambda d/n\right)^{n-k} \;=\; p(k)$$

- Now let n get large

$$\binom{n}{k} \;=\; \frac{n!}{k!(n-k)!} \;\approx\; \frac{n^k}{k!}$$

Definition Of Poisson Process

$$p(k) = \frac{n^k}{k!}\left(\frac{\lambda d}{n}\right)^k \left(1-\frac{\lambda d}{n}\right)^n \left(1-\frac{\lambda d}{n}\right)^{-k}$$

$$\lim_{n\to\infty}\left(1-\frac{\lambda d}{n}\right)^{-k} = 1$$

$$\lim_{n\to\infty}\left(1-\frac{\lambda d}{n}\right)^{n} = e^{-\lambda d}$$

$$p(k) = \frac{n^k}{k!}\left(\frac{\lambda d}{n}\right)^k e^{-\lambda d}$$

- Which is the definition of the Poisson process.

What Does Poisson Mean?

- Poisson arrivals $\Longleftrightarrow$ memoryless arrivals.
- That means that the time to the next arrival is independent of how long it was since the last arrival.
- Memoryless holding times $\Longleftrightarrow$ Poisson departures, which have exponential distribution service times.
- This means that the probability of a customer leaving is independent of how long the customer has been in service!
- Piece on psychology: https://www.youtube.com/watch?v=_CBD2z51u5c
- Interest to our research is traffic: https://www.youtube.com/watch?v=iHzzSao6ypE