Introduction to Queueing Theory

Prof. N. Selvaraju
Department of Mathematics
Indian Institute of Technology Guwahati, India

Lecture - 28
Introduction to Queueing Networks, Two-Node Network

Hi and hello, everyone. What we have seen so far are all queueing systems or queueing models, which are used to model a single node or a single service facility kind of thing. So, all these things, then we can together call them as a single node queueing systems, but in reality, like there is very less likely that you always have encounter only a single node service system. As opposed to this, you have what we call as a network of queues or queueing networks where more than one node is connected in some sense, and the service has to be obtained from more than one node. So, that is the topic of our further analysis in the next few lectures. So, what we are going to study are these queueing networks; in general, it is called all networks of queues, and this is an important area that has wide applications in diverse fields and occurs quite naturally. And, because of the complexity of this, this also poses extremely difficult problems for the analysis also; once you go a little bit beyond that. So, that is the nature of queueing network. So, what do we have?

While a (single-node) queues represent situations where the customer demands one service, that is what we say, meaning that suppose you go to a petty shop kind of place where you just go and buy something and come, that is it. So, that is a single node system, basically. But, whereas, if you go to some other place even when you want to get service say in an airport terminal, for example, first you have this luggage screening, then checking in, then security check then like you have multiple stages where you need to go through and is that is the situation that we will come here actually. So, a queueing network, on the other hand, represents situations where a customer may need more than one service or different kinds of service. If you are going to a hospital, for example, where first you may be seen some preliminary investigation, then some tests followed by some specialist examination and so on, different kinds of service might be there. So, this may be from different servers and may be required to wait before each of these different service channels for service, bank counters going from one counter to the; car repair facility. For example, when a car is being repaired, it is not that every part of the car needs repair. So, there may be one service facility, or service repairmen also like who will repair only the batteries; someone will look after the tires, someone will look after the engine component like this different people will be there. A car may not require repair from all the servicemen, but it has come for repair. So, you will have a scenario where they will go through some of the nodes, some of these service centers for a complete service; the whole system is now a network. And, the customer would need to get service from more than one node possibly. Even if it is one node out of those network nodes, it is fine.

But, some customers will need in two, some customers will need in three, that three could be different for different customers, and so on. This is a scenario that you have here, and that is what is trying to model through a network of queues. Here is why we have to do it as a network rather than as a single node; the reason is clear only after you do one processing hospital, for example, if you take it. Only the preliminary investigation or the screening is done, then the doctor will prescribe what are the tests to be done, then accordingly you go for the other test and then a specialist will

come, then some more test, some more screenings and so on things will go. So, it has to go through. So, from here, the first stage, you will go to the second stage; you directly will not go to the second stage; suppose if you call these as first and second stages. So, that kind of scenario it requires. So, it has to be networked; it is not like individually you can study this. So, when you are studying together, obviously, the problems will become more complex, and that is what we are going to encounter.

But, it has diverse applications, and it is quite naturally occurring in different situations like in an assembly line where products are being assembled part by part in each stage; maintenance operations again the car up repair facility with that of maintenance of also in a similar way; airport terminals; quite naturally in computer and communication systems and networks. So, you have communication networks, computer networks that is how you call it by nature because things are networked together, and it has to pass through different stages. Even if you are, some multiprogramming system that you have in mind or a cellular network kind of thing that you have in mind, anything like it has to go through multiple nodes, what we call in general.

So, what we give here is certain basic concepts and results in a very simplified setup that is what we are going to consider. But, they are also useful in their own light, and then they do, throw enough light on the behaviour of such queueing networks. In general, what you can expect and in general, what you have to look for in such queueing networks; it will exhibit these ideas here, and again this itself is important in designing of many manufacturing or production facilities or computer communication networks and so. Of course, we since we may not go to a deep except at the top layer level. So, there are many good books available if you want to get a deeper understanding of this. For example, Bolch et al. (2006), Gelenbe and Pujolle (1998) like and host of other like you know there is Kelly's book is freely available which is a very classical one and then Waldron and so many other books are fine, of course, you can find and then they can be looked into for a detailed look into this topic. So, basically, what we have is you are looking at the study of queueing networks.

Now, what are the different types of queueing networks that one can broadly classify into them. We basically have three, of course; the third one is a mixture of the first two. So, you can call it even you may say to open and closed alone that maybe you can classify them.

- Main types of queueing networks:
  - ⋆ **Open queueing network**: Customers (of one or more classes) enter the system from outside, may not be from the same node, or it may be from different nodes they may enter. And they pass through different stages or different nodes on their own as per the requirement as per the design, and eventually, they will leave the system after service at one or more nodes.

    So, there is an input to this system which is what is coming from outside; the external arrivals are there, and after getting service in one or more nodes, they eventually leave the system. So, that is the kind of thing that we call an open queueing network.

    As opposed to this open queueing network, we have what we call a closed queueing network.
  - ⋆ **Closed queueing network**: A fixed number of customers again, they could be of a single class, or multiple classes circulate in the system moving from one queue to the next and getting served at individual nodes.

So, here no external arrivals happen, and no departure happens from the system.

It is just that there is a fixed number of customers who go move from one queue to the other and get their service done. So, that is what is a closed queueing network, and as the name suggests, a mixed queueing network basically consists of both this open and close.

⋆ **Mixed queueing network**: Open for one class of customers and closed for another class of customers.

So, it is not just for one; it could be for some number of suppose if you have multiple classes of customers. So, there is at least one class for which this is open; there is at least one class for which this is closed. Whereas, in the open and closed all of the classes, if you have multiple classes of customers. So, all of them have to be either a single type, either open or closed, to be called as the open or closed queueing network. So, we might mainly be concentrating on this open and closed network only.

- Just as in the case of single queues, performance measures of a queueing networks can be obtained.
- One important measure is the time taken to serve one customer in the system.

So, this is more meaningful here because in the open context because someone comes from outside, and then, he eventually leaves the system. So, it may so happen that this particular customer may loop n numbers so many times, and it can go, and again it can come he can loop many times before eventually, he leaves the system. So, in that case, the time taken for serving one particular customer in the system is what is more relevant here.

- Analysis would give us measures for the number of customers in each node so that estimates of buffer requirements can be made.
- Bottlenecks may be identified leading to better design (in terms of more servers/waiting spaces).
- It may be easier to model a complicated service scenario as a queueing network in order to capture better the way the service is actually provided rather than treating the whole thing as a single block box rather than that if you do it individually that would play a much more convenient way you can capture the behavior.
- Networks of queues requires some additional specification such as
  - Interconnection between the queues

- Routing strategy (deterministic, class based or probabilistic)
- Strategy to handle blocking if the destination queue is a finite-capacity queue

- Possible that a customer returns to a queue for another service (i.e., feedback).
  ▶ Feedforward networks are open networks with no feedback (a queue will be visited at most once).

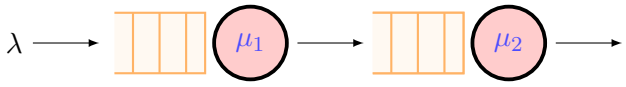
Why are we putting it as feed-forward network feedback? Because these features play a critical role or make the complexity of the analysis easier or tougher depending upon whether there is a feed-forward or feedback in those cases. As we go along, you will understand why. That is why now we want to; there are many other things like joint queue and things like that which we will come later if there is a possibility of time. So, that kind of thing can happen. So, that is why we want to specify certain terminologies words to make it familiar that in these kinds of situations, things might be easier or a little tougher depending upon what the features that you may have. So, in a network, since this kind of thing can happen, a customer may visit all three nodes, and again he may come back to node 1. He can go here, come here and then go again, come here possible. So, it is possible in this particular scenario, but still, he can come back to this, right. So, feedback is very much a possibility in general in such networks. So, if you do not have feedback, feedback essentially in production, and everything could be represented by the defective items cases. So, that is possible one has to keep that in mind.

So, these are certain basic notions and basic ideas about queues. Now, let us take a very simple two-node queueing network and see what we are trying to do is; the number of customers in each node we will see what happens to that.

- Consider the following simple network of two queues (or nodes):
  ▶ Customers arrive at the first node according to a Poisson process with rate λ.
  ▶ The fist node is a single-server system with exponentially distributed service times (with parameter μ_1) having infinite queueing capacity and the service discipline here FCFS.
  ▶ Once the service is completed in the first node, the customer moves to the second node which is also a single-server system with exponentially distributed service times (with parameter μ_2) having infinite queueing capacity and again the queueing discipline is FCFS.
  ▶ Once the service is completed in the second node, the customer departs the system.
  ▶ No feedback or departure at the first node. No arrival at the second node.

  So, the second stage could be represented in this manner like there is some arrival process, but the service is exponential M, and there is a single server infinite capacity FCFS everything is there.

  ▶ The system can be represented as $M/M/1 \rightarrow \bullet/M/1$



- The system state can be represented as a two-dimensional CTMC with state space $S = \{(n_1, n_2) : n_1, n_2 = 0, 1, 2, \dots\}$.

- Denote the probability of n_1 customers in the first node and n_2 customers in the second node in steady state by p_{n_1,n_2}.

  When we say in the queueing network in the first queue, second queue, we always mean the number in the system that you have to understand; because the word queue is used does not mean that it is a number in the queue. So, it is always a number in the system when you are talking about this kind of thing.

- Then, the steady state solution for this system exists under the condition that $\rho_1 = \lambda/\mu_1 < 1$ and $\rho_2 = \lambda/\mu_2 < 1$, (provided CTMC is positive recurrent and has a steady-state unique limiting steady-state distribution) and can be obtained from the balance equations given by

$$\begin{aligned}
(\lambda + \mu_1 + \mu_2)p_{n_1,n_2} &= \lambda p_{n_1-1,n_2} + \mu_1 p_{n_1+1,n_2-1} + \mu_2 p_{n_1,n_2+1}, \quad n_1 \geq 1, n_2 \geq 1 \\
(\lambda + \mu_1)p_{n_1,0} &= \lambda p_{n_1-1,0} + \mu_2 p_{n_1,1}, \quad n_1 \geq 1 \\
(\lambda + \mu_2)p_{0,n_2} &= \mu_1 p_{1,n_2-1} + \mu_2 p_{0,n_2+1}, \quad n_2 \geq 1 \\
\lambda p_{0,0} &= \mu_2 p_{0,1}
\end{aligned}$$

  It can be shown that

$$p_{n_1,n_2} = \rho_1^{n_1}\rho_2^{n_2}p_{0,0} \quad \text{and} \quad p_{0,0} = (1-\rho_1)(1-\rho_2).$$

  Thus,

$$p_{n_1,n_2} = \left[(1-\rho_1)\rho_1^{n_1}\right]\left[(1-\rho_2)\rho_2^{n_2}\right],\ n_1, n_2 \geq 0.$$

Now, if you look at here, $\left[(1-\rho_1)\rho_1^{n_1}\right]$ is a geometric distribution with parameter ρ_1, and $\left[(1-\rho_2)\rho_2^{n_2}\right]$ is a geometric distribution with parameter ρ_2, and this is the first one is already you know it is an $M/M/1$ queueing system, and $\left[(1-\rho_1)\rho_1^{n_1}\right]$ is nothing, but the steady-state distribution of number in the system in an $M/M/1$ queueing system and $\left[(1-\rho_2)\rho_2^{n_2}\right]$ is also a similar thing that you are obtaining. And, that is the quantity, or that is the nice things that you are seeing it in this expression here now this tells some stories. So, what is this?

p_{n_1,n_2} is the joint distribution of number in the system or number of customers in node 1 and node 2 together, and $\left[(1-\rho_1)\rho_1^{n_1}\right]$ is you can think of it as if the number of customers in the system in node 1 and $\left[(1-\rho_2)\rho_2^{n_2}\right]$ is the number of customers in the node 2, both of them act behaving like an $M/M/1$ queue if you think. So, that is what it is.

- A queueing network of this type where the joint distribution of the number of customers in each node can be written as a product of terms involving the number in individual nodes is referred to as a **product-form network**.

  What we mean is not $p_{n_1,n_2} = \left[(1-\rho_1)\rho_1^{n_1}\right]\left[(1-\rho_2)\rho_2^{n_2}\right]$ expression, but we mean $p_{n_1,n_2} = \rho_1^{n_1}\rho_2^{n_2}p_{0,0}$ expression. So, p_{n_1,n_2} is the joint distribution of n_1 and n_2. So, this is some function of n_1, and some function of n_2 multiplied by some constant $p_{0,0}$ is what is the form that you are having here. And, any network where you get this $p_{n_1,n_2} = \rho_1^{n_1}\rho_2^{n_2}p_{0,0}$ kind of situation, we will see then analysis becomes much simpler in that case. So, that is what is called a product form network.

  ■ And, in many situations and under some conditions on the parameters, this type of solution is observed to hold either exactly or approximately at least.
  There is a whole a lot of literature on product form network itself because like, in $p_{n_1,n_2} = \rho_1^{n_1}\rho_2^{n_2}p_{0,0}$ case, things will become much easier. Imagine that this is the case, and then you are looking at a function of n_1, a

function of n_2, and a constant.

Now, if I want to obtain the mean performance measures, I know in this particular case that I can handle this very easily because of this product form that you have here. There is nothing like $\rho_1^{n_1}$ plus or n_1, n_2 kinds of terms by which you cannot separate it out, whereas $n_1 + n_2$ or $\rho_1^{n_1}$ and $\rho_2^{n_2}$ it is very simple here or anything that is a function of n_1 function of n_2 multiplied by a constant is what should be the form, and that is what is called as the product form networks. So, this is, say, for example, of a product form network, but this is a bit more than that, even. Here what you have?

- The form of the solution indicates that in steady state each node behaves independently of the other (the joint distribution factors into product of marginals).

Here actually, you are seeing is $p_{n_1,n_2} = \left[(1-\rho_1)\rho_1^{n_1}\right]\left[(1-\rho_2)\rho_2^{n_2}\right]$ not at $p_{n_1,n_2} = \rho_1^{n_1}\rho_2^{n_2}p_{0,0}$, but if you look at this line, what you see here is that $\left[(1-\rho_1)\rho_1^{n_1}\right]$ is a proper probability distribution, $\left[(1-\rho_2)\rho_2^{n_2}\right]$ is a proper probability distribution, and p_{n_1,n_2} is the joint probability distribution this factors into a product of its marginals. This is the nicest thing that you could have.

$p_{n_1,n_2} = \left[(1-\rho_1)\rho_1^{n_1}\right]\left[(1-\rho_2)\rho_2^{n_2}\right]$ is also under $p_{n_1,n_2} = \rho_1^{n_1}\rho_2^{n_2}p_{0,0}$ scenario, but $p_{n_1,n_2} = \rho_1^{n_1}\rho_2^{n_2}p_{0,0}$ is sufficient for us to call this as a product form network, but this is what is happening here. So, this form of solution indicates that in a steady-state, each node behaves independently of the other; otherwise, this would not have happened. One could expect that say, from node 1 when they come out, the average rate of output could equal λ, but what we see here is more than that, not just the mean rate of arrivals or the mean, but even the variance, in fact, the whole distribution sense.

- The second stage behaves like a system with an input process that is Poisson with rate λ. That is, it behaves as an $M/M/1$ queue independent of the behaviour of the first stage.
- This can be proved if we can characterize the output process of the first node.
- The output process (distribution of times between successive departures) of the first node is the input process to the second node.
- Now, how do we characterize is what is the result, which is known as Burke's theorem, who gave the result in 1956, and this determined the output process of the $M/M/c$ queues.

So, what we have considered as $M/M/1$, but in general, it is $M/M/c$ queue.
♦ Of all the systems with FCFS, $M/M/c$ is the only system with the stated property.

So, that is what we will see, and with that result, then we will generalize this two-node to multiple nodes in this particular fashion before we go take it up to any other node, ok so that we will see in the next lecture.
Thank you. Bye.