Introduction to Queueing Theory

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

Lecture - 46
Vacation Queues: Introduction, M/M/1 Queues with Vacations

Hi and hello, everyone. What we have seen in the previous two lectures was General Queueing Models. Now, what we see next is what is called as Queues with Vacations, and what is this? This is not, I mean, in any way, a special case of $G/G/1$ models, but it is more we are going to look at the Markov or semi semi-Markov setup only. Since this is also queues with vacation is also a broad class of models which like the impatience behaviour or like the retrial behavior is also a very important class, but we will use both the $M/M/1$ and $M/G/1$ together when we want to analyze this. We are not going to look at in the $G/G/1$ setup. So, in that way, this falls into the category of semi Markovian queues, one can say.

Example.
(i) Production systems: Machines producing certain items may need periodic checking and maintenance. The periods of random lengths of preventive maintenance may be considered as periods of server vacation when the server is unavailable.
(ii) Computer and communication systems: A server in such a system, besides being engaged in primary functions (such as receiving, processing, and transmitting data), has to undertake secondary works such as preventive maintenance or has to scan for new work for occasional periods of time.

Tian and Zhang (2006) is one of the recent books or rather monograph which gives lots of ideas about this. So this is called vacation queueing models theory and applications. Another good reference is basically the little earlier one which is Takagi (1991), which is what one can refer to for various kinds of vacation queueing models, where it arises and what their applications are, and so on. So, that is what we are going to consider.

- There are many variations of queues with vacations.
- Each time when a busy period ends and the system becomes empty, the server starts "vacation" of random length of time.

  So, that is what you normally consider as vacation. I mean, there is no more customer to serve at that point of time the server decides to take a vacation.

- When the server returns from vacation and finds one or more customers waiting, he goes on serving until the system becomes empty (called exhaustive service discipline). Otherwise (i.e. while the server is working), the

queue behaves like a normal queue.

Only when he is in on the server is on vacation; when the customer arrives to an empty system, service will not start because server is in some sleeping mode, you can think of vacation or in sleeping mode or in downtime. It is not in up condition; it is in down condition for whatever reason; that is what it is. But, once he finds on his return from vacation one or more customers, he will start serving the customers, then it becomes like a normal queue, and if that service continues until it becomes empty. Then that is what is called exhaustive service discipline.

- If on return from a vacation (at the end of a busy period), the server finds no customer waiting, he waits for the arrival of a customer. This is called a single-vacation system means he will go for only one vacation, and he will come back. On his return, if he finds that there is at least one customer waiting, he will switch to serving the customer; if not, he will wait for the customer to arrive, much like the normal queue. So, this is a single vacation system, which is denoted by V_s the single vacation case.

- On the other hand, when he returns from vacation, if he finds that no customer is waiting, he will go for another vacation. Again on his return, if he finds at least one customer, he will start serving; if it is not, then he will go for another vacation. So, he will keep on taking vacation until, on his return, he finds at least one customer waiting whenever on his return from a vacation on any return. So, this is called as multiple vacation system, and we denote it by V_m.

  V_m we denote it as multiple vacation, and V_s is single. The s is single, m is multiple. This is what the notation that we will use it just to refer to this. So, there are two scenarios that you could have; you could have multiple vacations and single vacation.

- Now, there are many many other models possible for example, the server goes on vacation, and once it becomes empty and then he comes back, he sees that some, one customer, two customer are waiting, he does not start service. He goes for another vacation, or he will wait, whichever way it is multiple or single vacation period, but he will not start the service until there are N number of customers are waiting in this case. This is referred to as N-policy like this; one could have different variations, and there are plenty of them depending upon the requirement that you can explore it further.

Now, vacations with exhaustive services what we will be considering, meaning that whatever we have mentioned here that once he starts serving the customers, he will serve until the systems become empty; that is what is the case. And, in that case, if v n in a single vacation, it is a single random variable because after a duration of a random amount of time, he will come back to normal period, which is and then he will wait for the customer, or he will start serving depending upon whether the custom there are no customer or there were at least one customer waiting in the queue.

Now, in case of multiple vacation, we can denote the nth vacation by the random variable v_n, and we will assume that this $\{v_n, n = 1, 2, \dots\}$s are a sequence of IID random variable which is independent of the service time and the sequence $\{v_n\}$ of the random variables may be independent of the arrival period, or it may be dependent on the arrival

process. It may be dependent or independent of the arrival process, but we will always assume that this is independent of the service process.

- Let us call this F_v to be the CDF of this random variable v_n, and F_v^* is the Laplace - Stieltjes transform of this random variable.
- Now, if you consider a standard $G/G/1$ queue with inter-arrival time as A with its mean as $1/\lambda$ and variance as σ_A^2 and the service time S with a mean of $1/\mu$ and the variance of σ_B^2, then we shall denote a queue with general inter-arrival and service times and with single vacation by this notation $G/G/1 - V_s$.
- These are all not standard notations, but we will use this and the corresponding queue with multiple vacations by $G/G/1 - V_m$, and also one more thing that you can notice the inter-arrival time here we are calling it is by A, which was equal to T in the case of $G/G/1$ that we have used here.

  So, this is what you have to remember; that is what it is A here. It does not matter; it is just we are defining what it is in the particular context. And the corresponding multiple vacation case is what we call it $G/G/1 - V_m$.
- Here we consider only vacation queues with exhaustive service.
- But, then, as against this exhaustive service discipline, there could be situations that give rise to different varieties of nonexhaustive service discipline, under which the server vacation may start even when some customers are present in the system or he will not start service and so on.
- So, that means two cases:

(1) There is a preemptive or non-preemptive one can introduce where vacations may preempt an ongoing service, for example, in the case of breakdown of the service mechanism. When service is going on for a certain customer suddenly, the server breaks down.

(2) Nonpreemptive, in this case, what where the vacations may commence only at epochs of service completion or vacation termination as in the case of a scheduled maintenance.

So, when the vacation starts, it whether it preempts any ongoing service or without any preempting the ongoing service because the server can take vacation while there are customers waiting, meaning that when they are even even when the service is going on, he can take vacation. So, whether that vacation is a preemptive one or nonpreemptive one, but meaning whether it is preempting the ongoing service of the customer or not, is the question here. If it is a scheduled maintenance, obviously, you would wait for a service completion, but if it is a sudden failure, you will not be able to control whether the what will happen to the current service. So, you could have these two cases.

- There are very different kinds of specific nonexhaustive services such as gated service; what is that which is also prevalent in communication network or computer network areas server takes a vacation, he comes back

from vacation what he will do is that he will close the gate at that point of time and whatever the number of customers who are available at that point of time when the server returns from vacation he will serve only that many customers before he takes another vacation.

It is called gated service. You just put a gate, and then you serve these customers and then leave. And different kinds of limited services. I can assume that a server takes a vacation after serving, say, one customer, maybe like some kind of refueling or something is required. Now, after serving one customer, you will take a vacation. So to replenish or refill whatever item that is required before you start serving the other customers. So, that is a kind of a limited-service decrementing service. Once it comes back, it will be at the full capacity, and then it might be lower, or whatever is the case, there could be different variations that one can think in these cases even within the nonexhaustive service cases, which are all of them are important from their own perspective, from their own applications point of view. They are beyond this exhaustive, nonexhaustive even with the exhaustive then you could have other like now recent phenomena is whatever I which is mentioned as working vacation queue.

One important property that you would observe when you are dealing with such vacation queues is a specific stochastic decomposition property. So, this is for the vacation queues; this is an important property stochastic decomposition because you always look for such kind of decomposition in such models; and what is that?

Under certain conditions on this vacation sequence, $\{v_n\}$ is what? The vacation duration periods, both for the single vacation $(G/G/1 - V_s)$ and multiple vacation models, the steady-state waiting time is the sum of two independent random variables. The steady-state waiting time can be written as a sum of two independent random variables - one is the waiting time in the same $G/G/1$ queue but without vacation. The other is a random variable related to this vacation duration sequence $\{v_n\}$.

So, this is what is the decomposition. So, the steady-state waiting time is a sum of two random variables; one is the usual $G/G/1$ model without vacation. The other quantity is the quantity related to the vacation sequence.

Now, like when this has this, then one need not worry about only this second part alone; you need to worry because the first part comes from the analysis of the the corresponding model, but without vacation whatever be the whether it is $M/M/1$, $M/G/1, G/M/1$ whichever model you take it or $G/G/1$, in general, we will say. The the corresponding model without vacation plus this to the the quantity that is related to the vacation duration.

Now, if the queues have Poisson input, then such a decomposition property holds for the queue length distribution, not just for the waiting time. Waiting time holds in that generality $G/G/1$, but for queue length, you need Poisson input case; in that case, even for queue length also you have this decomposition property. It can be written as the sum of two random variables independent random variables; one is the queue length of the corresponding model, but without vacation, the other is connected to the vacation sequence. So, because of this kind of stuff that we want us to say, like we are studying this actually after our $G/G/1$ model though the model that we consider will not be considered under $G/G/1$ case. So, this is an important property you always look for in any vacation queueing model, such a stochastic decomposition property.

Now, let us consider as the first model an $M/M/1$ queue with vacations. So, this is we could have done along with the Markovian queue, but we are doing it for the reasons which we have already said. What we have now scenario?

- You have a usual $M/M/1$ queueing system with usual notation and the stability condition rho which is $\lambda/\mu < 1$.

  In addition, what we have? The server vacation is what we are introducing. The server goes on vacation for an exponentially distributed duration with rate θ. Now, this is happens as soon as and only when the queue becomes empty. So, that means the exhaustive case we already said that we are looking at only the exhaustive service discipline case is what we are considering.

- So, the server goes on vacation for an exponential duration with rate θ as soon as and only when the queue becomes empty, which is the exhaustive case.

  ▶ Now, on return from vacation, the server will go for another vacation or possibly more vacation if required if the system was empty on return from a vacation; that means, what? We have a multiple vacations case.

  So, this is basically $M/M/1 - V_m$ model. So, basically, we have here $M/M/1 - V_m$ is the model that we have here, in that notation that we have just introduced here. Now, this is what the model that we are considering. Now, as you see, this model can also be thought about as a model connected with setup in the in a machine production system context. So, that is what we are saying.

- This multiple vacation $M/M/1$ model can also be thought about as an $M/M/1$ model with a setup time that is a machine with setup.

  How? As soon as you know, the server becomes free, or the machine becomes free of any customer, in this case, jobs, then the machine is switched off.

  ♦ That is what we call the server is deactivated as soon as the queue becomes empty. Now, when the new customer arrives at an empty system, a setup process needs to be initiated, which starts the server, and it takes an exponential amount of time with rate θ is what. So, this model is exactly same as the model for this situation.

  This is where this N-policy business might come here because you do not want to start as soon as one customer comes. You want to wait because there is a setup cost and setup duration, and so on. So, you would want to wait till certain number of customers accumulate and then start the machine; that is a normal thing to do in such situations. So, that is what this situation is. So, the N-policy is just extension in this context; one can make it, but here as soon as the one customer comes, the setting up of server starts, and it takes an exponential amount of time with parameter θ. After that, the server starts serving the customer again. This is same as an $M/M/1$ working with a multiple vacation concept; this model is what is here.

- Now, we will assume that inter-arrival time service times, vacation times, or set up times, whichever it is they are all mutually independent.

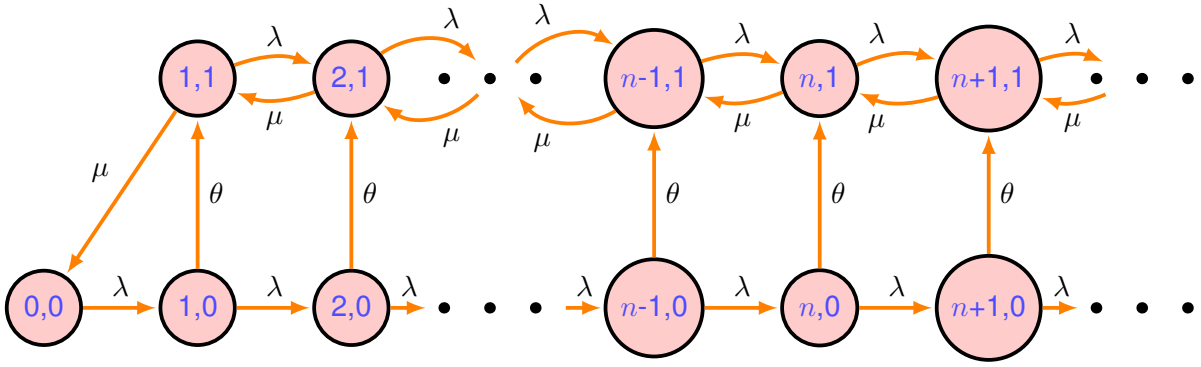
So, usual $M/M/1$, we have introduced a multiple vacation idea here. That is what we have here.

- Now, this model can be represented as a continuous-time Markov chain, where the state of the system at time t is characterized by this two-dimensional quantity $\{I(t), J(t)\}$, where $I(t)$ denotes the number of customers in

the system and $J(t)$ denotes the state of the server. 0 means server is in vacation; 1 means the server is in busy period, meaning it is in working condition or normal period.

- It is clear that the state space of the two-dimensional CTMC is $S = (0,0) \cup \{(i,j) \mid i \geq 1, j = 0,1\}$, where state $(i,0), i \geq 0$, indicates that the server is in the vacation period and there are i customers, and state $(i,1), i \geq 1$ indicates that the server is in the busy period and there are i customers.
- Let $p(i,j)$ denote the steady state probability of the system being in state (i,j).

Now, let us look at how we can model this by a CTMC and what are the state transition diagrams.



So, this is the state transition diagram in the case of the $M/M/1$ queue with multiple vacations.

- We have following stochastic flow balance equations for this model.

$$\begin{aligned}
\lambda p(0,0) &= \mu p(1,1) \\
(\mu+\lambda)p(1,1) &= \theta p(1,0) + \mu p(2,1) \\
(\lambda+\theta)p(i,0) &= \lambda p(i-1,0), \qquad i \geq 1 \\
(\lambda+\mu)p(i,1) &= \lambda p(i-1,1) + \mu p(i+1,1) + \theta p(i,0), \qquad i \geq 2
\end{aligned}$$

So, this is what is the steady-state balance equations. Now, you can solve this, again; it is two-dimensional solving will be little difficult, but it is solvable one can obtain this like what we have done for retrial queue and so on in a similar fashion one can do this.

- We will now try to solve the balance equations by the matrix-geometric method, thereby exhibiting the essence of this approach. Note that here the transitions happen to adjacent levels only.
- The last two equations can be rewritten in vector-matrix notation as

$$p_{i-1}A_0 + p_iA_1 + p_{i+1}A_2 = 0, \quad i \geq 2$$

where $p_i = (p(i,0), p(i,1))$ and

$$A_0 = \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix}, A_1 = \begin{pmatrix} -(\lambda+\theta) & \theta \\ 0 & -(\lambda+\mu) \end{pmatrix}, A_2 = \begin{pmatrix} 0 & 0 \\ 0 & \mu \end{pmatrix}.$$

So, this is the vector-matrix equation in this case. In matrix-geometric method, any such equation is what you are trying to get. You are trying to put it in such a form that you can look at the similarity of this equation with an $M/M/1$

system. In $M/M/1$ system, assume that suppose if this was scalars you had, p_{i-1}, p_i and p_{i+1} is what was related, but in a scalar fashion, but here in a vector fashion or matrix notation. With this (A) coefficient of matrix and these (p) as vectors, $p_{i-1}A_0 + p_iA_1 + p_{i+1}A_2 = 0$ relationship is true. This is what is the similarity that when you make use of it when you try to use this matrix-geometric method. Geometric method or geometric probability distribution is what you would obtain in $M/M/1$ case. So, this is matrix version of that is what you are looking at here in a way. So, this is what is the equation.

- We will first simplify the vector-matrix equation above by eliminating p_{i+1}.
- By equating the flow from level i to level $i+1$ to the flow from level $i+1$ to i, we obtain

$$(p(i,0) + p(i,1))\lambda = p(i+1,1)\mu \quad \Rightarrow \quad p_iA_3 = p_{i+1}A_2, \quad \text{where } A_3 = \begin{pmatrix} 0 & \lambda \\ 0 & \lambda \end{pmatrix}.$$

- Substituting this into the vector-matrix equation gives

$$p_{i-1}A_0 + p_i(A_1 + A_3) = 0, \quad i \geq 2,$$

  or

$$p_i = -p_{i-1}A_0(A_1 + A_3)^{-1} = p_{i-1}R,$$

  where

$$R = -A_0(A_1 + A_3)^{-1} = \begin{pmatrix} \lambda/(\lambda+\theta) & \lambda/\mu \\ 0 & \lambda/\mu \end{pmatrix}.$$

- Iterating leads to

$$p_i = p_1R^{i-1}, \quad i \geq 1.$$

  `[Note the similarity with M/M/1]`

- Finally $p(0,0)$ and p_1 follow from the remaining balance equations and the normalization condition

$$1 = \sum_{i,j} p(i,j) = p(0,0) + p_1(I-R)^{-1}\mathbf{e},$$

  where I is the identity matrix and $\mathbf{e}$ is the column vector of ones.

- We can obtain the mean number of customers in the system as

$$L = \sum_i ip_i\mathbf{e} = \sum_{i=1}^{\infty} ip_1R^{i-1}\mathbf{e} = p_1(I-R)^{-2}\mathbf{e}.$$

  And, from Little's law, we can obtain W as $W = L/\lambda$.

  This is what is the essence of matrix-geometric. In this way, you can solve this system directly also after a lengthy process; you can also solve. There is no problem with that. You can use operator method or generative function method; you will be able to solve this model as well.

- From mean-value arguments, and using PASTA and Little's law, we can obtain W directly as

$$W = \frac{1/\mu}{1-\rho} + \frac{1}{\theta}.$$

That is, the mean vacation time is exactly the extra mean delay caused by the vacation.

► In fact, it can be shown (by using, e.g., a sample path argument) that the extra delay is an exponential time with parameter θ.

So, this is what is the $M/M/1$ model with multiple vacations. Now, this model can be analyzed with single vacation as well. So, what one has to do? You have to make a slight change to the model setup the server after returning from a single vacation, either start service if the system is nonempty at the point of time or waits for a customer to arrive like in the normal queue. So, what one has to do is that one more state, for example, here what would happen here, now here there is no point in multiple vacation case vacation will not end while the system is empty, because whenever on return from a vacation if the server finds the system empty he will take another vacation. But, in the case of single vacation case, like he can, he will end his vacation, and it will bring down to the normal mode. So, there will be another state $(0,1)$ that you would introduce here, and there will be a moment θ here, and from here, there is a λ here it will come here. These are the only two extra links that you need to add to this diagram to get a model, which is an $M/M/1$ with single vacation, and one can also analyze along similar vacation. And that is the reason why we did not call this as simply as 0. $(0,0)$; we kept it because in single vacation case, we could have added another state $(0,1)$. In that case, vacation will end while the system is empty, and the arrival can happen; that is only two links that you need to establish, which means that there will be a curve here, and then there is an arrival. So, this is what would be the structure that you will have with appropriate rates, then you will get a system of equations, and then you will try to solve for the case of this $M/M/1$ queue single vacation it is in a similar way. So, one need not do that is what they need. So, this is what we do for the $M/M/1$ with vacations in this particular case. So, what we have seen is how the vacation comes and when in this vacation is introduced in a simple $M/M/1$ model, how one can analyze. And we have also given the essence of a matrix-geometric approach one can adopt to solve such problems when you have this kind of multi-dimensional situation when clubbing one. Basically, you are condensing one dimension to make it look like a single dimension, but now not in a scalar terms, but in vector terms. That will give rise to the applicability of matrix-geometric, which is a very powerful tool that the essence you can see from this analysis. So, we will stop here we will continue in the next lecture.

Thank you, bye.