Sirindhorn International Institute of Technology

Thammasat University

School of Information, Computer and Communication Technology

ECS 455: Mobile Communications
Call Blocking Probability

Prapun Suksompong, Ph.D.
prapun@siit.tu.ac.th

December 23, 2012

In this note, we look at derivations of the Erlang B formula and the Engsett formula, both of which estimate the call blocking probability when trunking is used. To do this, we need to borrow some concepts from queueing theory. Moreover, some basic analysis of stochastic processes including Poisson processes and Markov chains is needed. For completeness, working knowledge on these processes is summarized here as well. However, we do assume that the readers are familiar with concepts from basic probability course.

Contents

In performance evaluation of cellular systems or telephone networks, ***Erlang B formula*** [1, p 23], to be revisited in Section 2, is a formula for estimating the call blocking probability for a cell (or a sector, if sectoring is used) which has m "trunked" channels and the amount of ("offered") traffic is A Erlang:

$$P_b = \frac{\frac{A^m}{m!}}{\sum\limits_{i=0}^{m} \frac{A^i}{i!}}. \tag{1}$$

It is directly used to determine the probability P_b that call requests will be blocked by the system because all channels are currently used. The amount of traffic (A) can be found by the product of the total call request rate λ and the average call duration ($1/\mu$).

When we design a cellular system, the blocking probability P_b should be less than some pre-determined value. In which case, the function above can be used to suggest the minimum number of channels per cell (or sector). If we already know the number of channels per cell (or sector) of the system, the (inverse of this) function can also be used to determined how many users the system can support.

The Erlang B formula (1) is derived under the following M/M/m/m assumptions:

(a) Blocked calls cleared

- No queuing for call requests.
- For every user who requests service, there is no setup time and the user is given immediate access to a channel if one is available.
- If no channels are available, the requesting user is blocked without access and is free to try again later.

(b) Calls arrive as determined by a ***Poisson process***.

(c) Arrivals of requests are memoryless: all users, including blocked users, may request a channel at any time.

(d) There are an infinite number of users (with finite overall request rate).

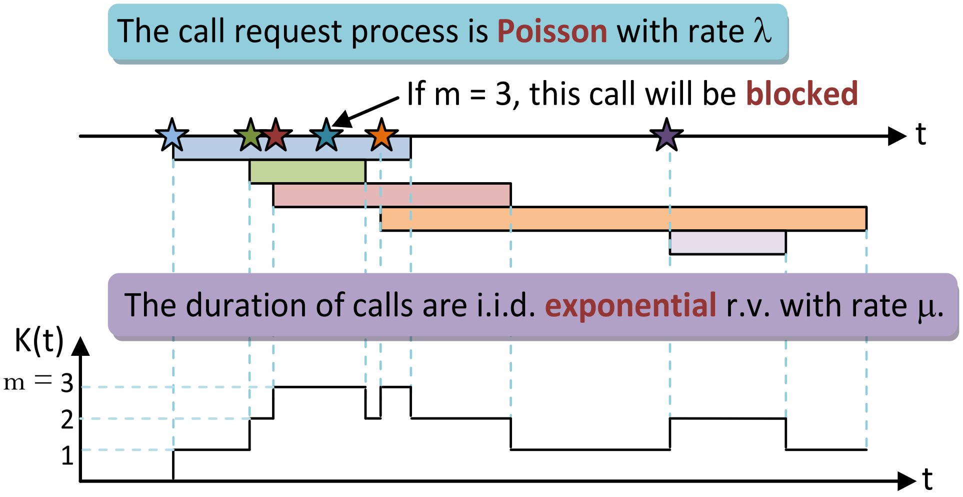
- The finite user results always predict a smaller likelihood of blocking. So, assuming infinite number of users provides a conservative estimate.

(e) The duration of the time that a user occupies a channel is ***exponentially*** distributed, so that longer calls are less likely to occur.

(f) There are m channels available in the trunking pool.

- For us, m = the number of channels for a cell or for a sector.

Some of these conditions are captured by Figure 1.



The call request process is Poisson with rate λ
If m = 3, this call will be blocked
t
The duration of calls are i.i.d. exponential r.v. with rate μ.
K(t)
m = 3
2
1
t

Figure 1: M/M/m/m Assumption

In this note, we will look more closely at these assumptions and see how they lead to the Erlang B formula. The goal is not limited to simply understanding of the derivation of the formula itself but, later on, we also want to try to get a new formula that relaxes some of the assumptions above to make the analysis more realistic.

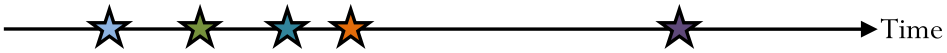
In Figure 1, we also show one important parameter of the system: $K(t)$. This is the number of used channels at time t. When $K(t) < m$, new call request can be serviced. When $K(t) = m$, new call request(s) will be blocked. So, we can find the call blocking probability by looking at the value of $K(t)$. In particular, we want to find out the proportion of time the system has $K = m$. This key idea will be revisited again in Section 2.

As seen in the assumptions, understanding of the Poisson process is important for the derivation of the Erlang B formula. Therefore, this process and some probability concepts will be reviewed in Section 1. Most of the probability review will be put in footnotes so that they do not interfere with the flow of the presentation.

1 Poisson Processes

In this section, we consider an important random process called ***Poisson process*** (PP). This process is a popular model for customer arrivals or calls requested to telephone systems.

1.1. We start by picturing a Poisson Process as a random arrangement of “marks” (denoted by “x” or) on the time axis. These marks usually indicate the time that customers arrive in queueing models. In the language of “queueing theory”, the marks denote ***arrival times***. For us, they indicate the time that call requests are made:



1.2. In this class, we will focus on one kind of Poisson process called ***homogeneous***[1] ***Poisson process***. Therefore, from now on, when we say “Poisson process”, what we mean is “homogeneous Poisson process”.

1.3. The first property that you should remember is that

> there is only one parameter for Poisson process.

This parameter is the ***rate*** or ***intensity*** of arrivals (the average number of arrivals per unit time. We use λ to denote[2] this parameter.

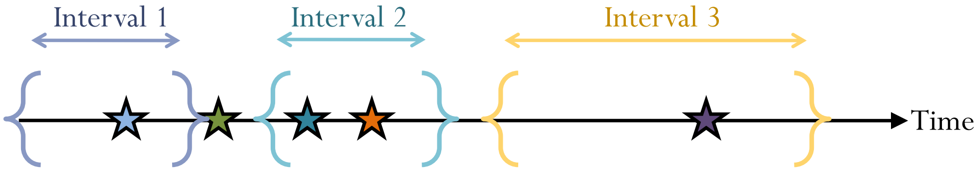
1.4. How can λ, which is the only parameter, controls Poisson process?

The key idea is that the Poisson process is as random/unstructured as a process can be.

[1] This is a special case of Poisson processes. More general Poisson process (which are called non-homogenous Poisson processes) would allow the rate to be time-dependent. Homogeneous Poisson processes have constant rates.

[2] For homogeneous Poisson process, λ is a constant. For non-homogeneous Poisson process, λ is a function of time, say $\lambda(t)$. Our λ is constant because we focus on homogeneous Poisson process

Therefore, if we consider many non-overlapping intervals on the time axis, say interval 1, interval 2, and interval 3 below,



and count the number of arrivals N_1, N_2 and N_3 in these intervals[3]. Then, the numbers N_1, N_2 and N_3 in our example above should be independent[4]; for example, knowing the value of N_1 does not tell us anything at all about what values of N_2 and N_3 will be. This is what we are going to take as a vague definition of the "complete randomness" of the Poisson process.

To summarize, now we have one more property of a Poisson process:

> The number of arrivals in non-overlapping intervals are independent.

[3] Note that the numbers N_1, N_2, and N_3 are random. Because they are counting the number of arrivals, we know that they can be any non-negative integers:

$$0, 1, 2, 3, \dots.$$

Because we don't know their exact values, we describe them via the likelihood or probability that they will take one of these possible values. For example, for N_1, we describe it by

$$P\left[N_1 = 0\right], P\left[N_1 = 1\right], P\left[N_1 = 2\right], \dots$$

where $P\left[N_1 = k\right]$ is the probability that N_1 takes the value k. Such list of numbers is a bit tedious. So, we define a function

$$p_{N_1}(k) = P\left[N_1 = k\right].$$

This function $p_{N_1}(\cdot)$ tells the probability that N_1 will take a particular value (k). We call p_{N_1} the probability mass function (pmf) of N_1. At this point, we don't know much about $p_{N_1}(k)$ except that its values will be between 0 and 1 and that

$$\sum_{k=0}^{\infty} p_{N_1}(k) = 1.$$

These two properties are the necessary and sufficient conditions for any pmf.

[4] By saying that something are independent, we mean it in terms of probability. In particular, when we say that N_1 and N_2 are independent, it means that

$$P\left[N_1 = k \text{ and } N_2 = m\right]$$

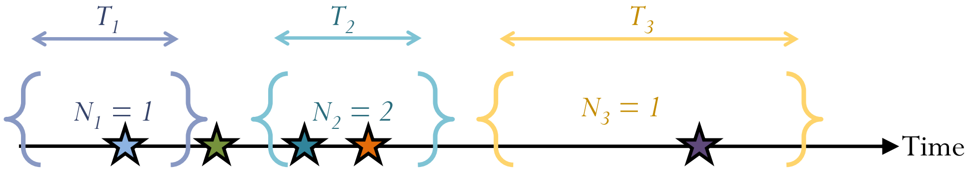
(which is the probability that $N_1 = k$ and $N_2 = m$) can be written as the product

$$p_{N_1}(k) \times p_{N_2}(k).$$

1.5. Do we know anything else about N_1, N_2, and N_3? Again, we have only one parameter λ for a Poisson process. So, can we connect λ with N_1, N_2, and N_3?

Recall that λ is the average number of arrivals per unit time. So, if $\lambda = 5$ arrivals/hour, then we expect that N_1, N_2, and N_3 should conform with this λ, statistically.

Let's first be more specific about the time durations of the intervals that we have earlier. Suppose their lengths are T_1, T_2, and T_3 respectively.



Then, you should expect[5] that

$$\begin{aligned} \mathbb{E}N_1 &= \lambda T_1, \\ \mathbb{E}N_2 &= \lambda T_2, \text{ and} \\ \mathbb{E}N_3 &= \lambda T_3. \end{aligned}$$

For example, suppose $\lambda = 5$ arrivals/hour and $T_1 = 2$ hour. Then you would see about $\lambda \times T_1 = 10$ arrivals during the first interval. Of course, the number of arrivals is random. So, this number 10 is an average or the expected number, not the actual value.

To summarize, we now know one more property of a Poisson process:

> For any interval of length T, the expected number of arrivals in this interval is given by
>
> $$\mathbb{E}N = \lambda T. \tag{2}$$

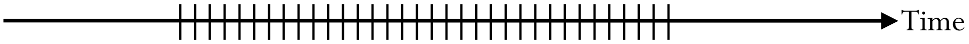
[5] Recall that $\mathbb{E}N_1$ is the expectation (average) of the random variable N_1. Here, the random variables are discrete. Therefore, formula-wise, we can calculate $\mathbb{E}N_1$ from

$$\mathbb{E}N_1 = \sum_{k=0}^{\infty} k \times P\left[N_1 = k\right];$$

that is the sum of the possible values of N_1 weighted by the corresponding probabilities

1.1 Discrete-time (small-slot) approximation of a Poisson process

1.6. The next key idea is to consider a small interval:
Imagine dividing a time interval of length T into n equal slots.



Then each slot would be a time interval of duration $\delta = T/n$. For example, if $T = 20$ hours and $n = 10,000$, then each slot would have length

$$\delta = \frac{T}{n} = \frac{20}{10,000} = 0.002 \text{ hour.}$$

Why do we consider small interval? The key idea is that as the interval becomes very small, then it is extremely unlikely that there will be more than 1 arrivals during this small amount of time. This statement becomes more accurate as we increase the value of n which decreases the length of each interval even further. What we are doing here is an approximation[6] of a continuous-time process by a discrete-time process.[7]

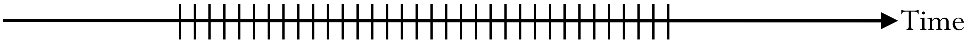
To summarize, we will consider the discrete-time approximation of the (continuous-time) Poisson process. In such approximation, the time axis is divided into many small time intervals (which we call "slots").

> When the interval is small enough, we can assume that at most 1 arrival occurs.

[6] You also do this when you plot a graph of any function $f(x)$. You divide the $x-$axis by many (equally spaced) values of x and then evaluate the values of the function at these values of x. You need to make sure that the values of x used are "dense" enough such that no surprising change in the function f is overlooked.

[7] If we wnat to be rigorous, we would have to bound the error from such approximation and show that the error disappear as $n \to \infty$. We will not do that here.

1.7. Let's look at the small slots more closely. Here, we let N_1 be the number of arrivals in slot 1, N_2 be the number of arrivals in slot 2, N_3 be the number of arrivals in slot 3, and so on as shown below.



Then, these N_i's are all Bernoulli random variables because they can only take the values 0 or 1. In which case, for their pmfs, we only need to specify one value $P[N_i = 1]$. Of course, knowing this, we can calculate $P[N_i = 0]$ by $P[N_i = 0] = 1 - P[N_i = 1]$.

Recall that the average $\mathbb{E}X$ of any Bernoulli random variable X is simply $P[X = 1]$.[8] So, if we know $\mathbb{E}X$ for Bernoulli random variable, then we know right away that $P[X = 1] = \mathbb{E}X$ and $P[X = 0] = 1 - \mathbb{E}X$.

Now, it's time to use what we learned about Poisson process. The slots that we consider before are of length $\delta = T/n$. So, the random variables $N_1, N_2, N_3, \ldots$ share the same expected value

$$\mathbb{E}N_1 = \mathbb{E}N_2 = \mathbb{E}N_3 = \cdots = \lambda\delta.$$

For example, with $\lambda = 5$, $T = 20$, and $n = 10,000$, the expected number of arrivals in a slot is $\lambda\delta = \lambda\frac{T}{n} = 0.01$ arrivals.

Because these N_i's are all Bernoulli random variables and because they share the same expected value, we can conclude that they are identically distributed; that is their pmf's are all the same. Furthermore, because the slots do not overlap, we also know that the N_i's are independent. Therefore,

> for small non-overlapping slots, the corresponding number of arrivals N_i's are i.i.d. Bernoulli random variables whose pmf's are given by
>
> $$p_1 = P[N_i = 1] = \lambda\delta \quad \text{and} \quad p_0 = P[N_i = 0] = 1 - \lambda\delta,$$
>
> where δ is the length of each slot.

[8] For Bernoulli random variable X, the average is

$$\mathbb{E}X = 0 \times P[X = 0] + 1 \times P[X = 1] = P[X = 1].$$

For conciseness, we usually let $p_0 = P[X = 0]$ and $p_1 = P[X = 1]$. Hence, $\mathbb{E}X = p_1$.

1.8. Discrete-time approximation provides a method for `MATLAB` to generate a Poisson process with arrival rate λ. Here are the steps:

(a) First, we fix the length T of the whole simulation. (For example, $T = 20$ hours.)

(b) Then, we divide T into n slots. (For example, $n = 10,000$.)

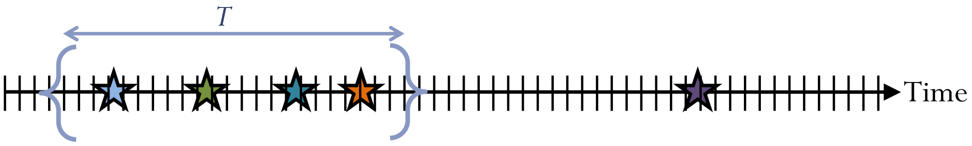
(c) For each slot, only two cases can happen: 1 arrival or no arrival. So, we generate Bernoulli random variable for each slot with $p_1 = \lambda \times T/n$. (For example, if $\lambda = 5$ arrival/hr, then $p_1 = 0.01$.)

To do this for n slots, we can use the command `rand(1,n) < p1` or `binornd(1,p1,1,n)`.

1.9. Note that what we have just generated is exactly ***Bernoulli trials*** whose success probability for each trial is $p_1 = \lambda\delta$. In other words, a Poisson process can be approximated by Bernoulli trials with success probability $p_1 = \lambda\delta$.

1.2 Properties of Poisson Processes

1.10. What we want to do next is to revisit the description of the number of arrivals in a time interval. Now, we will NOT assume that length of the time interval is short. In particular, let's reconsider an interval of length T below.



Let N be the number of arrivals during this time interval. In the picture above, $N = 4$.

Again, we will start with a discrete-time approximation; we divide T into n small slots of length $\delta = \frac{T}{n}$. In the previous subsection, we know that the number of arrivals in these intervals, denoted by $N_1, N_2, \dots, N_n$ can be well-approximated by i.i.d. Bernoulli with probability of having exactly one arrival $= \lambda\delta$. (Of course, we need δ to be small for the approximation to be precise.) The total number of arrivals during the original interval of length T can be found by summing the values of the N_i's:

(3)

You may recall, from introductory probability class, that

(a) summation of n Bernoulli random variables with success probability p gives a binomial(n, p) random variable[9]

and that

(b) a binomial(n, p) random variable whose n is large and p is small can be well approximated by a Poisson random variable with parameter $\alpha = np$ [10].

Therefore, the pmf of the random variable N in (3) can be approximated by a Poisson pmf whose parameter is

$$\alpha = np_1 = n\lambda \frac{T}{n} = \lambda T.$$

This approximation[11] gets more precise when n is large (δ is small). In

[9] X is a ***binomial*** random variable with size $n \in \mathbb{N}$ and parameter $p \in (0, 1)$ if

$$p_X(x) = \begin{cases} \binom{n}{x} p^x (1-p)^{n-x}, & x \in \{0, 1, 2, \ldots, n\} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

We write $X \sim \mathcal{B}(n, p)$ or $X \sim$ binomial(p). Observe that $\mathcal{B}(1, p)$ is Bernoulli with parameter p. Note also that $\mathbb{E}X = np$. Another important observation is that such random variable is simply a sum of n independent and identically distributed (i.i.d.) Bernoulli random variables with parameter p.

[10] X is a ***Poisson*** random variable with ***parameter*** $\alpha > 0$ if

$$p_X(x) = \begin{cases} e^{-\alpha} \frac{\alpha^x}{x!}, & x \in \{0, 1, 2, \ldots\} \\ 0, & \text{otherwise} \end{cases}$$

We write $X \sim \mathcal{P}(\alpha)$ or Poisson(α). Note also that $\mathbb{E}X = \alpha$.

[11] In introductory probability class, you may have seen pointwise convergence in terms of the pmfs. An alternative view on this convergence is to look at the characteristic function. Recall that the characteristic function $\varphi_X(u)$ of a random variable X is given by $\mathbb{E}\left[e^{jXu}\right]$. In particular, for Bernoulli random variable with parameter p, the characteristic function is $1 - p + pe^{ju}$. One important property of the characteristic function is that the characteristic function of a sum of independent random variables is simply the product of the characteristic functions of the individual random variables. Hence, the characteristic function of the binomial random variable with parameter pair (n, p) is simply $\left(1 - p + pe^{ju}\right)^n$ because it is a sum of n independent Bernoulli random variables. For us, the parameter p is $\lambda T/n$. So, the characteristic function of the binomial is

$$\left(1 - \frac{1}{n}\left(\lambda T + \lambda T e^{ju}\right)\right)^n.$$

As $n \to \infty$, the binomial characteristic function converges to $\exp\left(-\lambda T + \lambda T e^{ju}\right)$ which turns out to be the characteristic function of the Poisson random variable whose parameter is $\alpha = \lambda T$.

fact, in the limit as $n \to \infty$ (and hence $\delta \to 0$), the random variable N is governed by $\mathcal{P}(\lambda T)$. Recall that the expected value of $\mathcal{P}(\alpha)$ is α. Therefore, λT is the expected value of N. This agrees with what we have discussed before in (2).

In conclusion,

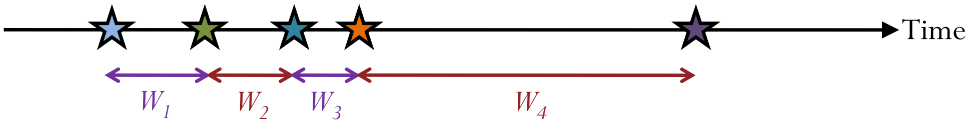
> the number N of arrivals in an interval of length T is a Poisson random variable with mean (parameter) λT

1.11. Now, to sum up what we have learned so far, the following is one of the two main properties of a Poisson process

> The number of arrivals $N_1, N_2, N_3, \ldots$ during non-overlapping time intervals are independent Poisson random variables with mean $\lambda\times$ the length of the corresponding interval.

1.12. Another main property of the Poisson process, which provides another method for simulation of Poisson process, is that

> The lengths of time between adjacent arrivals $W_1, W_2, W_3, \ldots$ are i.i.d. exponential[12] random variables with mean $1/\lambda$.



The exponential part of this property can be seen easily by considering the complementary cumulative distribution function (CCDF) of the W_k's.

[12] The ***exponential*** distribution is denoted by $\mathcal{E}(\lambda)$. An exponential random variable X is characterized by its probability density function

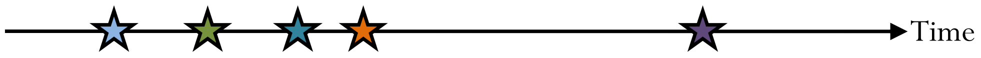
$$f_X(x) = \begin{cases} \lambda e^{-\lambda x}, & x > 0, \\ 0, & x \le 0. \end{cases}$$

Note that $\mathbb{E}X = \frac{1}{\lambda}$. The cumulative distribution function (CDF) of X is given by

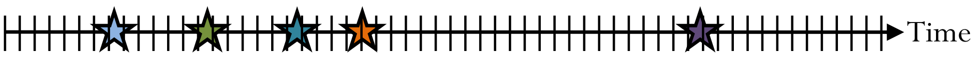
$$F_X(x) \equiv P[X \le x] = \begin{cases} 1 - e^{-\lambda x}, & x > 0, \\ 0, & x \le 0. \end{cases}$$

Often, we also talked about the complementary cumulative distribution function (CCDF) which, for exponential random variable, is given by

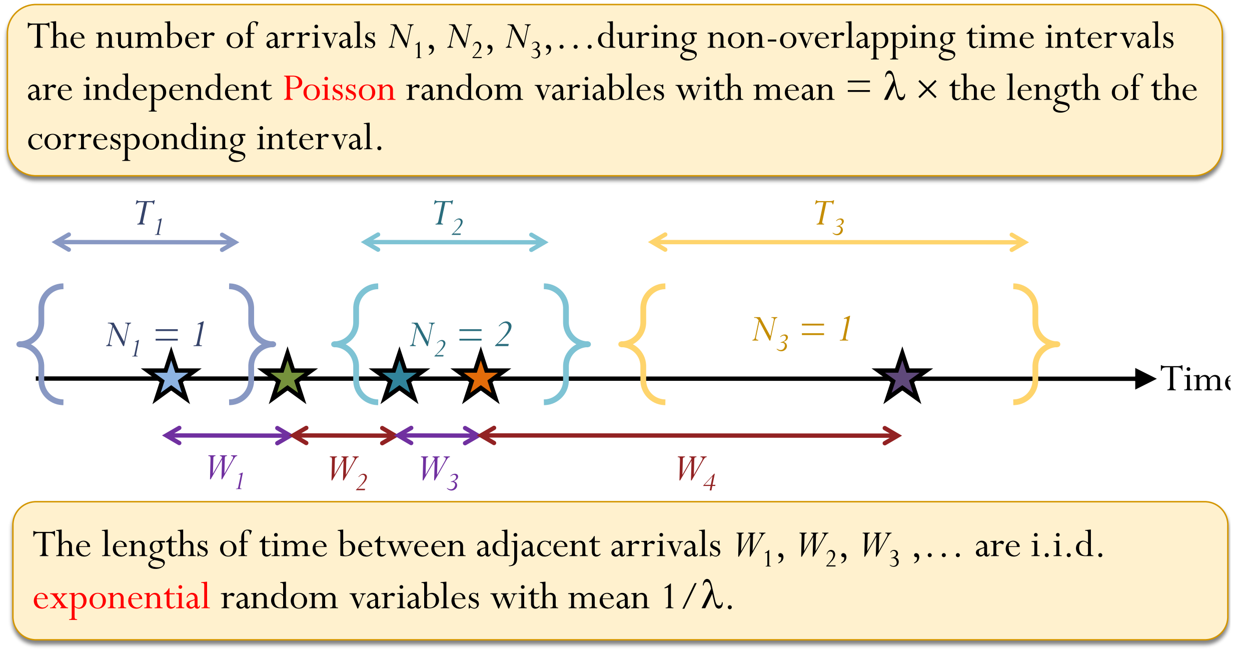
$$P[X > x] \equiv 1 - F_X(x) = \begin{cases} e^{-\lambda x}, & x > 0, \\ 1, & x \le 0. \end{cases}$$



Alternatively, this property can be derived by looking at the discrete-time approximation of the Poisson process. In the discrete-time version, the time until the next arrival is geometric[13]. In the limit, the geometric random variable becomes exponential random variable.



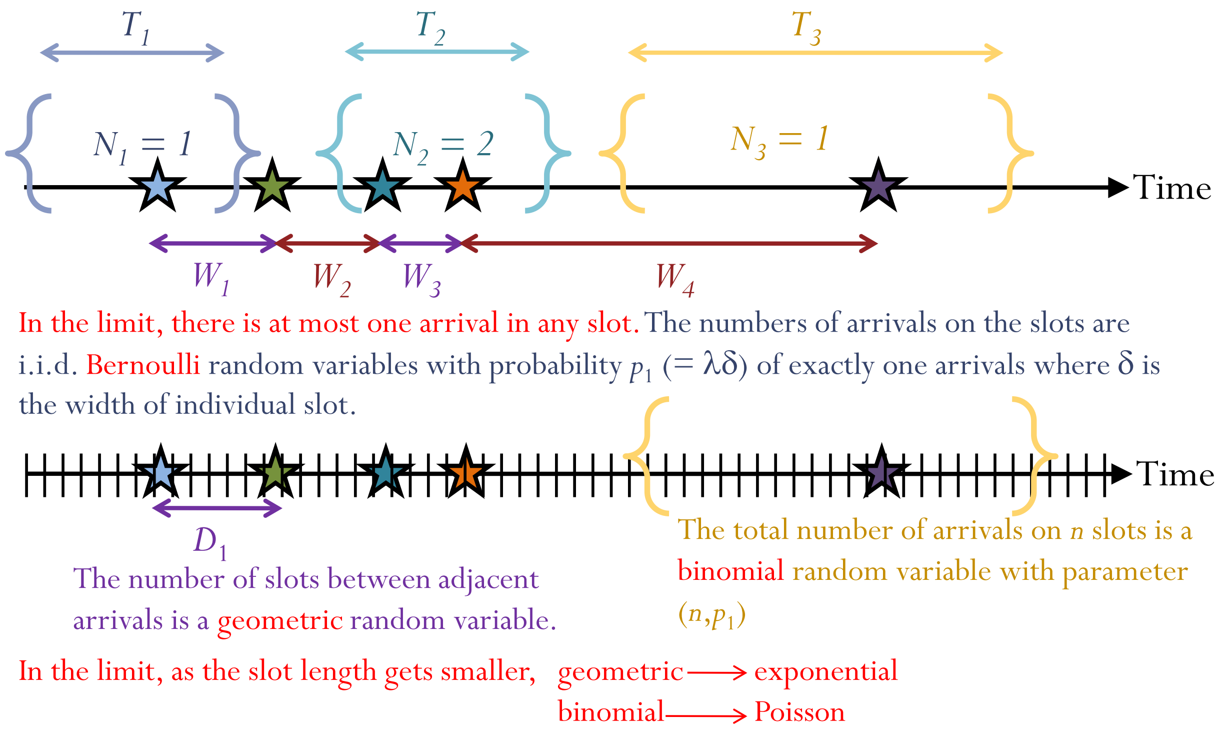
Both main properties of Poisson process are shown in Figure 2. The small-slot analysis (discrete-time approximation), which can be used to prove the two main properties, is shown in Figure 3.


The number of arrivals N_1, N_2, N_3,…during non-overlapping time intervals are independent Poisson random variables with mean = $\lambda \times$ the length of the corresponding interval.

The lengths of time between adjacent arrivals W_1, W_2, W_3 ,… are i.i.d. exponential random variables with mean $1/\lambda$.

Figure 2: Two main properties of a Poisson process

[13] You may recall from your introductory probability class that, in Bernoulli trials, the number of trials until the next success is geometric.



In the limit, there is at most one arrival in any slot. The numbers of arrivals on the slots are i.i.d. Bernoulli random variables with probability p_1 ($= \lambda\delta$) of exactly one arrivals where δ is the width of individual slot.

The number of slots between adjacent arrivals is a geometric random variable.

The total number of arrivals on n slots is a binomial random variable with parameter (n,p_1)

In the limit, as the slot length gets smaller, geometric ⟶ exponential
binomial ⟶ Poisson

Figure 3: Small slot analysis (discrete-time approximation) of a Poisson process

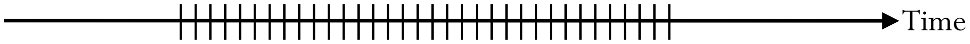
2 Derivation of the Markov chain for Erlang B Formula

In this section, we combine what we know about Poisson process discussed in the previous section with the assumption on the call duration. The goal is to characterize how the random quantity K, eluded to before we begin Section 1, is evolved as a function of time. The key idea is again to use the small-slot (discrete-time) approximation.

2.1. Recall that, for the Erlang B formula, we assume that there are m channels available in the trunking pool. Therefore, the probability P_b that a call requested by a user will be blocked is given by the probability that none of the m channels are free when a call is requested.

We will consider the long-term behavior of this system, i.e. the system is assumed to have been operating for a long time. In which case, at the instant that somebody is trying to make a call, we don't know how many of the channels are currently free.

2.2. Let's first divide the time into small slots (of the same length δ) as we have done in the previous section.



Then, consider any particular slot. Suppose that at the beginning of this time slot, there are K channels that are currently used.[14] We want to find out how this number K changes as we move forward one slot time. This random variable K will be call the ***state*** of the system[15]. The system moves from one state to another one as we advance one time slot.

Example 2.3. Suppose there are 5 persons using the channels at the beginning of a slot. Then, $K = 5$.

(a) Suppose that, by the end of this slot, none of these 5 persons finish their calls.

(b) Suppose also that there is one new person who wants to make a call at some moment of time during this slot.

Then, at the end of this time slot, the number of channels that are occupied becomes

So, the state K of the system changes from 5 to 6 when we reach the end of the slot, which can now be regarded as the beginning of the next slot.

2.4. Our current intermediate goal is to study how the state K evolves from one slot to the next slot. Note that it might be helpful to label the state K as K_1 (or $K[1]$) for the first slot, K_2 (or $K[2]$) for the second slot, and so on.

As shown in Example 2.3, to determine how the K_i's progress from K_1 to K_2, K_2 to K_3, and so on, we need to know two pieces of information:

[14] The value of K can be any integer from 0 to m.

[15] This is the same "state" concept that you have studied in digital circuits class.

Q1 How many calls (that are ongoing at the beginning of the slot under consideration) ***end*** during the slot that we are considering?

Q2 How many ***new*** call requests are made during the slot under consideration?

Note that Q1 depends on the characteristics of the call duration and Q2 depends on the characteristics of the call request/arrival process. After we know the answers to these two questions, then we can find K_i via

$$K_{i+1} = K_i - \underbrace{(\text{\# old call ends})}_{\text{Q1}} + \underbrace{(\text{\# new call requests})}_{\text{Q2}}$$

2.5. Q2 is easy to answer.
A2: If the interval is small enough (δ is small), then there can be at most one new arrival (new call request) which occurs with probability

$$p_1 = \lambda\delta.$$

2.6. For Q1, we need to consider the ***call duration model***. The M/M/m/m assumption states that the call duration[16] is exponentially distributed with parameter (rate) μ. Let's consider the call duration D of a particular call.

Recall that the probability density function (pdf) of an exponential random variable X with parameter μ is given by

$$f_X(x) = \begin{cases} \mu e^{-\mu x}, & x > 0, \\ 0, & x \leq 0, \end{cases}$$

and the average (or expected value) is given by

$$\mathbb{E}[X] = \int_0^\infty x f_X(x) dx = \frac{1}{\mu}.$$

You may remember that in the Erlang B formula, we assume that the average call duration is $\mathbb{E}[D] = H = \frac{1}{\mu}$.

[16] In queueing theory, this is sometimes called the ***service time***. The parameter μ then captures the service rate.

An important property of an exponential random variable X is its memoryless property[17]:

$$P[X > x + \delta | X > x] = P[X > \delta].$$

For example,

$$P[X > 7 | X > 5] = P[X > 2].$$

What does this memoryless property mean? Suppose you have a lightbulb and you have used it for 5 years already. Let it's lifetime be X. Then, of course, X is a random variable. You know that $X > 5$ because it is still working now. The conditional probability $P[X > 7|X > 5]$ is the probability that it will still work after two more years of use (given the fact that currently it has been working for five years). Now, if X is an exponential random variable, then the memoryless property says that $P[X > 7|X > 5] = P[X > 2]$. In other words, the probability that you can use it for at least two more years is exactly the same as the probability that you can use a new lightbulb for more than two years. So, your old lightbulb essentially forgets that it has been used for 5 years, It always performs like a new lightbulb (statistically). This is we we mean by the memoryless property of an exponential random variable.

2.7. To answer Q1, we will now return to our small slot approximation. Again, consider one particular slot. At the beginning of our slot, there are $K = k$ ongoing calls. The probability that a particular call, which is still ongoing at the beginning of this slot, will be unfinished by the end of this slot is $e^{-\mu\delta}$.

[17] To see this, first recall the definition of conditional probability:$P(A|B) = \frac{P(A \cap B)}{P(B)}$. Therefore,

$$P[X > x + \delta | X > x] = \frac{P[X > x + \delta \text{ and } X > x]}{P[X > x]} = \frac{P[X > x + \delta]}{P[X > x]}.$$

Now, $P[X > x] = \int\limits_x^\infty \mu e^{-\mu x} dx = e^{-\mu x}$. Hence,

$$P[X > x + \delta | X > x] = \frac{e^{-\mu(x+\delta)}}{e^{-\mu x}} = e^{-\mu\delta} = P[X > \delta].$$

To see this, consider a particular call. Suppose the duration of this call is D. By assumption, we know that D is exponential with parameter μ.

Let s be the length of time from the call initiation time to the beginning of our slot. Note that the call is still ongoing. Therefore $D > s$. Now, to say that this call will be unfinished by the end of our slot is equivalent to requiring that $D > s + \delta$. By the memoryless property, we have

$$P\left[D > s + \delta \,|D > s\right] = P\left[D > \delta\right] = e^{-\mu\delta}.$$

Recall that we have $K = k$ ongoing calls at the beginning of our slot. So, by the end of our slot, the probability that none of them finishes is

$$\left(e^{-\mu\delta}\right)^k = e^{-k\mu\delta}.$$

The probability that exactly one of them finishes is

Now, note that $e^x \approx 1 + x$ for small x. Therefore, A1:

(a) the probability that none of the $K = k$ calls ends during our slot is

(b) the probability that exactly one of them ends during our slot is

Magically, these two probabilities sum to one. So, we don't have to consider other events/cases.

2.8. Summary:

(a) Call generation/request/initiation process:

(b) Call duration process:

So, after one (small) slot, there can be four events:

The corresponding probability for each case is

Therefore, if we have $K = k$ at the beginning of our time slot, the at the end of our slot, K may

(a) remain unchanged with probability $1 - \lambda\delta - k\mu\delta$, or

(b) decrease by 1 with probability $k\mu\delta$, or

(c) increase by 1 with probability $\lambda\delta$.

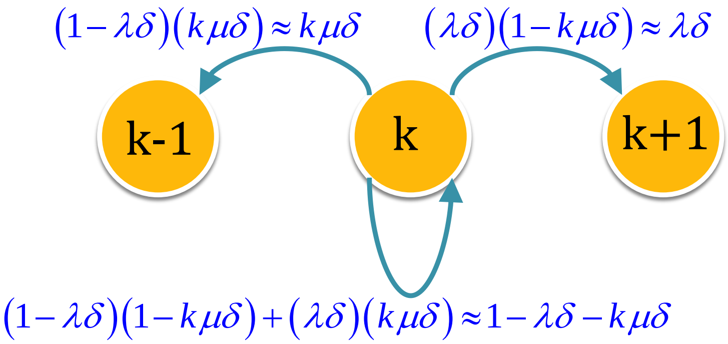

$(1-\lambda\delta)(k\mu\delta) \approx k\mu\delta$
$(\lambda\delta)(1-k\mu\delta) \approx \lambda\delta$
k-1
k
k+1
$(1-\lambda\delta)(1-k\mu\delta)+(\lambda\delta)(k\mu\delta) \approx 1-\lambda\delta-k\mu\delta$

Figure 4: State transition diagram for state k.

This can be summarized via a diagram in Figure 4:

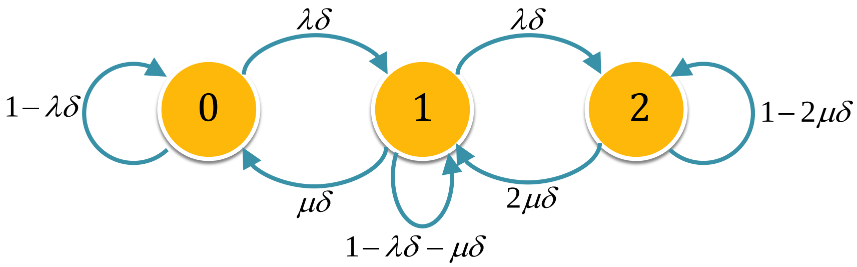
Note that the labels on the arrows indicate transition probabilities which are conditional probabilities of going from some value of K to another value.

2.9. When there are m trunked channels, the possible values of K are $0, 1, 2, \ldots, m$. We can combine the diagram above into one diagram that includes all possible values of K:

This diagram is called the ***Markov chain state diagram*** for Erlang B.

Note that the arrow $\lambda\delta$ which should go out of state m will return to state m itself because it corresponds to blocked calls which do not increase the value of K.

Example 2.10. $m = 2$:



What we need from our model is to find out the behavior of the system in the long run. The complete discussion of the Markov chain itself deserves a whole textbook. (See e.g. [2].) In the next section, we introduce the concept of Markov chain and some necessary properties.

3 Markov Chains

3.1. Markov chains are simple mathematical models for random phenomena evolving in time. Their simple structure makes it possible to say a great deal about their behavior. At the same time, the class of Markov chains is rich enough to serve in many applications. This makes Markov chains the most important examples of random processes. Indeed, the whole of the mathematical study of random processes can be regarded as a generalization in one way or another of the theory of Markov chains. [2]

3.2. The characteristic property of Markov chain is that it retains no memory of where it has been in the past. This means that only the current state of the process can influence where it goes next.

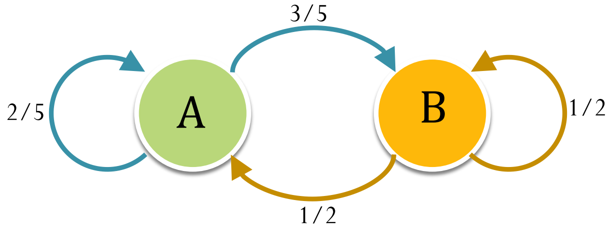
3.3. Markov chains are often best described by their (state) diagrams. You have seen a Markov chain in Example 2.10.

Example 3.4. Draw the (state) diagram of a Markov chain which has three states: 1,2 and 3. You move (transition) from state 1 to state 2 with probability 1. From state 3 you move either to 1 or to 2 with equal probability 1/2, and from 2 you jump to 3 with probability 1/3, otherwise stay at 2.

3.5. The Markov chains that we have seen in the Example 3.4 and in the previous section (Example 2.10) are all ***discrete-time*** Markov chains. The Poisson process that we have seen earlier is an example of a continuous-time Markov chain. However, equipped with our small-slot approximation (discrete-time approximation) technique, we may analyze the Poisson process as a discrete-time Markov chain as well.

3.6. We will now introduce the concept of stationary distribution, steady-state distribution, equilibrium distribution, and limiting distribution. For the purpose of this class, we will not distinguish these terms. We shall see in the next example that for the Markov chains that we are considering, in the long run, it will reach a steady-state distribution.

Example 3.7. Consider the Markov chain characterized by the state transition diagram given below:



Let's try a thought experiment – imagine that you start with 100,000 trials of these Markov chain, all of which start in state B. So, during slot 1 (the first time slot), all trials will be in state B. For slot 2, about 50% of these will move to state A; but the other 50% of the trials will stay at B.

By the time that you reach slot 6, you can observe that out of the 100,000 trials, about 45.5% will be in state A and about 55.5% will be in state B. These numbers stay roughly the same as you proceed to slot 7, 8, 9, and so on. Note also that it does not matter how you start your 100,000 trials. You may start with 10,000 in state A and 90,000 in state B. Eventually, the same numbers, 45.5% and 54.5%, will emerge.

In conclusion,

(a) If you look at the long run behavior of the Markov chain at a particular slot, then the probability that you will see it in state A is 0.455 and the probability that you will see it in state B is 0.545.

(b) In addition, one can also show that if you look at behavior of a Markov chain for a long time, then the proportion of time that it stays in state A is 45.5% and the proportion of time that it stays in state B is 54.5%.

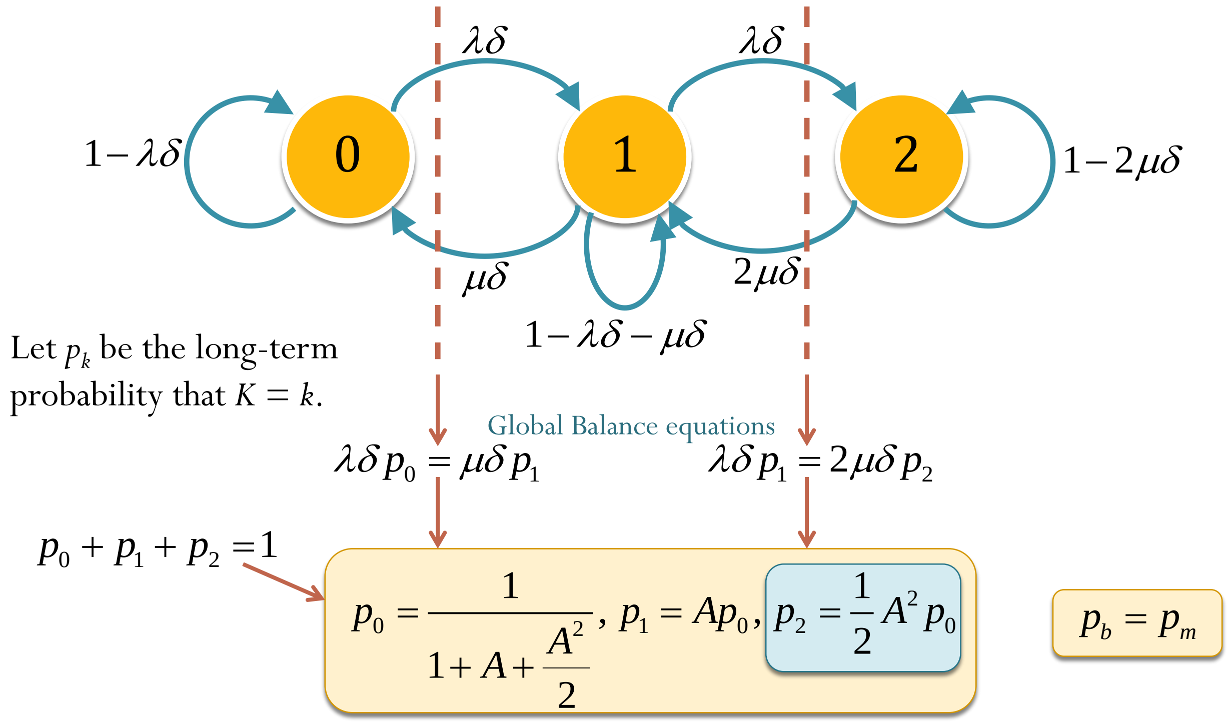
The distribution $(0.455, 0.545)$ is what we called stationary distribution, steady-state distribution, equilibrium distribution, or limiting distribution above.

3.8. In [3], the steady-state distribution for the M/M/m/m assumption is derived via the use of the ***global balance equation*** instead of finding the limit of the distribution as we have done in Example 3.7. The basic idea that leads to the global balance equation is simple. When we look back at the numbers that we got in Example 3.7, if we assume that the system will reach some steady-state values, then at the steady-state, we must have roughly the same number of transitions from state A to B and transitions from state B to state A.

Therefore,

Finally, we can now use what we learned to derive the Erlang B formula.

Example 3.9. Let's reconsider Example 2.10 where $m = 2$.



Let p_k be the long-term probability that $K = k$.

Global Balance equations

$\lambda\delta p_0 = \mu\delta p_1$ $\lambda\delta p_1 = 2\mu\delta p_2$

$p_0 + p_1 + p_2 = 1$

$p_0 = \dfrac{1}{1 + A + \dfrac{A^2}{2}}$, $p_1 = Ap_0$, $p_2 = \dfrac{1}{2}A^2 p_0$

$p_b = p_m$

The same process can be used to derive the the Erlang B formula.

3.10. In general, if we have m channels, then

$$p_m = \frac{\frac{A^m}{m!}}{\sum\limits_{k=0}^{m} \frac{A^k}{k!}}.$$

Note that p_m is the (long-run) probability that the system is in state m. When the system is in state m, all channels are used and therefore any new call request will be blocked and lost.

Here, p_m is the same as call blocking probability, which is the long-run proportion of call requests that get blocked.

4 Engsett Model

In this section, we will consider a more realistic system where the infinite-user assumption in M/M/m/m is relaxed.

4.1. Modified (more realistic) assumptions:

(a) Finite number of users: n users/customers

(b) Each user generates new call request with rate λ_u

- Total call request rate $= n \times \lambda_u = \lambda$

4.2. We need to modify our small slot analysis. Here, for each small slot, each user do the following:

(a) If it is idle (not using the channel) at the beginning of the slot,

(i) it may request/generate a new call in a small slot with probability $\lambda_u \delta$.

i. If there is at least one available channel, then it may start its conversation. (In which case, at the beginning of the next slot, its call is ongoing.)

ii. If there is no channel available, then the call is blocked and it is idle again (at the beginning of the next slot).

(ii) With probability $1 - \lambda_u \delta$, no new call is requested by this user during this slot. (In which case, it is idle at the beginning of the next slot.

(b) If it is making a call at the beginning of the slot,

(i) the call may end with probability $\mu\delta$. (In which case, at the beginning of the next slot, it is idle.)

(ii) With probability $1 - \mu\delta$, the call is still ongoing at the end of this slot (which is the same as the beginning of the next slot.)

4.3. Observe that the call generation process for each user is not a Poisson process with rate λ_u. This is because it get interleaved with the call duration for each successful call request. Part of the Poisson assumption that is left is that, in fact, for an idle user, the time until the next call request will be exponential with rate λ_u.

4.4. A user can not generate new call when he/she is already involved in a call. Therefore, if the system is in state $K = k$, there are only $n - k$ users that can generate new calls. Hence, the "total" call request rate for state $K = k$ is $(n - k)\lambda_u$.

- Earlier, when we consider the Erlang B formula, we always have λ as the total new call request rate regardless of how many users are using the channels. This is because we assumed infinite number of users and hence having k users using the channels will not significantly change the total call request rate.

4.5. Comparison of the state transition probabilities:

4.6. New state transition diagram:

4.7. Comparison of the steady-state probabilities:

4.8. It is tempting to conclude that the call blocking probability is p_m. However, this is not the case for us here. Recall that p_m is the long-run probability (and the long-run proportion of time) that the system will be in state $K = m$. In this state, any new call request will be blocked. So, p_m gives the blocking probability in terms of the ***time*** (time congestion).

However, if you look back at how we define P_b which is the ***call*** blocking probability, this is the probability that a ***call*** is blocked. So, what we need to find out is, out of all the new calls that are requested, how many will be blocked.

To do this, consider s slots. Here the value of s is very large. Then,...

(a) About $p_k \times s$ slots will be in state k.

(b) Each of these slots will generate new call request with probability

$$(n-k)\lambda_u \delta.$$

(c) So, the number of new calls request from slots that are in state k will be approximately

$$(n-k)\lambda_u \delta \times (p_k \times s).$$

(d) Therefore, total number of new call requests will be approximately

$$\sum_{k=0}^{m} (n-k)\lambda_u \delta \times (p_k \times s).$$

(e) However, the number of the new call requests that get blocked is

$$(n-m)\lambda_u \delta \times (p_m \times s).$$

(f) Hence, the proportion (probability) of calls that are blocked is

$$\frac{(n-m)\lambda_u \delta \times (p_m \times s)}{\sum_{k=0}^{m}(n-k)\lambda_u \delta \times (p_k \times s)} = \frac{(n-m)p_m}{\sum_{k=0}^{m}(n-k)p_k}.$$

Plugging in the values of p_k and p_m which we got earlier, we then get

$$P_b = \frac{(n-m)p_m}{\sum\limits_{k=0}^{m}(n-k)p_k} = \frac{(n-m)\frac{A_u^m \binom{n}{m}}{z(m,n)}}{\sum\limits_{k=0}^{m}(n-k)\frac{A_u^k \binom{n}{k}}{z(m,n)}} = \frac{(n-m)A_u^m \binom{n}{m}}{\sum\limits_{k=0}^{m}(n-k)A_u^k \binom{n}{k}}.$$

4.9. Comparison of the call blocking probability:

4.10. Remarks:

(a) If we keep the total rate λ constant and let $n \to \infty$, then the call blocking probability in the Engsett model will be the same as the call blocking probability in the Erlang model.

(b) If $n \leq m$, the call blocking probability in Engsett model will be 0.

References

[1] Andrea Goldsmith. *Wireless Communications.* Cambridge University Press, 2005. (document)

[2] J. R. Norris. *Markov Chains.* Cambridge University Press, 1998. 2, 3.1

[3] Theodore S. Rappaport. *Wireless Communications: Principles and Practice.* Prentice Hall PTR, 2 edition, 2002. 3.8