

Exercise 6: Erlang Formulas and M/G/1 Systems

Roman Dunaytsev

Department of Communications Engineering
Tampere University of Technology
dunaytse@cs.tut.fi

December 04, 2009

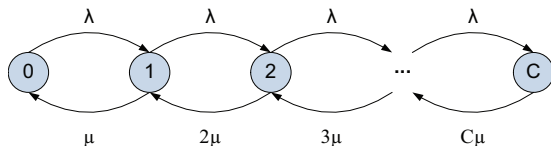
Outline

- 1 $M/M/C/C$
 - Erlang B formula
- 2 $M/M/C$
 - Erlang C formula
- 3 $M/G/1$
 - Performance characteristics

Outline

- 1 $M/M/C/C$
 - Erlang B formula
- 2 $M/M/C$
 - Erlang C formula
- 3 $M/G/1$
 - Performance characteristics

- Some network nodes can be built with no buffering elements
 - One application of such systems is for cases in which delay-sensitive traffic, such as voice, needs a queueless service
- These types of nodes can be represented by **queueless** models (i.e., $C = K$)
- One example of such models is the $M/M/C/C$ (aka $M/M/a/a$ or $M/M/m/m$) queueing system, which has a number of servers but no buffers
 - The capacity of the system is equal to the number of servers (i.e., no waiting positions)
 - If all servers are busy, further arrivals are turned away



Erlang B Formula

- In $M/M/C/C$ systems, we can compute the blocking probability, using **the Erlang B formula** :

$$E_B(C, \rho) = \frac{\frac{\rho^C}{C!}}{\sum_{k=0}^C \frac{\rho^k}{k!}}$$

- It is called after the Danish telephone engineer Agner Krarup Erlang
- The Erlang B formula is tabulated in **the Erlang B tables** in terms of the following 3 parameters:
 - Blocking probability, $E_B(C, \rho)$
 - Number of servers/channels/etc., C
 - Offered load, ρ , measured in **Erlangs**
- Given 2 of the 3 parameters, the value of the third parameter can be found from the Erlang B tables

Erlang B Formula (cont'd)

- An Erlang (Erl or E, for short) is a **dimensionless unit** of telecommunications traffic (aka load) measurement
- Originally, an Erlang represents the continuous use of 1 voice path
- In practice, it is used to describe the total traffic volume of 1 hour
- E.g., the maximum traffic intensity of a line system with 30 channels is 30 Erl
 - I.e., all the 30 channels are in use 60 minutes during the busy hour

Erlang B Formula (cont'd)

- Probability of **blocking** (Erlang B formula):

$$E_B(C, \rho) = P\{N(t) = C\} = p_c = \frac{\frac{\rho^C}{C!}}{\sum_{k=0}^C \frac{\rho^k}{k!}}$$

- **Offered traffic** (aka **traffic intensity**):

$$\rho = \lambda E[X] = \frac{\lambda}{\mu}$$

- **Carried traffic** (aka **serviced traffic**):

$$\rho(1 - E_B(C, \rho))$$

- Probability that there are n jobs in the system:

$$P\{N(t) = n\} = p_n = \frac{\frac{\rho^n}{n!}}{\sum_{k=0}^C \frac{\rho^k}{k!}}, \quad n = 0, 1, \dots, C$$

Erlang B Formula (cont'd)

- **Example:** Telco
- A company has decided to install a tie-line phone system between its East Coast and West Coast facilities. A caller receives a busy signal if the call is dialed when all the lines are in use. On average, 105 calls per hour with an average length of 4 minutes are expected. Enough lines are to be provided to ensure that the probability of getting a busy signal will not exceed 0.005
- How many lines should be provided? With this number of lines, what will be the carried traffic during the peak period? How many lines are required if the probability of a busy signal should not exceed 0.01? What would the performance be with 10 lines?
- **Erlang B Calculator**
 - www.erlang.com/calculator/erlb/
 - <http://personal.telefonica.terra.es/web/vr/erlang/eng/cerlangb.htm>

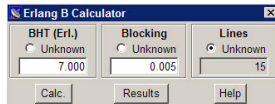
Erlang B Formula (cont'd)

- **Example:** Telco (cont'd)
- How many lines should be provided to ensure that the probability of getting a busy signal will not exceed 0.005?
- The traffic intensity is

$$\rho = 105 * \frac{4}{60} = 7 \text{ Erl}$$

- We apply the Erlang B formula to solve the problem:

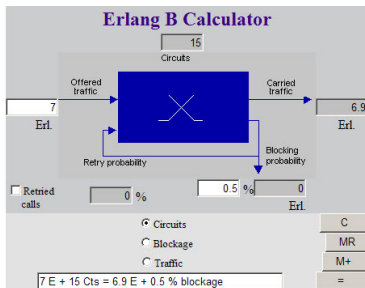
$$B(C, 7) = 0.005 \Rightarrow C = 15 \text{ lines}$$



Erlang B Formula (cont'd)

- **Example:** Telco (cont'd)
- With this number of lines, what will be the carried traffic during the peak period?
- With 15 lines, the carried traffic is:

$$\rho(1 - B(C, \rho)) = 7(1 - 0.005) \approx 6.9 \text{ Erl}$$



Erlang B Formula (cont'd)

- **Example:** Telco (cont'd)
- How many lines are required if the probability of a busy signal should not exceed 0.01?
- The smallest C such that $B(C, 7) \leq 0.01$ is 14
- Thus, we save only 1 line if we double the allowed probability of a busy signal

The screenshot shows a window titled "Erlang B Calculator" with three input fields and three buttons. The "BHT (Erl.)" field has a radio button for "Unknown" and a text box containing "7.000". The "Blocking" field has a radio button for "Unknown" and a text box containing "0.010". The "Lines" field has a radio button for "Unknown" and a text box containing "14". Below the fields are three buttons: "Calc.", "Results", and "Help".

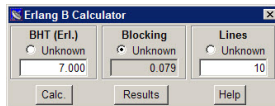
Erlang B Formula (cont'd)

- **Example:** Telco (cont'd)
- What would the performance be with 10 lines?
- If only 10 lines are provided, the probability of a busy signal is

$$B(10, 7) \approx 0.079$$

- And the blocked traffic is

$$\rho B(C, \rho) = 7 * 0.079 \approx 0.6 \text{ Erl}$$

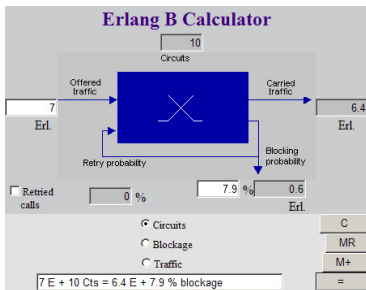


The screenshot shows a window titled "Erlang B Calculator" with three input fields and three buttons. The "BHT (Erl.)" field has a radio button for "Unknown" and a text box containing "7.000". The "Blocking" field has a radio button for "Unknown" and a text box containing "0.079". The "Lines" field has a radio button for "Unknown" and a text box containing "10". Below the fields are three buttons: "Calc.", "Results", and "Help".

Erlang B Formula (cont'd)

- **Example:** Telco (cont'd)
- What would the performance be with 10 lines?
- If only 10 lines are provided, the probability of a busy signal is $B(10, 7) \approx 0.079$, so the carried traffic is

$$\rho(1 - B(10, 7)) = 7(1 - 0.079) \approx 6.4 \text{ Erl}$$



Outline

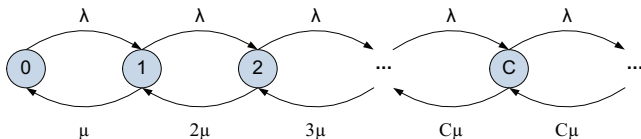
- 1 $M/M/C/C$
 - Erlang B formula
- 2 $M/M/C$
 - Erlang C formula
- 3 $M/G/1$
 - Performance characteristics

Erlang C Formula

- The probability of **delay**, the most common index of **grade of service (GoS)** for queuing systems when dealing with full availability and a Poisson arrival process, is calculated using **the Erlang C formula**, which assumes an infinitely long queue length:

$$M/M/C$$

- The state transition diagram



Erlang C Formula (cont'd)

- Just as the Erlang B formula, Erlang C assumes an infinite population of sources, which jointly offer traffic of ρ Erl to C servers
- However, if all servers are busy, further arrivals are queued
- An unlimited number of jobs may be held in the queue simultaneously
- The formula calculates the probability of queuing the offered traffic, assuming that blocked jobs stay in the system until they can be handled
- The Erlang C formula is tabulated in **the Erlang C tables**
 - www.stuffsoftware.com/trafficerlangctable.html

Erlang C Formula (cont'd)

- Probability that a call may have to wait (i.e., the probability of blocking):

$$E_C(C, \rho) = P\{W(t) > 0\} = \frac{\frac{\rho^C}{C!}}{\frac{\rho^C}{C!} + (1 - \frac{\rho}{C}) \sum_{k=0}^{C-1} \frac{\rho^k}{k!}}$$

- Or, expressed in terms of the Erlang B formula:

$$E_C(C, \rho) = \frac{C * E_B(C, \rho)}{C - \rho(1 - E_B(C, \rho))}$$

Erlang C Formula (cont'd)

- For dimensioning of queuing systems, the desired metric is the number of servers required to provide a given probability that a **target waiting time** W_0 is not exceeded:

$$P\{W(t) \leq W_0\} = 1 - E_C(C, \rho) * e^{-(C-\rho)\frac{W_0}{E[X]}}$$

- Where $E[X]$ is the average duration of a call
- Conditional probability that the delay is greater than W_0 given that the call is delayed

$$P\{W(t) > W_0\} = E_C(C, \rho) * e^{-(C-\rho)\frac{W_0}{E[X]}}$$

- **Erlang C Calculator**

- www.math.vu.nl/%7Ekoole/ccmath/ErlangC/index.php

Erlang C Formula (cont'd)

- **Example:** Call center
- Let us assume that there are 720 calls per hour, with an average call duration of 4 minutes
- How many agents are needed if the target answer time should be **no more than** 15 seconds
- The traffic intensity is

$$\rho = 720 * \frac{4}{60} = 48 \text{ Erl}$$

- We apply the Erlang C formula and get that $C = 1764$ agents

The screenshot shows a software window titled "Erlang-C Calculator". It contains a "Data" section with the following fields and values:

Field	Value	Unit / Note
Arrivals	720	per hour
Service time	4	minutes
Number of agents	1764	(integer required)
Average waiting time	0.00	seconds
Service level	100	% waits less than 15.00 seconds

At the bottom of the window is a button labeled "compute the missing values".

Erlang C Formula (cont'd)

- **Example:** Call center (cont'd)
- How many agents are needed if the target **average** answer time should be 15 seconds
- To provide this service level, we need $C = 54$ agents
- In this case, the waiting time for 81.72% of calls will be less than 20 s

The screenshot shows a software window titled "Erlang-C Calculator" with a "Data" section. It contains several input fields and checkboxes:

- Arrivals: 720 per hour
- Service time: 4 minutes
- Number of agents: 54 (integer required)
- Average waiting time: 15.00 seconds
- Service level: 81.72 % waits less than 20.00 seconds

A button at the bottom is labeled "compute the missing values".

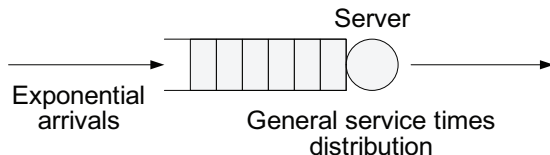
Outline

- 1 $M/M/C/C$
 - Erlang B formula
- 2 $M/M/C$
 - Erlang C formula
- 3 $M/G/1$
 - Performance characteristics

- Queueing systems with **general service time distribution**:

$$M/G/1$$

- Jobs arrive according to a Poisson process with parameter $\lambda = 1/E[\tau]$
- General (i.e., arbitrary) service time with the mean $E[X] = 1/\mu$ and **variance** $VAR[X]$, where $VAR[X] = E[X^2] - E[X]^2$
- There is a single server
- The buffer is of infinite size
- The number of potential jobs is infinite
- The jobs are processed in order of arrival (FCFS)



M/G/1 (cont'd)

- The $M/M/1$ model is extremely useful in obtaining a quick insight into the trade-offs between the basic queueing systems parameters (mean waiting times, average numbers of customers, etc.)
 - Poisson arrivals are in many cases a fairly realistic model for the arrival process
 - But exponential service times are not very common in practice
- The $M/G/1$ model allows to consider a more general class of queueing systems
 - I.e., the service time can have any distribution and is not restricted to exponential

Performance characteristics:

- **Utilization** of the server:

$$\rho = \lambda E[X] = \frac{\lambda}{\mu}$$

- Mean **waiting** time (aka **the Pollaczek-Khinchin formula**):

$$W_q = \frac{\lambda E[X^2]}{2(1 - \rho)} = \frac{\lambda(\text{VAR}[X] + E[X]^2)}{2(1 - \rho)} = \frac{\rho(1 + C_X^2)}{2(1 - \rho)} E[X]$$

- Since $\text{VAR}[X] = E[X^2] - E[X]^2$, then $E[X^2] = \text{VAR}[X] + E[X]^2$
- **Squared coefficient of variation of the service time** is given by

$$C_X^2 = \frac{\text{VAR}[X]}{E[X]^2}$$

Performance characteristics (cont'd):

- Mean **service** time:

$$W_s = E[X]$$

- Mean **sojourn** time:

$$W = W_q + W_s$$

- Expected number of jobs in the **queue**:

$$L_q = \rho^2 \frac{(1 + C_X^2)}{2(1 - \rho)}$$

- Expected number of jobs in the **server**:

$$L_s = \frac{\lambda}{\mu} = \frac{E[X]}{E[\tau]} = \rho$$

- Expected number of jobs in the **system**:

$$L = L_q + L_s$$

- The squared coefficient of variation defined for a random variable X is a useful parameter to measure the character of probability distributions used to represent **service** or **interarrival** times

$$C_X^2 = \frac{\text{VAR}[X]}{E[X]^2}$$

- Considering the service time distribution of a system with Poisson arrivals, we get:
- If X is a **constant** random variable, then $\text{VAR}[X] = 0 \Rightarrow C_X^2 = 0 \Rightarrow M/D/1$
- If X has an **exponential** distribution, then $C_X^2 = 1 \Rightarrow M/M/1$
- If X has an **Erlang** distribution, then $C_X^2 < 1 \Rightarrow M/Er/1$
- If X has an **hyperexponential** distribution, then $C_X^2 > 1 \Rightarrow M/H/1$

- Poisson process (exponential distribution):

$$E[X] = \frac{1}{\mu}, \quad \text{VAR}[X] = \frac{1}{\mu^2}, \quad C_X^2 = \frac{\text{VAR}[X]}{E[X]^2} = 1$$

- Note that the mean waiting time and the expected number of jobs in the queue increase with the squared coefficient of variation of the service time:

$$W_q = \frac{\rho(1 + C_X^2)}{2(1 - \rho)} E[X], \quad L_q = \rho^2 \frac{(1 + C_X^2)}{2(1 - \rho)}$$

- **Example:** ATM multiplexer
- The output buffer of an ATM multiplexer can be modeled as $M/D/1$ queue. ATM cells have a fixed size (53 bytes) and its transmission time is constant. The link speed is 155 Mbit/s and the average arrival rate on the link is 124 Mbit/s
- Find the average service time (i.e., the average transmission time), the link utilization, the average number of cells in the buffer (including the cell being transmitted), the average time that a cell spends in the buffer
- The average transmission time is

$$E[X] = \frac{53 * 8}{155 * 10^6} \approx 2.7 * 10^{-6} \text{ s}$$

- The link utilization is

$$\rho = \frac{\lambda}{\mu} = \frac{124 * 10^6}{155 * 10^6} = 0.8$$

- **Example:** ATM multiplexer (cont'd)
- The average number of cells in the buffer (including the cell being transmitted)?

$$M/D/1 \Rightarrow \text{VAR}[X] = 0 \Rightarrow C_X^2 = \frac{\text{VAR}[X]}{E[X]^2} = 0$$

- Therefore

$$L_q = \rho^2 \frac{(1 + C_X^2)}{2(1 - \rho)} = \frac{0.8^2(1 + 0)}{2(1 - 0.8)} = 1.6 \text{ cells}$$

- Then the expected number of cells in the system is

$$L_s = \rho \Rightarrow L = L_q + L_s = 1.6 + 0.8 = 2.4 \text{ cells}$$

- **Example:** ATM multiplexer (cont'd)
- The average time that a cell spends in the buffer?

$$M/D/1 \Rightarrow \text{VAR}[X] = 0 \Rightarrow C_X^2 = \frac{\text{VAR}[X]}{E[X]^2} = 0$$

- Using the Pollaczek-Khinchin formula, we get

$$W_q = \frac{\rho(1 + C_X^2)}{2(1 - \rho)} E[X] = \frac{0.8(1 + 0)}{2(1 - 0.8)} * 2.7 * 10^{-6} \approx 5.4 * 10^{-6} \text{ s}$$