Traffic Engineering

Traffic Engineering

- One billion+ terminals in voice network alone
  - Plus data, video, fax, finance, etc.
- Imagine all users want service simultaneously...its not even *nearly* possible (despite our common intuition)
  - In practice, the actual amount of equipment provisioned is <u>vastly less than would support all users</u> simultaneously
- And yet, by and large, we get the impression of phone and data networks that work very well!
- How is this possible?

→ Traffic theory !!

Traffic Engineering – Trade-offs

- Design number of transmission paths, or radio channels?
  - How many required normally?
  - What if there is an overload?
- Design switching and routing mechanisms
  - How do we route efficiently?
  - E.g.
    - High-usage trunk groups
    - Overflow trunk groups
    - Where should traffic flows be combined or kept separate?
- Design network topology
  - Number and sizing of switching nodes and locations
  - Number and sizing of transmission systems and locations
  - Survivability

Characterization of Telephone Traffic

- **Calling Rate** (γ) – also called arrival rate, or attempts rate, etc.
  - Average number of calls initiated per unit time (e.g. attempts per hour)
  - Each call arrival is independent of other calls (we assume)
  - Call attempt arrivals are random in time
  - Until otherwise, we assume a “large” calling group or source pool

If receive α calls from a terminal in time **T**: $\gamma = \frac{\alpha}{T}$

If receive α calls from **m** terminals in time T:

$$\gamma_g = \frac{\alpha}{T}$$

Group calling rate

$$\gamma = \frac{\alpha}{m \cdot T}$$

Per terminal calling rate

Characterization of Telephone Traffic (2)

- Calling rate assumption:
  - Number of calls in time T is **Poisson** distributed:

$$p(x) = \frac{e^{-\lambda} \cdot \lambda^{x}}{x!} \qquad x = 0, 1, 2\ldots$$

  - In our case $\lambda = \gamma T$

- $\therefore$ Time between calls is “-ve exponentially” distributed:

$$f(t) = \lambda \cdot e^{-\lambda t} \qquad 0 \le t \le \infty \qquad mean = \frac{1}{\lambda}$$

• **Class Question:** What do these observations about telephone traffic imply about the nature of the traffic sources?

-ve Exponential Holding Times

- **Implies the "Memory-less" property**
  - Prob. a call last another minute is *independent* of how long the call has already lasted! Call "forgets" that it has already survived to time T_1

$$P\left(T \geq T_1 + t \,\middle|\, T \geq T_1\right) = P\left(T \geq t\right)$$

- Proof:

$$P\left(T \geq T_1 + t \,\middle|\, T \geq T_1\right) = \frac{P\left(T \geq T_1 + t \cap T \geq T_1\right)}{P\left(T \geq T_1\right)}$$

$$= \frac{P\left(T \geq T_1 + t\right)}{P\left(T \geq T_1\right)} = \frac{e^{-(T_1 + t)/h}}{e^{-T_1/h}}$$

Recall:

$P(T \geq t) = e^{-t/h}$

$$= \frac{e^{-T_1/h} \cdot e^{-t/h}}{e^{-T_1/h}} = e^{-t/h} = P\left(T \geq t\right)$$

Characterization of Telephone Traffic (3)

- **Holding Time (h)**
  - Mean length of time a call lasts
  - Probability of lasting time t or more is also –ve exponential in nature:

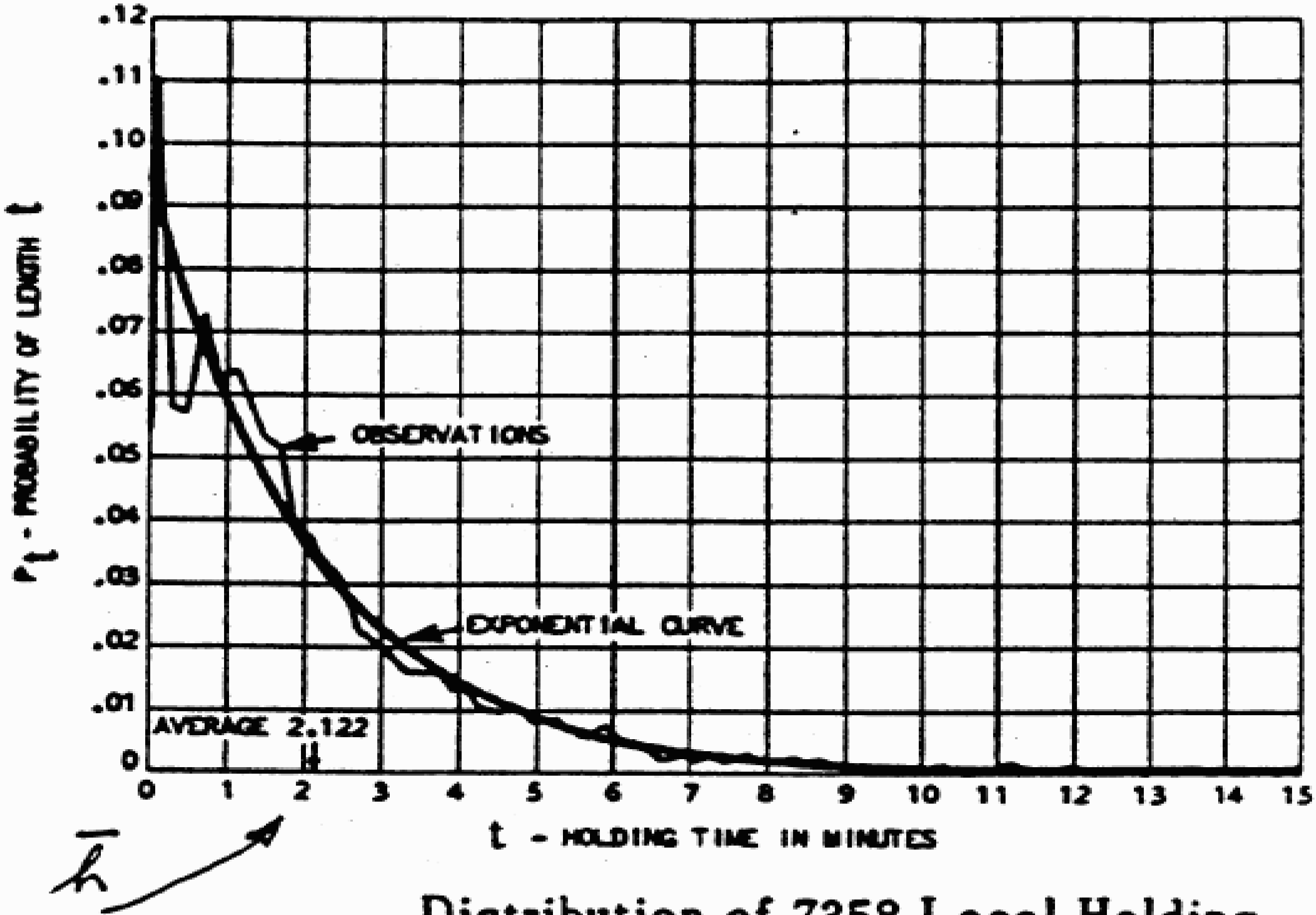
$$P(T \geq t) = e^{-t/h} \qquad t \geq 0$$

$$P(T \geq t) = 0 \qquad t < 0$$

  - Real voice calls fits very closely to the negative exponential form above
  - As non-voice “calls” begin to dominate, more and more calls have a constant holding time characteristic
- **Departure Rate (μ)**:

$$\mu = \frac{1}{h}$$

Some Real Holding Time Data



Distribution of 7358 Local Holding

Traffic *Volume (V)*

$$V = \alpha \cdot h$$

α = # calls in time period T

h = mean holding time

V = volume of calls in time period T

- In N. America this is historically usually expressed in terms of "**ccs**":
  - Hundred call seconds

    "c" "c" "s"

  - 1 ccs is volume of traffic equal to:
    - one circuit busy for 100 seconds, or
    - two circuits busy for 50 seconds, or
    - 100 circuits busy for one second, etc.

Traffic *Intensity (A)*

- Also called "**traffic flow**" or simply "**traffic**".

$$A = \frac{\alpha \cdot h}{\mathrm{T}} = \gamma \cdot h = \frac{\gamma}{\mu} = \frac{V}{\mathrm{T}}$$

Recall: $\gamma = \frac{\alpha}{\mathrm{T}}$

Recall: $\mu = \frac{1}{h}$

Recall: $V = \alpha \cdot h$

α = # calls in time period T

h = mean holding time

T = time period of observations

γ = calling rate

μ = departure rate

V = call volume

- Units:
  - "**ccs / hour**", or
  - dimensionless (if *h* and *T* are in the same units of time)

↳ "**Erlang**" unit

The *Erlang*

- Dimensionless unit of traffic *intensity*
- Named after Danish mathematician **A. K. Erlang** (1878-1929)
- Usually denoted by symbol **E**.
- 1 Erlang is equivalent to traffic intensity that keeps:
  - one circuit busy 100% of the time, or
  - two circuits busy 50% of the time, or
  - four circuits busy 25% of the time, etc.
- 26 Erlangs is equivalent to traffic intensity that keeps :
  - 26 circuits busy 100% of the time, or
  - 52 circuits busy 50% of the time, or
  - 104 circuits busy 25% of the time, etc.

Class

- Could 4 E be produced as a traffic intensity by:
  - 16 sources? (What is the utilization?)
  - 4 sources (same)
  - 1 source?

- What is special about the traffic intensity if it pertains to one source or terminal only?

Erlang (2)

- How does the **Erlang** unit correspond to **ccs**?

$$1\ \text{ccs/hour} = \frac{100\ \text{call}\cdot\text{seconds}}{1\ \text{hour}\times 60\ \text{min/hr}\times 60\ \text{sec/min}} = 0.027E$$

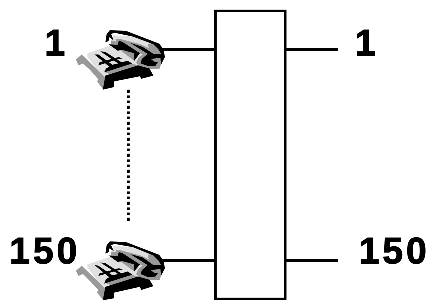
$$36\ \text{ccs/hour} = \frac{3600\ \text{call}\cdot\text{seconds}}{1\ \text{hour}\times 60\ \text{min/hr}\times 60\ \text{sec/min}} = 1E$$

- Percentage of time a terminal is busy is equivalent to the traffic generated by that terminal in Erlangs, or
- Average number of circuits in a group busy at any time
- Typical usages:
  - residence phone -> 0.02 E
  - business phone -> 0.15 E
  - interoffice trunk -> 0.70 E

Traffic Offered, Carried, and Lost

- **Offered Traffic** (T_O) equivalent to Traffic Intensity (**A**)
  - Takes into account **all attempted calls**, whether blocked or not, and uses their **expected** holding times
- Also **Carried Traffic** (T_C) and **Lost Traffic** (T_L)
- Consider a group of 150 terminals, each with 10% utilization (or in other words, 0.1 E per source) and **dedicated service**:

each terminal has an outgoing trunk (i.e. terminal:trunk ratio = 1:1)



T_O = A = 150 x 0.10 E = 15.0 E

T_C = 150 x 0.10 E = 15.0 E

T_L = 0 E

Traffic Offered, Carried, and Lost (2)

- $A = T_O = T_C + T_L$

  Traffic Intensity (A), Offered Traffic (T_O), Carried Traffic (T_C), Lost Traffic (T_L)

- $T_L = T_O$ x Prob. Blocking (or congestion)

  $= P(B) \times T_O = P(B) \times A$

- **Circuit Utilization** (ρ) - also called **Circuit Efficiency**
  - proportion of time a circuit is busy, or
  - average proportion of time each circuit in a group is busy

$$\rho = \frac{T_C}{\text{\# of Trunks}}$$

Grade of Service (gos)

- In general, the term used for some traffic design objective
- Indicative of customer satisfaction
- In systems where blocked calls are cleared, usually use:

$$gos = \frac{\mathrm{T_L}}{\mathrm{T_O}} = \frac{\mathrm{T_L}}{\mathrm{T_L} + \mathrm{T_C}} = P(B)$$

- Typical gos objectives:
  - in busy hour, range from 0.2% to 5% for local calls, however
  - generally no more that 1%
  - long distance calls often slightly higher

- In systems with queuing, gos often defined as the probability of delay exceeding a specific length of time

Grade of Service Related Terms

- **Busy Hour**
  - One hour period during which traffic volume or call attempts is the highest overall during any given time period
- **Peak (or Daily) Busy Hour**
  - Busy hour for each day, usually varies from day to day
- **Busy Season**
  - 3 months (not consecutive) with highest average daily busy hour
- **High Day Busy Hour (HDBH)**
  - One hour period during busy season with the highest load

Grade of Service Related Terms (2)

- **Average Busy Season Busy Hour (ABSBH)**
  - One hour period with highest **average daily busy hour during the busy season**
  - For example, assume days shown below make up the busy season:

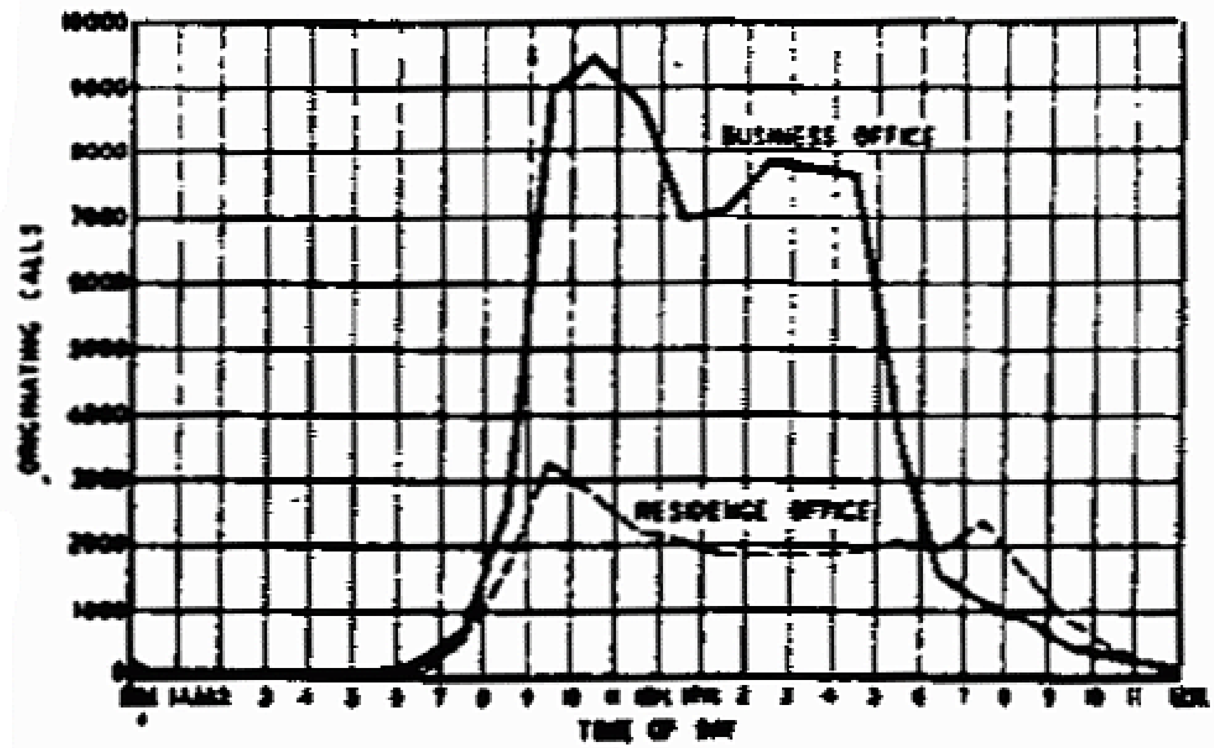
	1-Apr	2-Apr	3-Apr	4-Apr	5-Apr	6-Apr	7-Apr	8-Apr	9-Apr	10-Apr	11-Apr	12-Apr	13-Apr	14-Apr	15-Apr	16-Apr	17-Apr	18-Apr	19-Apr	20-Apr	21-Apr	Mean
00:00 to 01:00	1.4	1.4	1.2	1.5	1.1	1.5	1.7	1.5	1.0	1.0	1.8	1.5	1.8	1.6	1.2	1.9	1.8	1.6	1.4	1.5	1.2	1.5
01:00 to 02:00	1.2	1.8	1.6	1.3	1.0	1.6	1.1	1.1	1.0	1.2	1.7	2.0	2.0	1.8	1.3	1.7	1.4	1.9	1.1	1.4	1.5	1.5
02:00 to 03:00	1.4	1.8	1.5	1.9	1.2	1.0	1.2	1.1	1.1	1.7	1.5	1.5	1.9	1.9	1.3	1.5	1.8	1.1	1.1	1.2	1.5	1.4
03:00 to 04:00	1.2	1.8	1.7	1.4	1.7	1.1	1.5	1.6	1.1	1.9	1.0	1.0	1.4	1.5	1.6	1.1	1.4	1.9	1.4	1.2	1.1	1.4
04:00 to 05:00	1.8	1.8	2.3	2.2	2.0	1.7	2.3	1.6	2.2	1.5	2.1	1.6	2.3	2.1	1.7	2.5	1.6	2.0	1.7	1.5	2.3	1.9
05:00 to 06:00	2.2	2.3	1.9	2.4	2.5	2.0	2.0	1.7	1.8	1.6	2.0	2.0	2.2	2.2	2.1	1.8	1.6	1.7	2.0	2.3	2.1	2.0
06:00 to 07:00	1.7	2.2	1.7	2.5	2.2	2.1	2.2	2.0	2.3	1.6	2.4	2.2	1.5	2.1	2.2	1.8	1.8	1.7	2.1	2.0	2.1	2.0
07:00 to 08:00	2.0	2.8	2.2	2.4	2.3	2.4	2.9	2.0	2.4	2.4	2.1	2.9	2.3	2.1	2.9	2.7	2.8	2.3	2.1	2.1	2.7	2.4
08:00 to 09:00	3.4	3.1	2.8	2.9	2.5	2.7	2.9	3.0	3.4	3.4	3.1	2.9	2.9	2.9	3.3	3.2	3.5	3.1	3.1	3.1	2.5	3.0
09:00 to 10:00	3.4	3.4	4.0	3.2	3.5	3.4	3.1	3.7	3.3	3.3	3.5	3.9	3.4	**4.0**	3.7	3.7	3.1	3.4	3.9	**3.9**	3.4	3.5
10:00 to 11:00	**5.0**	4.4	**4.8**	**4.9**	4.1	3.0	**4.0**	**4.9**	4.2	**4.9**	4.7	4.2	**3.8**	3.0	4.6	**4.9**	4.4	**5.0**	**4.7**	3.6	**3.8**	**4.3**
11:00 to 12:00	4.8	**5.0**	4.7	4.3	4.5	3.8	3.4	4.2	**5.0**	4.6	**5.0**	**4.7**	3.2	3.4	**5.0**	4.8	4.1	4.3	4.4	3.6	3.7	4.3
12:00 to 13:00	4.5	4.2	4.1	4.8	4.6	**3.8**	3.3	4.0	4.2	4.6	4.7	4.0	3.3	3.1	5.0	4.9	4.6	4.1	4.2	3.2	3.6	4.1
13:00 to 14:00	4.3	4.2	4.7	4.5	**4.8**	3.2	3.1	4.1	4.5	4.6	4.9	4.7	3.6	3.6	4.8	4.2	**4.8**	4.9	4.4	3.3	3.0	4.2
14:00 to 15:00	4.8	4.7	4.5	4.1	4.4	3.6	3.7	4.5	4.3	4.3	4.9	4.5	3.5	3.5	4.3	4.3	4.3	4.5	4.3	3.3	3.2	4.2
15:00 to 16:00	4.4	4.9	4.4	4.8	4.5	3.8	3.2	4.1	4.8	4.4	4.5	4.2	3.3	3.9	4.3	4.9	4.4	4.3	4.5	3.7	3.3	4.2
16:00 to 17:00	3.2	[illegible]	[illegible]	[illegible]	[illegible]	3.1	3.5	3.5	3.2	3.2	3.8	3.4	3.2	4.0	3.3	4.0	3.9	3.0	3.3	3.5	3.3	3.5
17:00 to 18:00	2.7	[illegible]	[illegible]	[illegible]	[illegible]	3.1	3.4	2.9	3.2	2.8	2.7	3.0	3.3	3.2	2.5	2.9	2.8	3.4	3.5	2.9	3.2	3.0
18:00 to 19:00	3.0	[illegible]	[illegible]	[illegible]	[illegible]	3.4	3.3	3.4	2.7	3.3	3.5	3.5	2.7	3.1	3.1	3.3	3.4	3.1	3.0	3.3	3.3	3.1
19:00 to 20:00	3.3	[illegible]	[illegible]	[illegible]	[illegible]	2.7	2.7	3.4	3.4	3.0	3.0	3.4	3.1	2.8	3.2	3.4	3.0	3.4	3.4	3.1	2.9	3.1
20:00 to 21:00	2.9	2.3	2.1	2.9	2.9	3.0	3.0	2.4	2.3	2.9	3.0	2.1	2.2	2.9	3.0	2.6	2.4	2.5	2.7	2.7	2.6	2.6
21:00 to 22:00	2.1	1.6	2.3	1.6	2.2	2.1	2.4	1.9	1.6	2.1	2.4	1.7	1.8	2.4	1.8	1.9	2.2	1.9	2.2	2.2	1.6	2.0
22:00 to 23:00	1.5	2.1	1.9	1.6	1.7	1.6	2.3	2.5	2.4	1.7	2.1	1.8	2.0	2.4	1.7	1.9	2.2	2.3	1.7	2.4	1.8	2.0
23:00 to 00:00	1.5	1.0	1.1	1.1	1.5	1.8	1.5	1.4	1.8	1.1	1.9	1.2	1.6	1.9	1.8	1.1	1.5	2.0	1.8	1.6	1.4	1.5

ABSBH

Highest

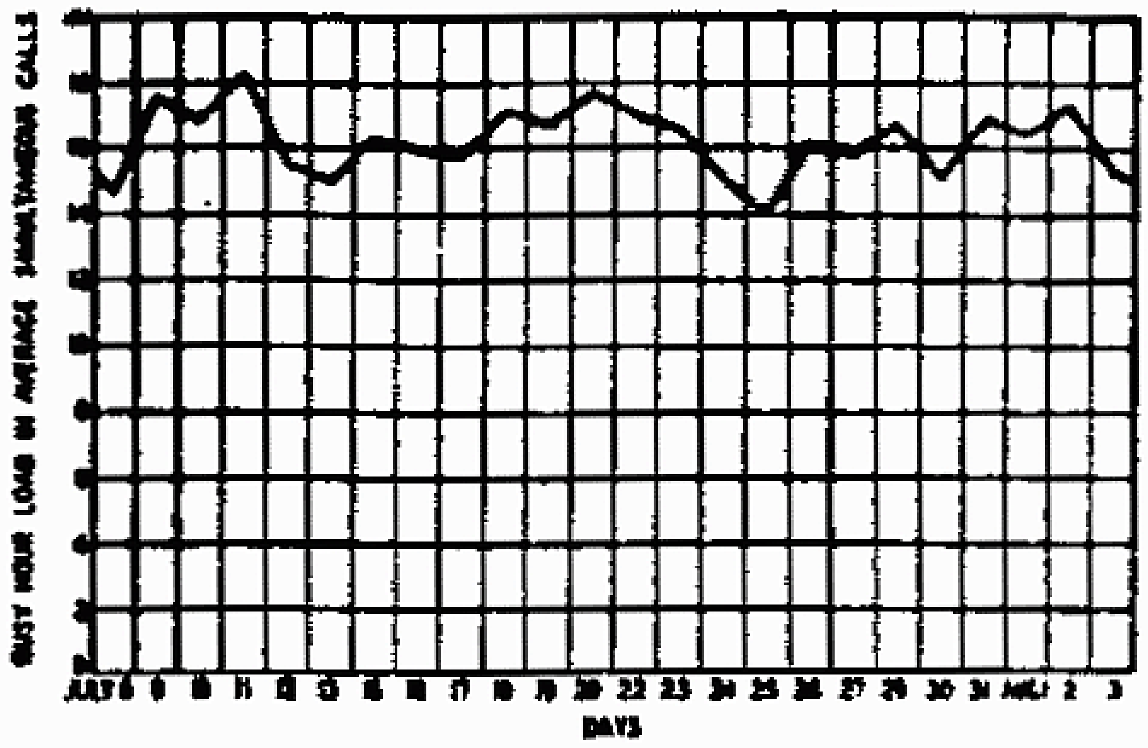
Note: Red indicates daily busy hour

Hourly Traffic Variations



Hourly Variation in Residential
and Business Calls

Daily Traffic Variations

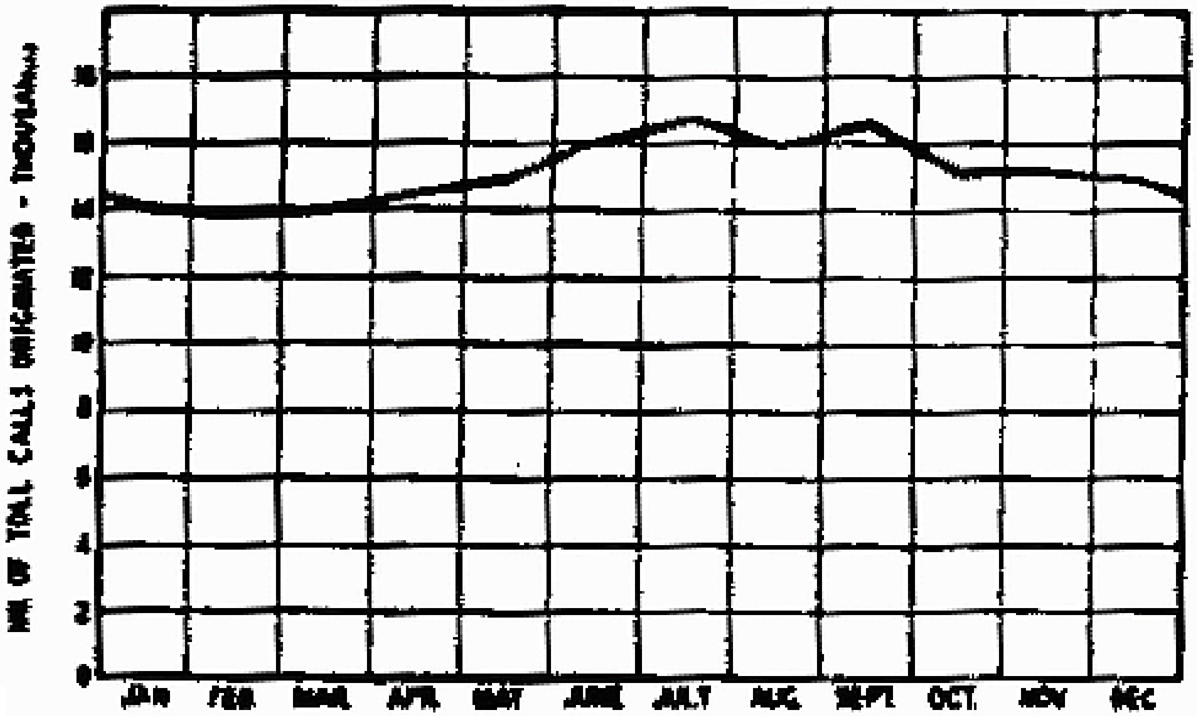


Day-to-Day Busy-Hour
Variation in Load

Seasonal Traffic Variations

Seasonal Variation in Daily Local Calls

Seasonal Traffic Variations (2)



Seasonal Variation in Daily
Toll Calls

Typical Call Attempts Breakdown

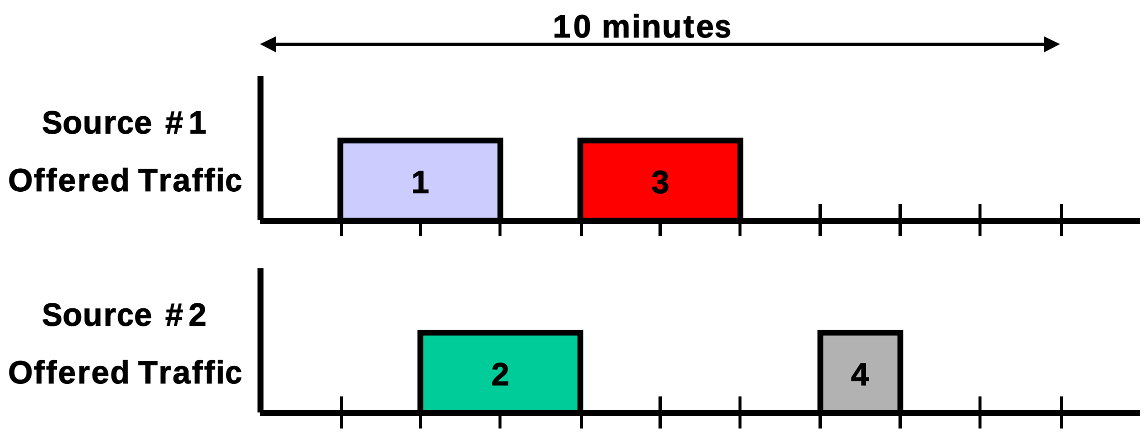
- Calls Completed - 70.7%
- Called Party No Answer - 12.7%
- Called Party Busy - 10.1%
- Call Abandoned - 2.6%
- Dialing Error - 1.6%
- Number Changed or Disconnected - 0.4%
- Blockage or Failure - 1.9%

3 Types of Blocking Models

- Blocked Calls Cleared (**BCC**)
  - Blocked calls leave system and do not return
  - Good approximation for calls in 1st choice trunk group
- Blocked Calls Held (**BCH**)
  - Blocked calls remain in the system for the amount of time it would have normally stayed for
  - If a server frees up, the call picks up in the middle and continues
  - Not a good model of real world behaviour (mathematical approximation only)
  - Tries to approximate call reattempt efforts
- Blocked Calls Wait (**BCW**)
  - Blocked calls enter a queue until a server is available
  - When a server becomes available, the call's holding time begins

Blocked Calls Cleared (BCC)

2 sources

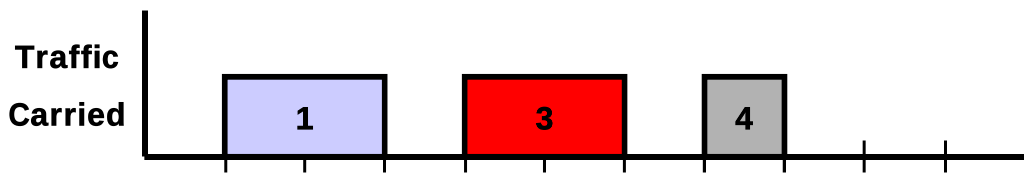


Total Traffic Offered:

T_O = 0.4 E + 0.3 E

T_O = 0.7 E

Only one server



Total Traffic Carried:

T_C = 0.5 E

1st call arrives and is served

2nd call arrives but server already busy

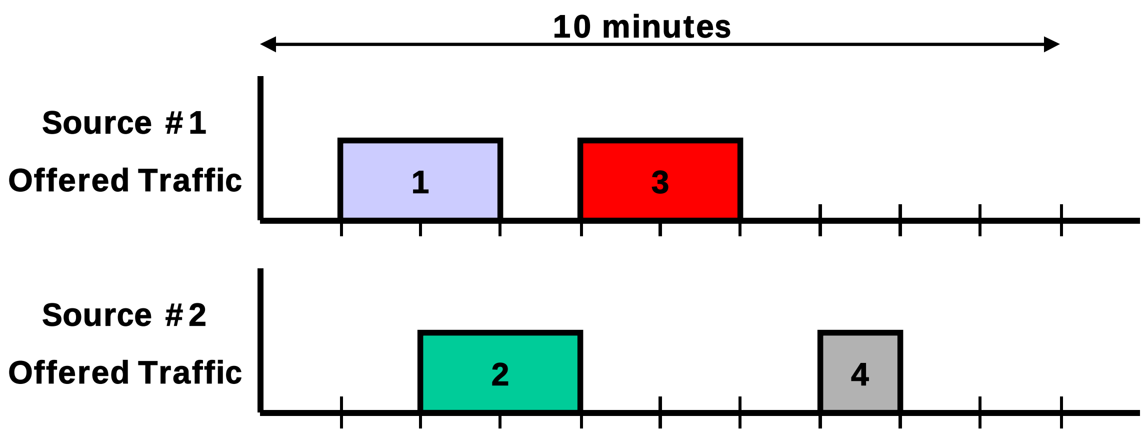
2nd call is cleared

3rd call arrives and is served

4th call arrives and is served

Blocked Calls Held (BCH)

2 sources

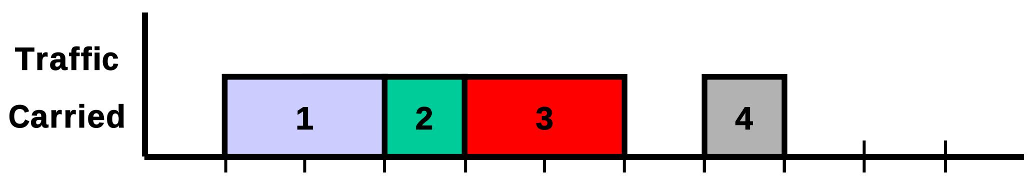


Total Traffic Offered:

T_O = 0.4 E + 0.3 E

T_O = 0.7 E

Only one server



Total Traffic Carried:

T_C = 0.6 E

1st call arrives and is served

2nd call arrives but server busy

2nd call is held until server free

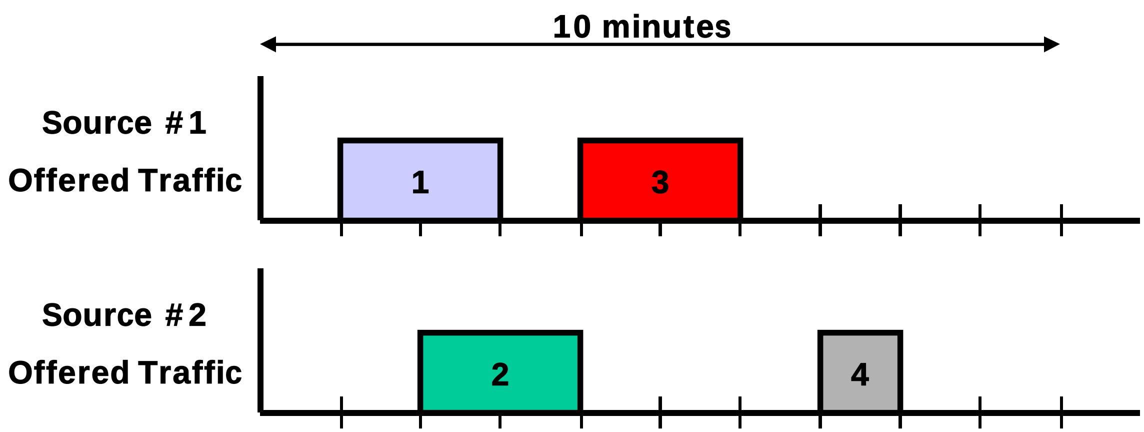
2nd call is served

3rd call arrives and is served

4th call arrives and is served

Blocked Calls Wait (BCW)

2 sources

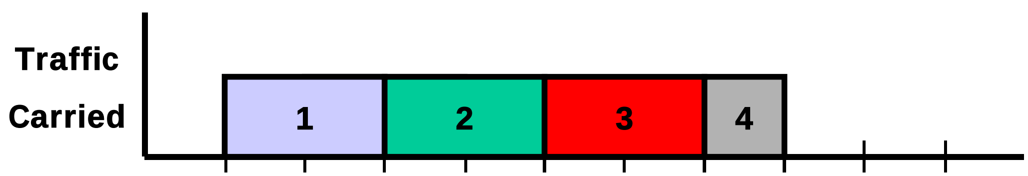


Total Traffic Offered:

T_O = 0.4 E + 0.3 E

T_O = 0.7 E

Only one server



Total Traffic Carried:

T_C = 0.7 E

1st call arrives and is served

2nd call arrives but server busy

2nd call waits until server free

2nd call served

3rd call arrives, waits, and is served

4th call arrives, waits, and is served

Blocking Probabilities

- System must be in a **Steady State**
  - Also called state of statistical equilibrium
  - **Arrival Rate** of new calls equals **Departure Rate** of disconnecting calls
  - Why?
    - If calls arrive faster that they depart?
    - If calls depart faster than they arrive?

Binomial Distribution Model

- Assumptions:
  - **m** sources
  - **A** Erlangs of offered traffic
    - per source: $\mathbf{T_O = A / m}$
    - probability that a **specific** source is busy: **P(B) = A / m**
- Can use Binomial Distribution to give the probability that a certain number (**k**) of those m sources is busy:

$$P(k) = \binom{m}{k}\left(\frac{A}{m}\right)^{k}\left(1-\frac{A}{m}\right)^{m-k} = \left(\frac{m!}{k!(m-k)!}\right)\left(\frac{A}{m}\right)^{k}\left(1-\frac{A}{m}\right)^{m-k}$$

Binomial Distribution Model (2)

- What does it mean if we only have **N servers** (N<m)?
  - We can have at most N busy sources at a time
  - What about the probability of blocking?
    - All N servers must be busy before we have blocking

$$P(B) = P(k \geq N) = P(k = N) + P(k = N+1) + \ldots + P(k = m)$$

$$= \sum_{k=N}^{m} \binom{m}{k} \left(\frac{A}{m}\right)^k \left(1 - \frac{A}{m}\right)^{m-k}$$

$$= 1 - \sum_{k=0}^{N-1} \binom{m}{k} \left(\frac{A}{m}\right)^k \left(1 - \frac{A}{m}\right)^{m-k}$$

Remember:

$$P(k) = \binom{m}{k} \left(\frac{A}{m}\right)^k \left(1 - \frac{A}{m}\right)^{m-k}$$

Binomial Distribution Model (3)

- What does it mean if k>N?
  - Impossible to have more sources busy than servers to serve them
  - Doesn't accurately represent reality
    - In reality, P(k>N) = 0
  - In this model, we still assign P(k>N) = A/m
  - Acts as good model of real behaviour
    - Some people call back, some don't
- Which type of blocking model is the Binomial Distribution?
  - Blocked Calls Held (BCH)

Time Congestions vs. Call Congestion

- Time Congestion
  - Proportion of time a system is congested (all servers busy)
  - Probability of blocking from point of view of servers
- Call Congestion
  - Probability that an arriving call is blocked
  - Probability of blocking from point of view of calls
- Why/How are they different?

Time Congestion:

$$P(B) = P(k \geq N)$$

Probability that all servers are busy.

Call Congestion:

$$P(B) = P(k > N)$$

Probability that there are more sources wanting service than there are servers.

Poisson Traffic Model

- Poisson approximates Binomial with **large m** and **small A / m**

$$P(k) = \frac{e^{-\lambda}\lambda^{k}}{k!}$$

λ = Mean # of Busy Sources

Note: $$Poisson = \lim_{m \to \infty}(Binomial)$$

- What is λ?
  - Mean number of busy sources
  - λ = A

$$\therefore P(k) = \frac{e^{-A}A^{k}}{k!}$$

Poisson Traffic Model (2)

- Now we can calculate probability of blocking:

$$P(B) = P(k \geq N) = P(N) + P(N+1) + \ldots + P(\infty)$$

$$= \sum_{k=N}^{\infty} \frac{e^{-A} A^k}{k!} = \sum_{k=N}^{\infty} \frac{A^k}{k!} e^{-A}$$

Remember:

$$P(k) = \frac{e^{-A} A^k}{k!}$$

$$= 1 - \sum_{k=0}^{N-1} \frac{A^k}{k!} e^{-A}$$

$$P(B) = P(N, A)$$

"P" = Poisson

"N" = # Servers

"A" = Offered Traffic

Example:

$$P(7{,}10)$$

Poisson P(B) with **10 E** offered to **7 servers**

Traffic Tables

- Consider a 1% chance of blocking in a system with N=10 trunks
  - How much offered traffic can the system handle?

$$0.01 = \sum_{k=10}^{\infty} \frac{A^k}{k!} e^{-A} = 1 - \sum_{k=0}^{9} \frac{A^k}{k!} e^{-A}$$

- How do we calculate A?
  - Very carefully, or
  - Use traffic tables

Traffic Tables (2)

P(B)=P(N,A)

N

A

Poisson Traffic Capacity in Erlangs

No. of Trunks (*N*)	Traffic (*A*) in Erlangs for *P* = .001	.002	.005	.010	.020	.050	.100
1	.001	.002	.005	.011	.021	.053	.106
2	.044	.065	.104	.150	.214	.358	.531
3	.192	.244	.338	.436	.567	.817	1.10
4	.428	.519	.673	.822	1.02	1.37	1.75
5	.739	.868	1.08	1.28	1.55	1.97	2.44
6	1.11	1.27	1.54	1.79	2.11	2.61	3.14
7	1.52	1.72	2.04	2.33	2.69	3.28	3.89
8	1.97	2.21	2.57	2.91	3.31	3.97	[illegible]
9	2.45	2.72	3.13	3.50	3.94	4.69	5.42
10	2.97	3.28	3.72	4.14	[illegible]	[illegible]	6.22
11	3.50	3.82	4.32	4.78	5.31	6.17	7.03
12	4.03	4.40	4.94	5.43	6.00	6.92	7.83
13	4.61	5.00	5.58	6.11	6.69	7.69	8.64
14	5.19	5.61	6.23	6.78	7.42	8.47	9.47
15	5.78	6.23	6.89	7.47	8.14	9.25	10.3
16	6.42	6.87	7.57	8.18	[illegible]	10.1	11.1
17	7.03	7.52	8.25	8.89	9.64	10.8	12.0
18	7.67	8.17	8.94	9.61	10.4	11.6	12.8
19	8.31	8.84	9.65	10.4	11.1	12.4	13.7
20	8.97	9.52	10.4	11.1	11.9	13.3	14.5
21	9.61	10.2	11.1	11.8	12.7	14.1	15.4
22	10.3	10.9	11.8	12.6	13.5	14.9	16.3
23	11.0	11.6	12.5	13.3	14.3	15.7	17.1
24	11.6	12.3	13.3	14.1	15.1	16.6	18.0
25	12.3	13.0	14.0	14.9	15.9	17.4	18.8
26	13.0	13.7	14.7	15.6	16.6	18.2	19.7
27	13.8	14.4	15.5	16.4	17.4	19.1	20.6
28	14.4	15.2	16.3	17.2	18.2	19.9	21.5
29	15.1	15.9	17.0	18.0	19.0	20.8	22.4
30	15.9	16.6	17.8	18.8	19.9	21.6	23.2
31	16.6	17.4	18.5	19.5	20.7	22.5	24.1
32	17.3	18.1	19.3	20.3	21.5	23.3	25.0
33	18.1	18.9	20.1	21.1	22.3	24.2	25.9
34	18.8	19.6	20.9	21.9	23.1	25.1	26.8
35	19.5	20.4	21.6	22.7	23.9	25.9	27.7
36	20.3	21.1	22.4	23.5	24.8	26.8	28.6
37	21.0	21.9	23.2	24.3	25.6	27.6	29.4
38	21.8	22.7	24.0	25.1	26.4	28.5	30.3
39	22.5	23.4	24.8	26.0	27.3	29.4	31.3
40	23.3	24.2	25.6	26.8	28.1	30.2	32.1

(table continues)

Poisson Traffic Capacity in Erlangs (*Continued*)

No. of Trunks (*N*)	Traffic (*A*) in Erlangs for *P* = .001	.002	.005	.010	.020	.050	.100
41	24.0	25.0	26.4	27.6	28.9	31.1	33.1
42	24.8	25.8	27.2	28.4	29.8	32.0	33.9
43	25.5	26.5	28.0	29.2	30.6	32.9	34.9
44	26.3	27.3	28.8	30.1	31.5	33.7	35.8
45	27.1	28.1	29.6	30.9	32.3	34.6	36.7
[illegible]	27.9	28.9	30.4	31.7	33.2	35.5	37.6
47	28.6	29.7	31.2	32.5	34.0	36.4	38.5
48	29.4	30.5	32.0	33.4	34.9	37.2	39.4
49	30.2	31.3	32.9	34.2	35.7	38.1	40.3
50	31.0	32.1	33.7	35.0	36.6	39.0	41.2
51	31.8	32.9	34.5	35.9	37.4	39.9	42.1
52	32.5	33.7	35.3	36.7	38.3	40.8	43.0
53	33.3	34.5	36.1	37.6	39.2	41.6	43.9
54	34.1	35.3	37.0	38.4	40.0	42.5	44.8
55	34.9	36.1	37.8	39.2	40.9	43.4	45.7
56	35.7	36.9	38.6	40.1	41.8	44.3	46.6
57	36.5	37.7	39.3	40.9	42.6	45.2	47.6
58	37.3	38.5	40.3	41.8	43.5	46.1	48.5
59	38.1	39.3	41.1	42.6	44.3	47.0	49.4
60	38.9	40.1	41.9	43.5	45.2	47.9	50.3
61	39.7	40.9	42.8	44.3	46.1	48.8	51.2
62	40.5	41.8	43.6	45.2	46.9	49.6	52.1
63	41.3	42.6	44.4	46.0	47.8	50.5	53.1
64	42.1	43.4	45.3	46.9	48.7	51.4	54.0
65	42.9	44.2	46.1	47.7	49.6	52.3	54.9
66	43.7	45.0	46.9	48.6	50.4	53.2	55.8
67	44.5	45.9	47.8	49.4	51.3	54.1	56.7
68	45.3	46.7	48.7	50.3	52.2	55.0	57.7
69	46.1	47.5	49.5	51.2	53.1	55.9	58.6
70	47.0	48.4	50.3	52.0	53.9	56.8	59.5
71	47.8	49.2	51.2	52.9	54.8	57.7	60.4
72	48.6	50.0	52.0	53.8	55.7	58.6	61.4
73	49.4	50.8	52.9	54.6	56.6	59.5	62.3
74	50.3	51.7	53.7	55.5	57.4	60.4	63.2
75	51.1	52.5	54.6	56.3	58.3	61.3	64.1
76	51.9	53.4	55.4	57.2	59.2	62.3	65.1
77	52.7	54.2	56.3	58.1	60.1	63.2	66.0
78	53.5	55.0	57.1	58.9	60.9	64.1	66.9
79	54.4	55.9	58.0	59.8	61.8	65.0	67.9
80	55.2	56.7	58.9	60.7	62.7	65.9	68.9

(table continues)

Traffic Tables (3)

P(N,A)=0.01

N=10

A=4.14 E

Poisson Traffic Capacity in Erlangs

No. of Trunks (*N*)	Traffic (*A*) in Erlangs for *P* = .001	.002	.005	.010	.020	.050	.100
1	.001	.002	.005	[illegible]	.021	.053	.106
2	.044	.065	.104	[illegible]	.214	.358	.531
3	.192	.244	.338	[illegible]	.567	.817	1.10
4	.428	.519	.673	[illegible]	1.02	1.37	1.75
5	.739	.868	1.08	[illegible]	1.55	1.97	2.44
6	1.11	1.27	1.54	[illegible]	2.11	2.61	3.14
7	1.52	1.72	2.04	[illegible]	2.69	3.28	3.89
8	1.97	2.21	2.57	[illegible]	3.31	3.97	4.67
9	2.45	2.72	3.13	[illegible]	3.94	4.69	5.42
10	[illegible]	[illegible]	[illegible]	4.14	4.61	5.42	6.22
11	3.50	3.82	4.32	[illegible]	5.31	6.17	7.03
12	4.03	4.40	4.94	5.43	6.00	6.92	7.83
13	4.61	5.00	5.58	6.11	[illegible]	7.69	8.64
14	5.19	5.61	6.23	6.78	[illegible]		
15	5.78	6.23	6.89	7.47	8.14		
16	6.42	6.87	7.57	8.18	8.89		
17	7.03	7.52	8.25	8.89	9.64		
18	7.67	8.17	8.94	9.61	10.4	11.6	12.8
19	8.31	8.84	9.65	10.4	11.1	12.4	13.7
20	8.97	9.52	10.4	11.1	11.9	13.3	14.5
21	9.61	10.2	11.1	11.8	12.7	14.1	15.4
22	10.3	10.9	11.8	12.6	13.5	14.9	16.3
23	11.0	11.6	12.5	13.3	14.3	15.7	17.1
24	11.6	12.3	13.3	14.1	15.1	16.6	18.0
25	12.3	13.0	14.0	14.9	15.9	17.4	18.8
26	13.0	13.7	14.7	15.6	16.6	18.2	19.7
27	13.8	14.4	15.5	16.4	17.4	19.1	20.6
28	14.4	15.2	16.3	17.2	18.2	19.9	21.5
29	15.1	15.9	17.0	18.0	19.0	20.8	22.4
30	15.9	16.6	17.8	18.8	19.9	21.6	23.2
31	16.6	17.4	18.5	19.5	20.7	22.5	24.1
32	17.3	18.1	19.3	20.3	21.5	23.3	25.0
33	18.1	18.9	20.1	21.1	22.3	24.2	25.9
34	18.8	19.6	20.9	21.9	23.1	25.1	26.8
35	19.5	20.4	21.6	22.7	23.9	25.9	27.7
36	20.3	21.1	22.4	23.5	24.8	26.8	28.6
37	21.0	21.9	23.2	24.3	25.6	27.6	29.4
38	21.8	22.7	24.0	25.1	26.4	28.5	30.3
39	22.5	23.4	24.8	26.0	27.3	29.4	31.3
40	23.3	24.2	25.6	26.8	28.1	30.2	32.1

(table continues)

Poisson Traffic Capacity in Erlangs (*Continued*)

No. of Trunks (*N*)	Traffic (*A*) in Erlangs for *P* = .001	.002	.005	.010	.020	.050	.100
41	24.0	25.0	26.4	27.6	28.9	31.1	33.1
42	24.8	25.8	27.2	28.4	29.8	32.0	33.9
43	25.5	26.5	28.0	29.2	30.6	32.9	34.9
44	26.3	27.3	28.8	30.1	31.5	33.7	35.8
45	27.1	28.1	29.6	30.9	32.3	34.6	36.7
46							
47							
48							
49							
50							
51							
52							
53							
58							
59							
60							
61							
62							
63							
64							
65							
66							
67							
68							
69							
70							
71							
72	48.6	50.0	52.0	53.8	55.7	58.6	61.4
73	49.4	50.8	52.9	54.6	56.6	59.5	62.3
74	50.3	51.7	53.7	55.5	57.4	60.4	63.2
75	51.1	52.5	54.6	56.3	58.3	61.3	64.1
76	51.9	53.4	55.4	57.2	59.2	62.3	65.1
77	52.7	54.2	56.3	58.1	60.1	63.2	66.0
78	53.5	55.0	57.1	58.9	60.9	64.1	66.9
79	54.4	55.9	58.0	59.8	61.8	65.0	67.9
80	55.2	56.7	58.9	60.7	62.7	65.9	68.9

(table continues)

If system with N = 10 trunks has P(B) = 0.01:

System can handle Offered traffic (A) = 4.14 E

Poisson Traffic Tables

P(N,A)=0.01

N=10

A=4.14 E

Poisson Traffic Capacity in Erlangs

No. of Trunks (*N*)	Traffic (*A*) in Erlangs for *P* = .001	.002	.005	.010	.020	.050	.100
1	.001	.002	.005	[illegible]	.021	.053	.106
2	.044	.065	.104	[illegible]	.214	.358	.531
3	.192	.244	.338	[illegible]	.567	.817	1.10
4	.428	.519	.673	[illegible]	1.02	1.37	1.75
5	.739	.868	1.08	[illegible]	1.55	1.97	2.44
6	1.11	1.27	1.54	[illegible]	2.11	2.61	3.14
7	1.52	1.72	2.04	[illegible]	2.69	3.28	3.89
8	1.97	2.21	2.57	[illegible]	3.31	3.97	4.67
9	2.45	2.72	3.13	[illegible]	3.94	4.69	5.42
10	[illegible]	[illegible]	[illegible]	4.14	4.61	5.42	6.22
11	3.50	3.82	4.32	[illegible]	5.31	6.17	7.03
12	4.03	4.40	4.94	5.43	6.00	6.92	7.83
13	4.61	5.00	5.58	6.11	[illegible]	7.69	8.64
14	5.19	5.61	6.23	6.78	[illegible]		
15	5.78	6.23	6.89	7.47	8.14		
16	6.42	6.87	7.57	8.18	8.89		
17	7.03	7.52	8.25	8.89	9.64		
18	7.67	8.17	8.94	9.61	10.4	11.6	12.8
19	8.31	8.84	9.65	10.4	11.1	12.4	13.7
20	8.97	9.52	10.4	11.1	11.9	13.3	14.5
21	9.61	10.2	11.1	11.8	12.7	14.1	15.4
22	10.3	10.9	11.8	12.6	13.5	14.9	16.3
23	11.0	11.6	12.5	13.3	14.3	15.7	17.1
24	11.6	12.3	13.3	14.1	15.1	16.6	18.0
25	12.3	13.0	14.0	14.9	15.9	17.4	18.8
26	13.0	13.7	14.7	15.6	16.6	18.2	19.7
27	13.8	14.4	15.5	16.4	17.4	19.1	20.6
28	14.4	15.2	16.3	17.2	18.2	19.9	21.5
29	15.1	15.9	17.0	18.0	19.0	20.8	22.4
30	15.9	16.6	17.8	18.8	19.9	21.6	23.2
31	16.6	17.4	18.5	19.5	20.7	22.5	24.1
32	17.3	18.1	19.3	20.3	21.5	23.3	25.0
33	18.1	18.9	20.1	21.1	22.3	24.2	25.9
34	18.8	19.6	20.9	21.9	23.1	25.1	26.8
35	19.5	20.4	21.6	22.7	23.9	25.9	27.7
36	20.3	21.1	22.4	23.5	24.8	26.8	28.6
37	21.0	21.9	23.2	24.3	25.6	27.6	29.4
38	21.8	22.7	24.0	25.1	26.4	28.5	30.3
39	22.5	23.4	24.8	26.0	27.3	29.4	31.3
40	23.3	24.2	25.6	26.8	28.1	30.2	32.1

(*table continues*)

Poisson Traffic Capacity in Erlangs (*Continued*)

No. of Trunks (*N*)	Traffic (*A*) in Erlangs for *P* = .001	.002	.005	.010	.020	.050	.100
41	24.0	25.0	26.4	27.6	28.9	31.1	33.1
42	24.8	25.8	27.2	28.4	29.8	32.0	33.9
43	25.5	26.5	28.0	29.2	30.6	32.9	34.9
44	26.3	27.3	28.8	30.1	31.5	33.7	35.8
45	27.1	28.1	29.6	30.9	32.3	34.6	36.7
46							
47							
48							
49							
50							
51							
52							
53							
58							
59							
60							
61							
62							
63							
64							
65							
66							
67							
68							
69							
70							
71							
72	48.6	50.0	52.0	53.8	55.7	58.6	61.4
73	49.4	50.8	52.9	54.6	56.6	59.5	62.3
74	50.3	51.7	53.7	55.5	57.4	60.4	63.2
75	51.1	52.5	54.6	56.3	58.3	61.3	64.1
76	51.9	53.4	55.4	57.2	59.2	62.3	65.1
77	52.7	54.2	56.3	58.1	60.1	63.2	66.0
78	53.5	55.0	57.1	58.9	60.9	64.1	66.9
79	54.4	55.9	58.0	59.8	61.8	65.0	67.9
80	55.2	56.7	58.9	60.7	62.7	65.9	68.9

(*table continues*)

If system with N = 10 trunks has P(B) = 0.01:

System can handle Offered traffic (A) = 4.14 E

Efficiency of Large Groups

- What if there are N = 100 trunks?
  - Will they serve A = 10 x 4.14 E = 41.4 E with same P(B) = 1%?
  - No!
  - Traffic tables will show that A = 78.2 E!
- Why will 10 times trunks serve almost 20 times traffic?
  - Called **efficiency of large groups**:

For N = 10, A = 4.14 E ⟶ $\rho = \frac{A}{N} = \frac{4.14}{10} = 41.4\%\ \text{efficiency}$

For N = 100, A = 78.2 E ⟶ $\rho = \frac{A}{N} = \frac{78.2}{100} = 78.2\%\ \text{efficiency}$

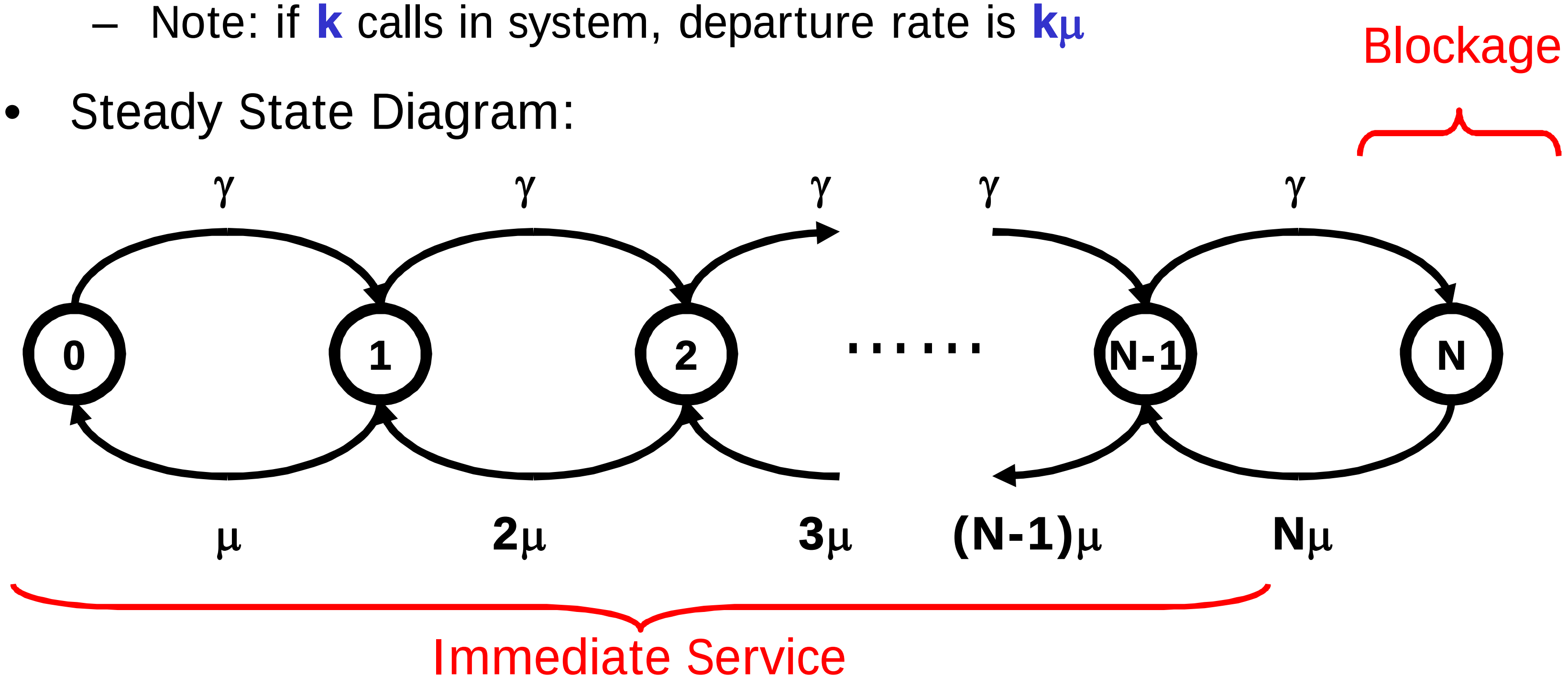
The larger the trunk group, the greater the efficiency

Erlang B Model

- More sophisticated model than Binomial or Poisson
- Blocked Calls Cleared (BCC)
- Good for calls that can reroute to alternate route if blocked
- No approximation for reattempts if alternate route blocked too
- Derived using **birth-death process**
  - See selected pages from Leonard Kleinrock, **Queueing Systems Volume 1: Theory**, John Wiley & Sons, 1975

Erlang B Birth-Death Process

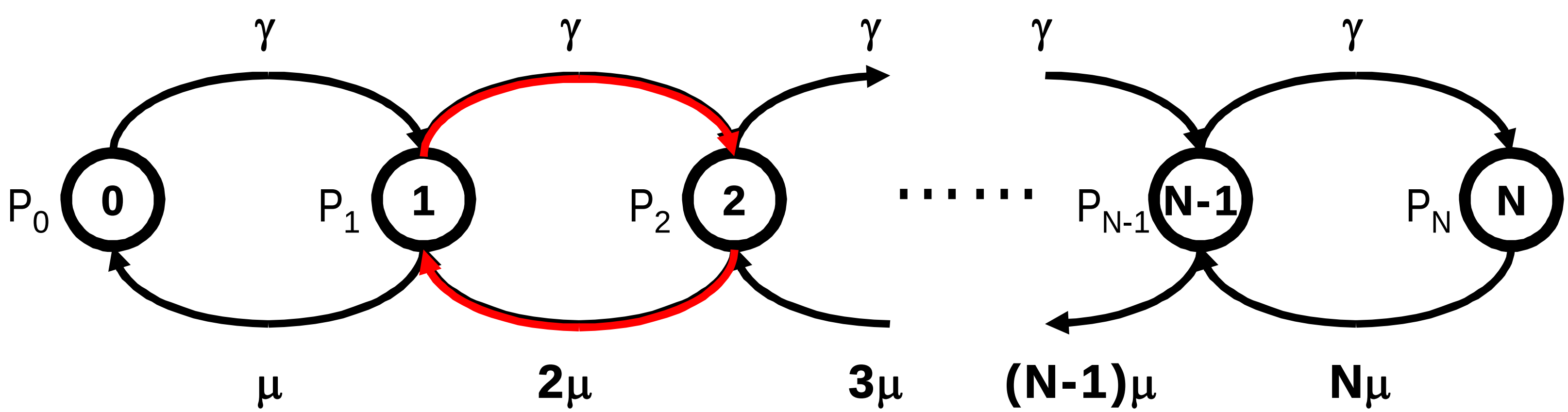
- Consider infinitesimally small time **δt** during which **only one** arrival or departure (or none) may occur
- Let **γ** be the arrival rate from an infinite pool or sources
- Let **μ = 1/h** be the departure rate per call
  - Note: if **k** calls in system, departure rate is **$k\mu$**
- Steady State Diagram:



Erlang B Birth-Death Process (2)

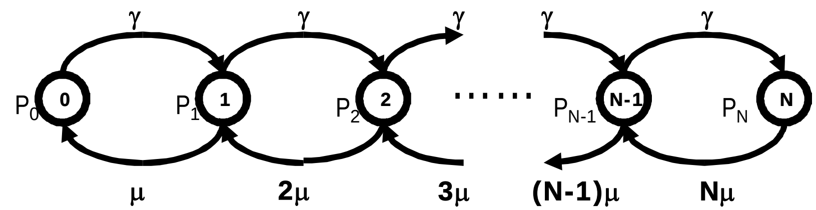
- Steady State (statistical equilibrium)
  - Rate of arrival is the same as rate of departure
  - Average rate a system enters a given state is equal to the average rate at which the system leaves that state

Probability of moving from state 1 to state 2? $\gamma \cdot P_1$



Probability of moving from state 2 to state 1? $2\mu \cdot P_2$

Erlang B Birth-Death Process (3)



- Set up balance equations:

$$\gamma P_0 = \mu P_1 \quad \longrightarrow \quad \gamma P_0 = \mu P_1 \quad \longrightarrow \quad P_1 = \frac{\gamma}{\mu} P_0$$

$$\mu P_1 + \gamma P_1 = 2\mu P_2 + \gamma P_0 \quad \longrightarrow \quad \gamma P_1 = 2\mu P_2 \quad \longrightarrow \quad P_2 = \frac{\gamma}{2\mu} P_1 = \left(\frac{\gamma}{\mu}\right)^2 \frac{P_0}{2}$$

$$2\mu P_2 + \gamma P_2 = 3\mu P_3 + \gamma P_1 \quad \longrightarrow \quad \gamma P_2 = 3\mu P_3 \quad \longrightarrow \quad P_3 = \frac{\gamma}{3\mu} P_2 = \left(\frac{\gamma}{\mu}\right)^3 \frac{P_0}{6}$$

$$3\mu P_3 + \gamma P_3 = 4\mu P_4 + \gamma P_2$$

$$\vdots$$

$$\gamma P_{k-1} = k\mu P_k$$

$$P_k = \left(\frac{\gamma}{\mu}\right)^k \frac{P_0}{k!}$$

$$\vdots$$

$$(N-1)\mu P_{N-1} + \gamma P_{N-1} = N\mu P_N + \gamma P_{N-2}$$

$$N\mu P_N = \gamma P_{N-1} \qquad \gamma P_{N-1} = N\mu P_N$$

Erlang B Birth-Death Process (4)

Rule of Total Probability:

$$\sum_{i=0}^{N} P_i = 1 = \sum_{i=0}^{N} \left(\frac{\gamma}{\mu}\right)^i \frac{P_0}{i!} \longrightarrow P_0 = \frac{1}{\sum_{i=0}^{N} \left(\frac{\gamma}{\mu}\right)^i \frac{1}{i!}}$$

Recall:

$$P_k = \left(\frac{\gamma}{\mu}\right)^k \frac{P_0}{k!}$$

Recall:

$$A = \gamma h = \frac{\gamma}{\mu}$$

$$P_k = \frac{A^k}{k!} \Bigg/ \sum_{i=0}^{N} \frac{A^i}{i!}$$

For blocking, must be in state k = N:

$$P(B) = B(N, A) = P_N = \frac{A^N}{N!} \Bigg/ \sum_{i=0}^{N} \frac{A^i}{i!}$$

"B" = Erlang B

"N" = # Servers

"A" = Offered Traffic

Erlang B Traffic Table

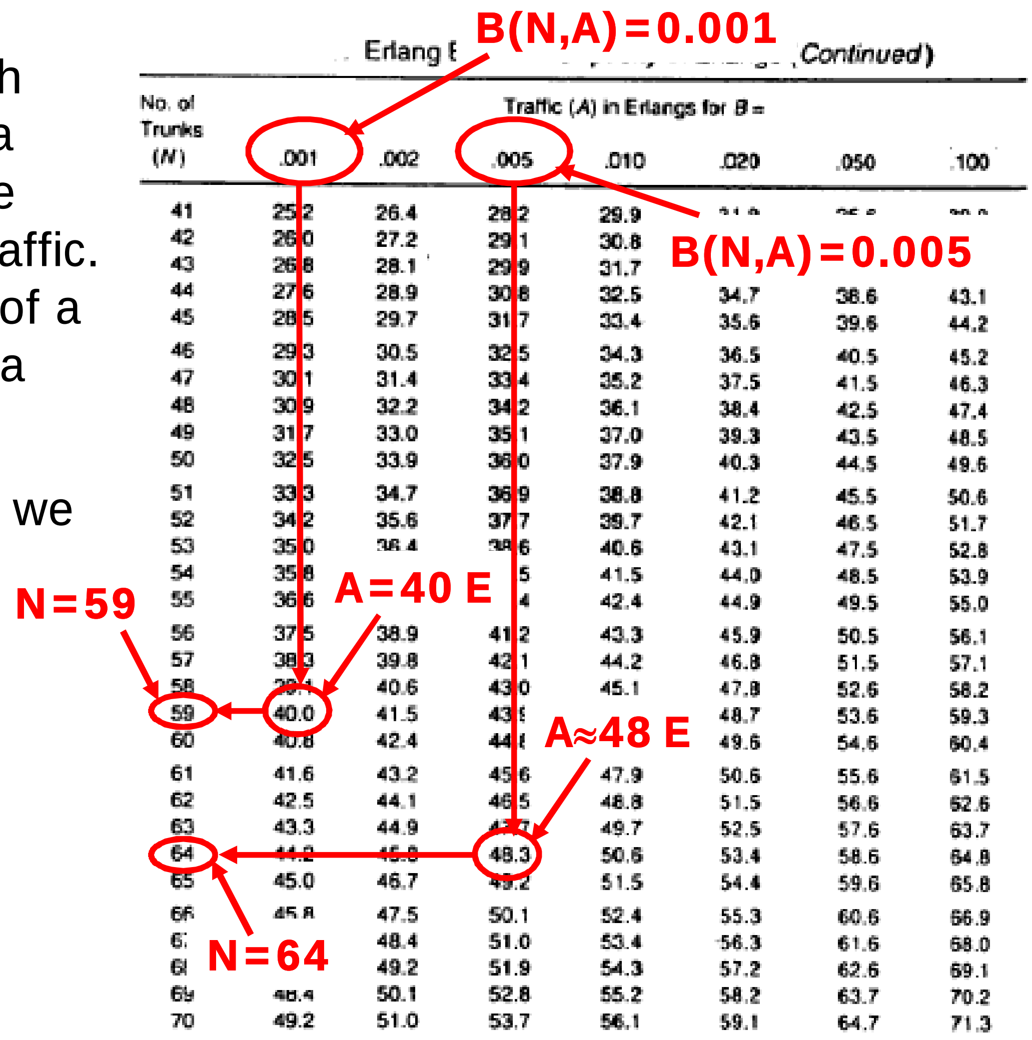
Example: In a BCC system with $m=\infty$ sources, we can accept a 0.1% chance of blocking in the nominal case of 40E offered traffic. However, in the extreme case of a 20% overload, we can accept a 0.5% chance of blocking.

How many outgoing trunks do we need?

Nominal design: 59 trunks

Overload design: 64 trunks

Requirement: 64 trunks



Erlang B [illegible] Continued)

No. of Trunks (N)	Traffic (A) in Erlangs for B = .001	.002	.005	.010	.020	.050	.100
41	25.2	26.4	28.2	29.9	[illegible]	[illegible]	[illegible]
42	26.0	27.2	29.1	30.8	[illegible]	[illegible]	[illegible]
43	26.8	28.1	29.9	31.7	[illegible]	[illegible]	[illegible]
44	27.6	28.9	30.8	32.5	34.7	38.6	43.1
45	28.5	29.7	31.7	33.4	35.6	39.6	44.2
46	29.3	30.5	32.5	34.3	36.5	40.5	45.2
47	30.1	31.4	33.4	35.2	37.5	41.5	46.3
48	30.9	32.2	34.2	36.1	38.4	42.5	47.4
49	31.7	33.0	35.1	37.0	39.3	43.5	48.5
50	32.5	33.9	36.0	37.9	40.3	44.5	49.6
51	33.3	34.7	36.9	38.8	41.2	45.5	50.6
52	34.2	35.6	37.7	39.7	42.1	46.5	51.7
53	35.0	36.4	38.6	40.6	43.1	47.5	52.8
54	35.8	[illegible]	[illegible]	41.5	44.0	48.5	53.9
55	36.6	[illegible]	[illegible]	42.4	44.9	49.5	55.0
56	37.5	38.9	41.2	43.3	45.9	50.5	56.1
57	38.3	39.8	42.1	44.2	46.8	51.5	57.1
58	[illegible]	40.6	43.0	45.1	47.8	52.6	58.2
59	40.0	41.5	[illegible]	[illegible]	48.7	53.6	59.3
60	40.8	42.4	[illegible]	[illegible]	49.6	54.6	60.4
61	41.6	43.2	45.6	47.9	50.6	55.6	61.5
62	42.5	44.1	46.5	48.8	51.5	56.6	62.6
63	43.3	44.9	[illegible]	49.7	52.5	57.6	63.7
64	[illegible]	[illegible]	48.3	50.6	53.4	58.6	64.8
65	45.0	46.7	[illegible]	51.5	54.4	59.6	65.8
66	45.8	47.5	50.1	52.4	55.3	60.6	66.9
67	[illegible]	48.4	51.0	53.4	56.3	61.6	68.0
68	[illegible]	49.2	51.9	54.3	57.2	62.6	69.1
69	[illegible]	50.1	52.8	55.2	58.2	63.7	70.2
70	49.2	51.0	53.7	56.1	59.1	64.7	71.3

Example (2)

P(N,A)=0.01

N=32

A=20.3 E

Poisson Traffic Capacity in Erlangs

No. of Trunks (*N*)	Traffic (*A*) in Erlangs for *P* = .001	.002	.005	.010	.020	.050	.100
1	.001	.002	.005	[illegible]	.021	.053	.106
2	.044	.065	.104	[illegible]	.214	.358	.531
3	.192	.244	.338	[illegible]	.567	.817	1.10
4	.428	.519	.673	[illegible]	1.02	1.37	1.75
5	.739	.868	1.08	[illegible]	1.55	1.97	2.44
6	1.11	1.27	1.54	[illegible]	2.11	2.61	3.14
7	1.52	1.72	2.04	[illegible]	2.69	3.28	3.89
8	1.97	2.21	2.57	[illegible]	3.31	3.97	4.67
9	2.45	2.72	3.13	[illegible]	3.94	4.69	5.42
10	2.97	3.28	3.72	[illegible]	4.61	5.42	6.22
11	3.50	3.82	4.32	[illegible]	5.31	6.17	7.03
12	4.03	4.40	4.94	[illegible]	6.00	6.92	7.83
13	4.61	5.00	5.58	[illegible]	6.69	7.69	8.64
14	5.19	5.61	6.23	[illegible]	7.42	8.47	9.47
15	5.78	6.23	6.89	[illegible]	8.14	9.25	10.3
16	6.42	6.87	7.57	[illegible]	8.89	10.1	11.1
17	7.03	7.52	8.25	[illegible]	9.64	10.8	12.0
18	7.67	8.17	8.94	[illegible]	10.4	11.6	12.8
19	8.31	8.84	9.65	[illegible]	11.1	12.4	13.7
20	8.97	9.52	10.4	[illegible]	11.9	13.3	14.5
21	9.61	10.2	11.1	[illegible]	12.7	14.1	15.4
22	10.3	10.9	11.8	[illegible]	13.5	14.9	16.3
23	11.0	11.6	12.5	[illegible]	14.3	15.7	17.1
24	11.6	12.3	13.3	[illegible]	15.1	16.6	18.0
25	12.3	13.0	14.0	[illegible]	15.9	17.4	18.8
26	13.0	13.7	14.7	[illegible]	16.6	18.2	19.7
27	13.8	14.4	15.5	[illegible]	17.4	19.1	20.6
28	14.4	15.2	16.3	[illegible]	18.2	19.9	21.5
29	15.1	15.9	17.0	[illegible]	19.0	20.8	22.4
30	15.9	16.6	17.8	[illegible]	19.9	21.6	23.2
31	16.6	17.4	18.5	19.5	20.7	22.5	24.1
32	[illegible]	[illegible]	[illegible]	20.3	21.5	23.3	25.0
33	18.1	18.9	20.1	21.1	22.3	24.2	25.9
34	18.8	19.6	20.9	21.9	23.1	25.1	26.8
35	19.5	20.4	21.6	22.7	23.9	25.9	27.7
36	20.3	21.1	22.4	23.5	24.8	[illegible]	[illegible]
37	21.0	21.9	23.2	24.3	25.6	[illegible]	[illegible]
38	21.8	22.7	24.0	25.1	26.4	[illegible]	[illegible]
39	22.5	23.4	24.8	26.0	27.3	29.4	31.3
40	23.3	24.2	25.6	26.8	28.1	30.2	32.1

(*table continues*)

Poisson Traffic Capacity in Erlangs (*Continued*)

No. of Trunks (*N*)	Traffic (*A*) in Erlangs for *P* = .001	.002	.005	.010	.020	.050	.100
41	24.0	25.0	26.4	27.6	28.9	31.1	33.1
42	24.8	25.8	27.2	28.4	29.8	32.0	33.9
43	25.5	26.5	28.0	29.2	30.6	32.9	34.9
44	26.3	27.3	28.8	30.1	31.5	33.7	35.8
45	27.1	28.1	29.6	30.9	32.3	34.6	36.7
46	27.9	28.9	30.4	31.7	33.2	35.5	37.6
47	28.6	29.7	31.2	32.5	34.0	36.4	38.5
48	29.4	30.5	32.0	33.4	34.9	37.2	39.4
49	30.2	31.3	32.9	34.2	35.7	38.1	40.3
50	31.0	32.1	33.7	35.0	36.6	39.0	41.2
51	31.8	32.9	34.5	35.9	37.4	39.9	42.1
52	32.5	33.7	35.3	36.7	38.3	40.8	43.0
53	33.3	34.5	36.1	37.6	39.2	41.6	43.9
54	34.1	35.3	37.0	38.4	40.0	42.5	44.8
55	34.9	36.1	37.8	39.2	40.9	43.4	45.7
56	35.7	36.9	38.6	40.1	41.8	44.3	46.6
57	36.5	37.7	39.3	40.9	42.6	45.2	47.6
58	37.3	38.5	40.3	41.8	43.5	46.1	48.5
59	38.1	39.3	41.1	42.6	44.3	47.0	49.4
60	38.9	40.1	41.9	43.5	45.2	47.9	50.3
61	39.7	40.9	42.8	44.3	46.1	48.8	51.2
62	40.5	41.8	43.6	45.2	46.9	49.6	52.1
63	41.3	42.6	44.4	46.0	47.8	50.5	53.1
64	42.1	43.4	45.3	46.9	48.7	51.4	54.0
65	42.9	44.2	46.1	47.7	49.6	52.3	54.9
66	43.7	45.0	46.9	48.6	50.4	53.2	55.8
67	44.5	45.9	47.8	49.4	51.3	54.1	56.7
68	45.3	46.7	48.7	50.3	52.2	55.0	57.7
69	46.1	47.5	49.5	51.2	53.1	55.9	58.6
70	47.0	48.4	50.3	52.0	53.9	56.8	59.5
71	47.8	49.2	51.2	52.9	54.8	57.7	60.4
72	48.6	50.0	52.0	53.8	55.7	58.6	61.4
73	49.4	50.8	52.9	54.6	56.6	59.5	62.3
74	50.3	51.7	53.7	55.5	57.4	60.4	63.2
75	51.1	52.5	54.6	56.3	58.3	61.3	64.1
[illegible]	51.9	53.4	55.4	57.2	59.2	62.3	65.1
[illegible]	52.7	54.2	56.3	58.1	60.1	63.2	66.0
[illegible]	53.5	55.0	57.1	58.9	60.9	64.1	66.9
79	54.4	55.9	58.0	59.8	61.8	65.0	67.9
80	55.2	56.7	58.9	60.7	62.7	65.9	68.9

(*table continues*)

P(N,A) & B(N,A) - High Blocking

- We recognize that Poisson and Erlang B models are only approximations but which is better?
  - Compare them using a 4-trunk group offered A=10E

Erlang B

$$B(4,10) = 0.64666$$

$$T_C = A \times (1 - P(B)) = 10 \times (1 - 0.64666)$$

$$T_C = 3.533E$$

$$\rho = \frac{3.533}{4} = 0.88$$

Poisson

$$P(4,10) = 0.98966$$

$$T_C = A \times (1 - P(B)) = 10 \times (1 - 0.98966)$$

$$T_C = 0.103E$$

$$\rho = \frac{0.103}{4} = 0.026$$

How can 4 trunks handle 10E offered traffic and be busy only 2.6% of the time?

P(N,A) & B(N,A) - High Blocking (2)

- Obviously, the Poisson result is so far off that it is almost meaningless as an approximation of the example.
  - 4 servers offered enough traffic to keep 10 servers busy full time (10E) should result in much higher utilization.
- Erlang B result is more believable.
  - All 4 trunks are busy most of the time.
- What if we extend the exercise by increasing A?
  - Erlang B result goes to 4E carried traffic
  - Poisson result goes to 0E carried
- Illustrates the failure of the Poisson model as valid for situations with high blocking
  - Poisson only good approximation when low blocking
  - Use Erlang B if high blocking

Engset Distribution Model

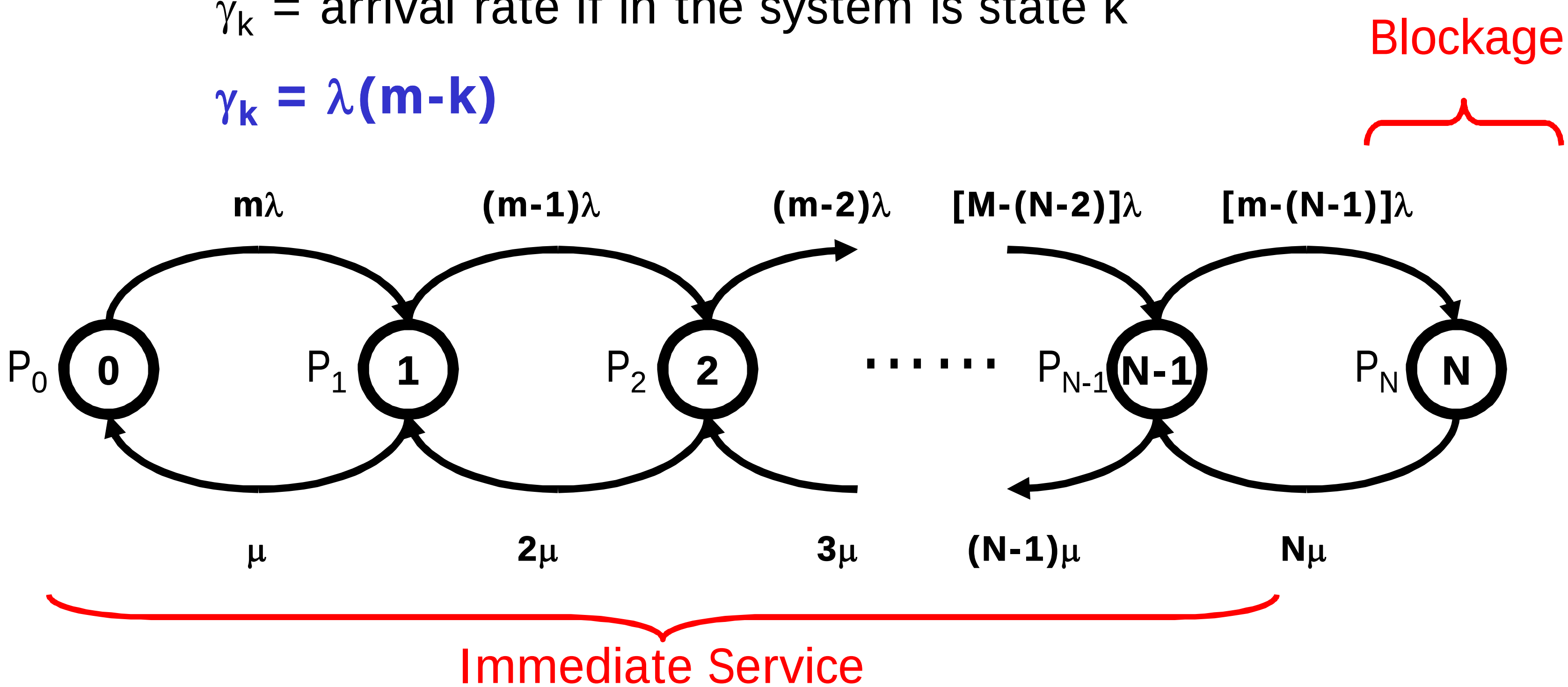
- BCC model with small number of sources (m > N)

  μ = mean departure rate per call

  λ = mean arrival rate of a single source

  γ_k = arrival rate if in the system is state k

  $\gamma_k = \lambda(m\text{-}k)$



Engset Traffic Model (2)

- Balance equations give:

$$P_k = P_0 \left(\frac{\lambda}{\mu}\right)^k \frac{m!}{k!(m-k)!}$$

and

$$P_0 = \frac{1}{\sum_{i=0}^{N} \left(\frac{\lambda}{\mu}\right)^i \binom{m}{i}}$$

therefore:

$$P_k = \frac{\left(\frac{\lambda}{\mu}\right)^k \binom{m}{k}}{\sum_{i=0}^{N} \left(\frac{\lambda}{\mu}\right)^i \binom{m}{i}}$$

but can show that:

$$\frac{\lambda}{\mu} = \frac{A}{m-A}$$

$$P(B) = P(k = N) = E(m, N, A) = \frac{\left(\frac{A}{m-A}\right)^N \binom{m}{N}}{\sum_{i=0}^{N} \left(\frac{A}{m-A}\right)^i \binom{m}{i}}$$

"E" = Engset

Engset Traffic Table

M = 30 sources

 # trunks (N)

Traffic offered (A)

Engset Loss Probability (M = 30 Sources)

Engset Loss Probability (M = 30 Sources Continued)

P(B)=E(m,N,A)

N=10

A=4.8 E

P(B)<0.01

NB: These tables are based on Engset call-congestion.

Example: 30 terminals each provide 0.16 Erlangs to a concentrator with a goal of less than 1% blocking.

How many outgoing trunks do we need?

A = 30 x 0.16 = 4.8 E

Requirement: **N = 10 Trunks**

Check m < 10 x N?

M=30 < 10 x 10 = 100

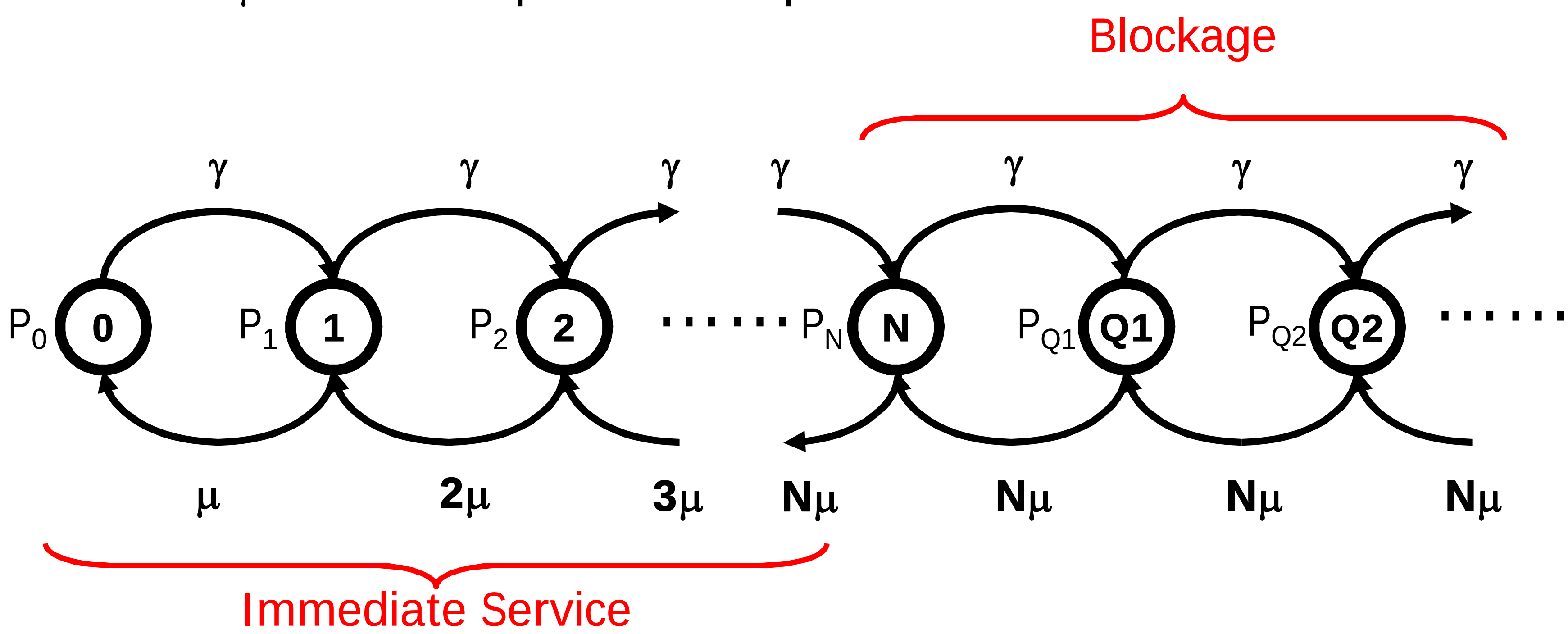
(table continues)

Erlang C Distribution Model

- BCW model with infinite sources (m) and infinite queue length

  γ = arrival rate of new calls

  μ = mean departure rate per call



Erlang C Distribution Model (2)

- Balance equations give:

$$P_k = \frac{A^k P_0}{k!}, \quad k \le N \qquad \text{and} \qquad P_k = \frac{A^k P_0}{N^{k-N} N!}, \quad k \ge N \qquad \text{and} \qquad P_0 = \frac{1}{\frac{A^N}{N!}\left(\frac{N}{N-A}\right) + \sum_{i=0}^{N-1} \frac{A^i}{i!}}$$

- But P(B) = P(k≥N):

$$P(B) = \sum_{k=N}^{\infty} \frac{A^k P_0}{N^{k-N} N!} = \sum_{k=N}^{\infty} \left(\frac{A}{N}\right)^k \frac{P_0}{N^{-N} N!} = \frac{A^N}{N!} P_0 \sum_{k=0}^{\infty} \left(\frac{A}{N}\right)^k$$

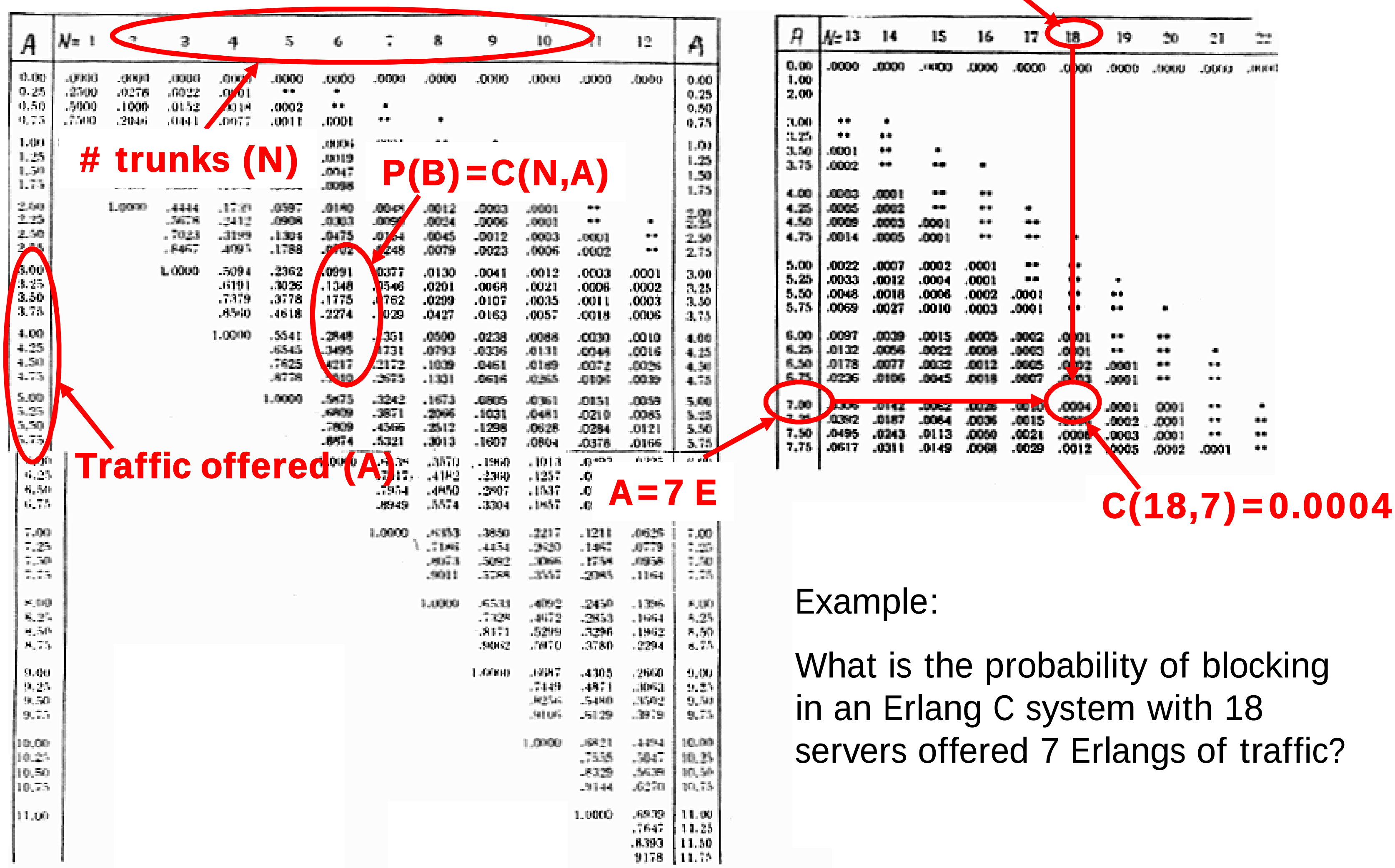
but can show that:

$$\sum_{k=0}^{\infty} \left(\frac{A}{N}\right)^k = \frac{N}{N-A}$$

$$P(B) = \frac{A^N}{N!} \frac{N}{N-A} P_0 \longrightarrow C(N,A) = \frac{\frac{A^N}{N!} \frac{N}{N-A}}{\frac{A^N}{N!}\left(\frac{N}{N-A}\right) + \sum_{i=0}^{N-1} \frac{A^i}{i!}}$$

"C" = Erlang C

Erlang C Traffic Tables



Example:

What is the probability of blocking in an Erlang C system with 18 servers offered 7 Erlangs of traffic?

Delay in Erlang C

- Expected number of calls in the queue?

$$= \sum_{k=N}^{\infty} (k-N) P_k = \sum_{k=N}^{\infty} (k-N) \frac{A^k}{N^{k-N} N!} P_0 = \frac{A^N}{N!} P_k \sum_{k=0}^{\infty} k \left(\frac{A}{N} \right)^k$$

$$= \frac{P_0 A^N}{N!} \frac{A}{N-A} \frac{N}{N-A} = \frac{A \cdot C(N,A)}{N-A} = \frac{\gamma h}{N-A} C(N,A)$$

$$\text{Mean Delay over All Calls} = \frac{\text{Mean \#Calls Delayed}}{\text{Arrival Rate of Calls}} = \frac{h}{N-A} C(N,A)$$

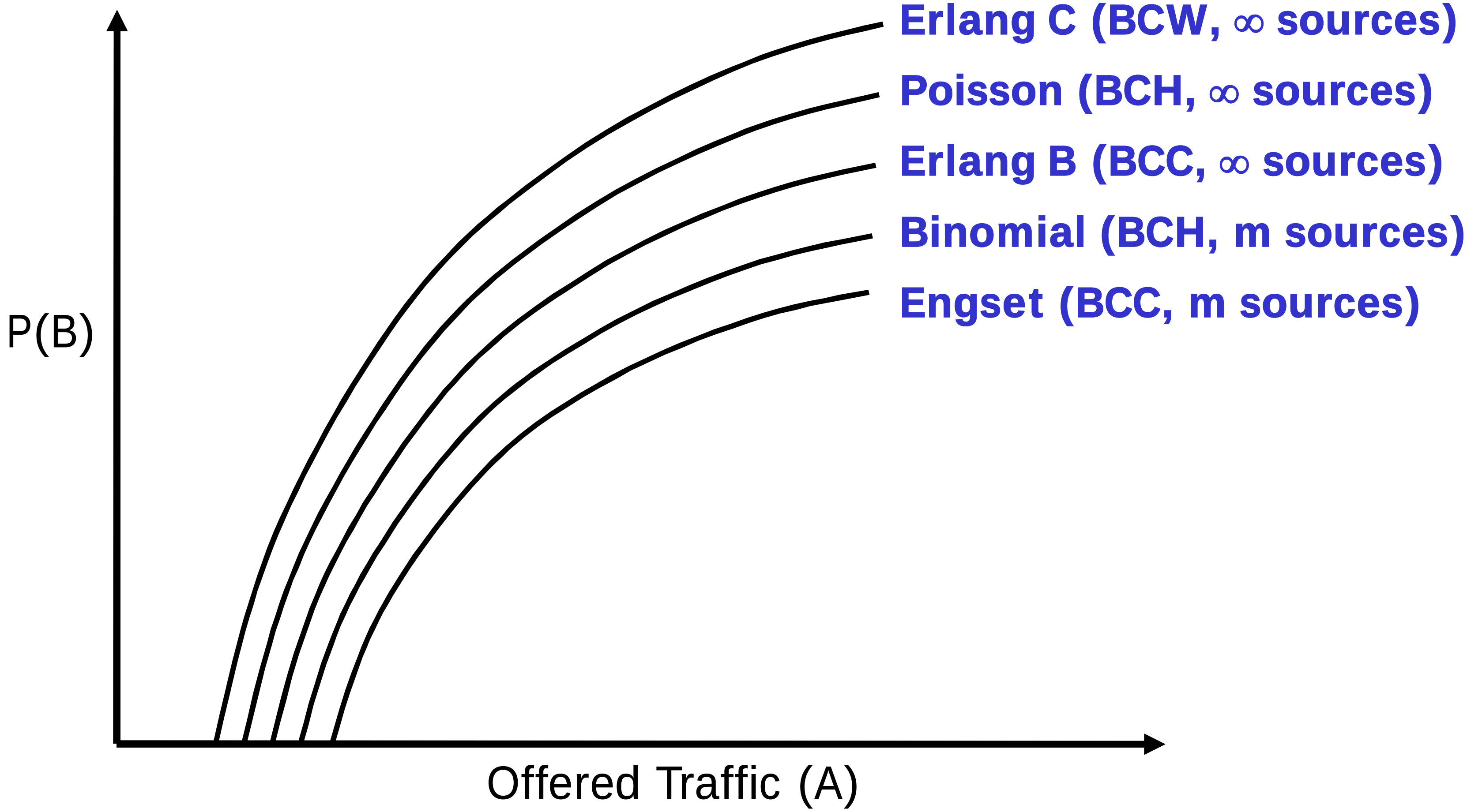
Recall: $\gamma = \frac{\alpha}{T}$ ↓ $T = \frac{\alpha}{\gamma}$

$$\text{Mean Delay of Delayed Calls} = \frac{h}{N-A}$$

Also:

$$P(delay > T) = C(N,A) e^{-T \Big/ \frac{h}{N-A}}$$

Comparison of Traffic Models



Efficiency of Large Groups

- Already seen that for same P(B), increasing servers results in more than proportional increase in traffic carried

example 1: $P(10, 4.14) = 0.01$ and $P(100, 78.2) = 0.01$

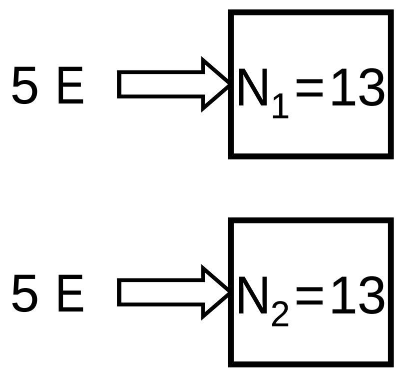
example 2: $P(32, 20.3) = 0.01$ and $P(33, 20.1) = 0.005$

example 3: $B(8, 2.05) = 0.001$ and $B(80, 57.8) = 0.001$

- What does this mean?
  - If it's possible to collect together several diverse sources, you can
    - provide better gos at same cost, or
    - provide same gos at cheaper cost

Efficiency of Large Groups (2)

- Two trunk groups offered 5 Erlangs each, and B(N,A)=0.002



How many trunks total?

From traffic tables, find B(13,5) $\approx$ 0.002

N_{total} = 13 + 13 = 26 trunks

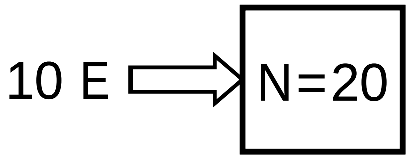
Trunk efficiency?

$$\rho = \frac{T_C}{N} = \frac{10(1-0.002)}{26} = 0.384$$

38.4% utilization

Efficiency of Large Groups (3)

- One trunk group offered 10 Erlangs, and B(N,A)=0.002



How many trunks?

From traffic tables, find B(20,10) ≈ 0.002

N = 20 trunks

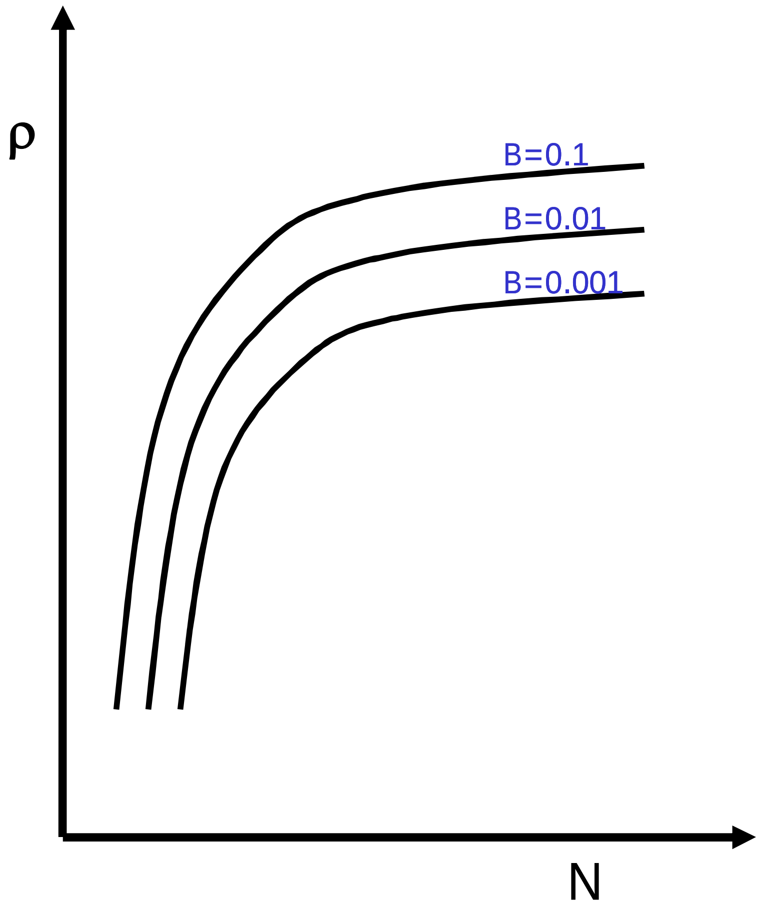
Trunk efficiency?

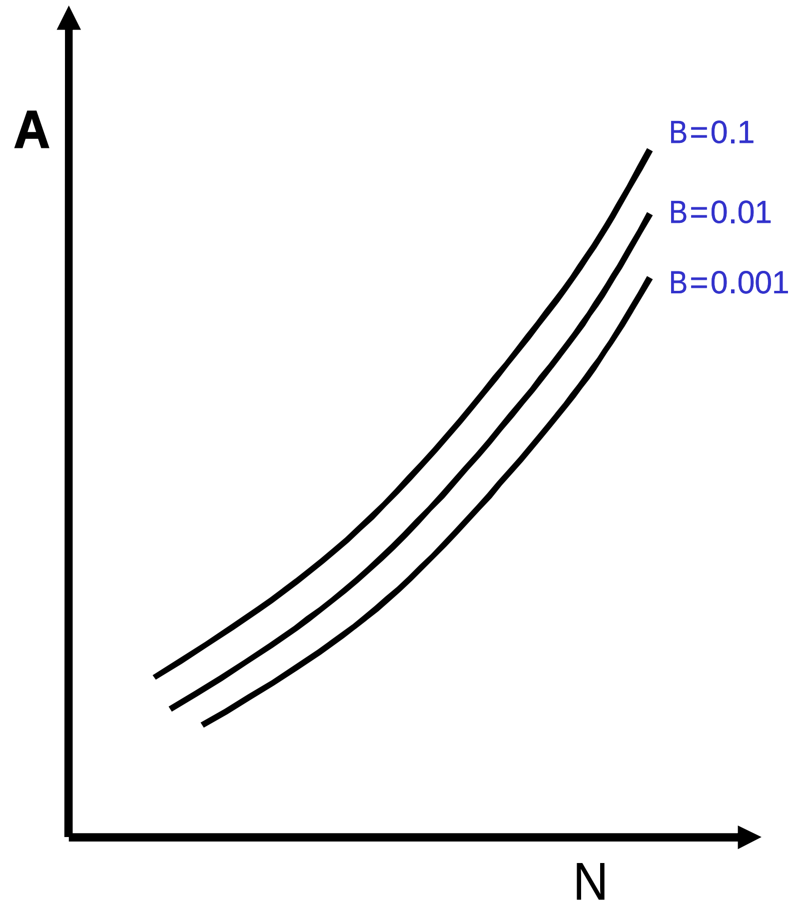
$$\rho = \frac{T_C}{N} = \frac{10(1-0.002)}{20} = 0.499$$

→ **49.9% utilization**

For same gos, we can save 6 trunks!

Efficiency of Large Groups (4)





Sensitivity to Overload

- Consider 2 cases:

Case 1: N = 10 and B(N,A) = 0.01

B(10,4.5) ≈ 0.01, so can carry 4.5 E

What if 20% overload (5.4 E)? ⟶ B(10,5.4) ≈ 0.03

3 times P(B) with 20% overload

Case 1: N = 30 and B(N,A) = 0.01

B(30,20.3) ≈ 0.01, so can carry 20.3 E

What if 20% overload (24.5 E)? ⟶ B(30,24.5) ≈ 0.08

8 times P(B) with 20% overload!

“Trunk Group Splintering”

- if high possibility of overloads, small groups may be better

Incremental Traffic Carried by Nth Trunk

- If a trunk group is of size N-1, how much extra traffic can it carry if you add one extra trunk?
  - Before, can carry: T_{C1} = A x [1-(B(N-1,A)]
  - After, can carry: T_{C2} = A x [1-(B(N,A)]

$$A_N = T_{C2} - T_{C1} = A\left(\left[1 - B(N,A)\right] - \left[1 - B(N-1,A)\right]\right)$$

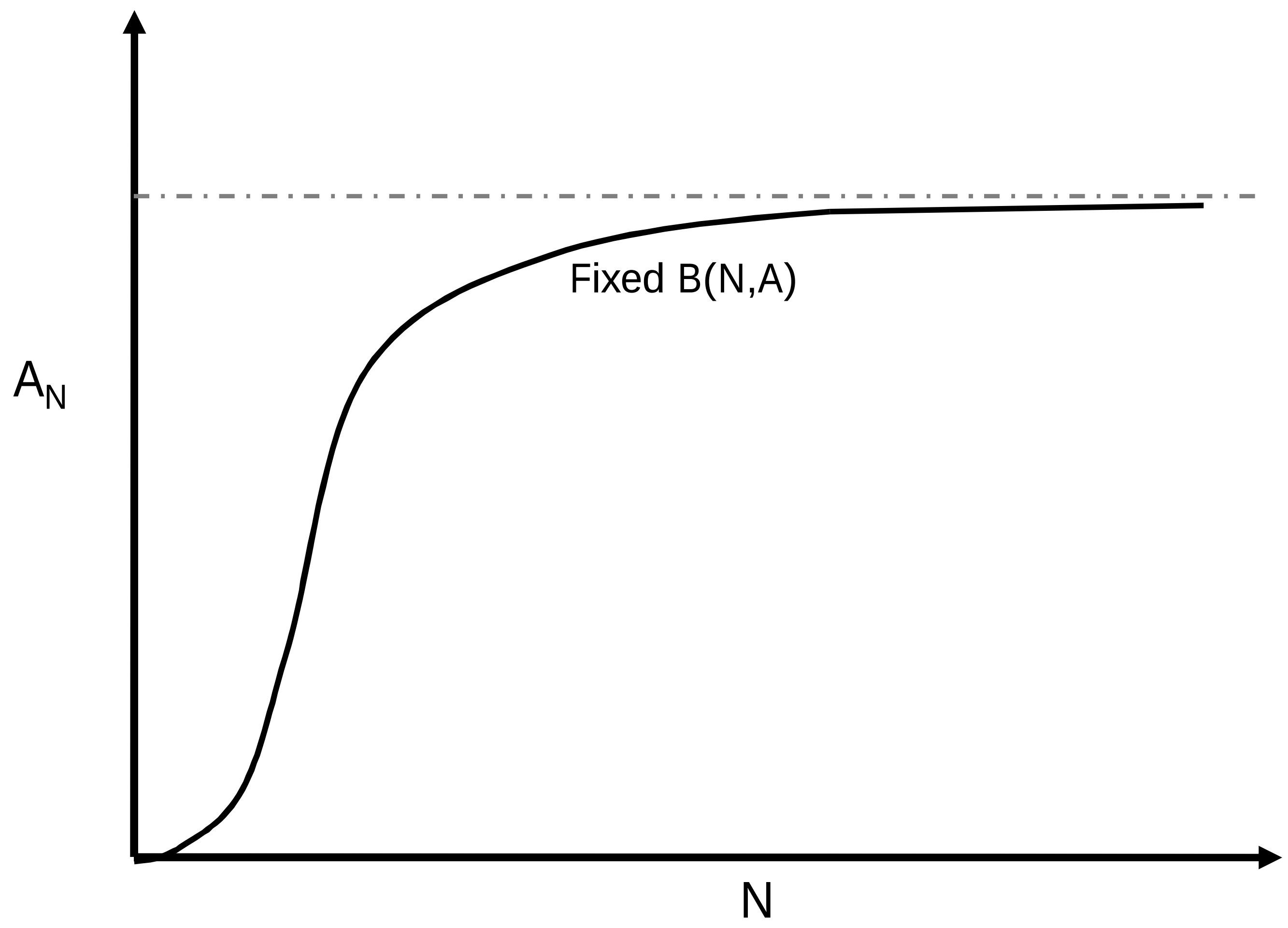
$$= A\left(B(N-1,A) - B(N,A)\right)$$

$$A_N \approx (N-A) \times B(N,A)$$

for very low blocking

- What does this mean?
  - **Random Hunting**: Increase in trunk group's total carried traffic after adding an Nth trunk
  - **Sequential Hunting**: **Actual** traffic carried by the Nth trunk in the group

Incremental Traffic Carried by N^{th} Trunk (3)



Example

- Individual trunks are only economic if they can carry 0.4 E or more. A trunk group of size N=10 is offered 6 E. Will all 10 trunks be economical?

$$A_N = A\left(B(N-1, A) - B(N, A)\right)$$

$$A_{10} = 6\left(B(9,6) - B(10,6)\right)$$

$$= 6\left(0.07514 - 0.04314\right)$$

$$= 0.192 \text{ E} < 0.4 \text{ E}$$

$\therefore$ At least the 10th trunk is not economical