Derivation of Erlang B formula:

Consider a system with C channels and U users. Then

$A_u = H\lambda_1$ erlangs, where λ_1 is average call arrival rate for one user, and

$A = UA_u = H\lambda$:- $\lambda = U\lambda_1$, the average call arrival rate in the system.

1. Call arrivals is a random process in time and it is modelled as a *Poisson process*.

A random variable x is called Poisson distributed with parameter α if x takes on values 0, 1, 2, …, n, … with a probability

$P\{x = k\} = e^{-\alpha}\ \alpha^k/k!$

Hence the probability density function is

$$f(x) = e^{-\alpha} \sum_{k=0}^{\infty} (\alpha^k/k!)\delta(x-k)$$

with $p_k = P\{x=k\}$, and

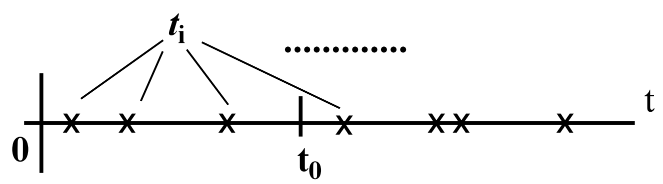
$p_k / p_{k+1} = (k+1)/\alpha$.

p_k reaches to maximum at $k=[\alpha]$;

if $\alpha < 1$, p_k is max at $k=0$,
if $\alpha > 1$ and α is not an integer, p_k is max at $k= [\alpha]$,
if $\alpha > 1$ and α is an integer, p_k is max at $k= \alpha$ and $k= \alpha -1$.

Poisson points:

Instants of call arrivals t_i are a set of points on t-axis:



Given a t_0 we define RV $a(t_0)$ as

$a(t_0)$ = total number of arrival intants t_i in interval $(0, t_0)$ with

$P\{ a(t_0) = k\}= \exp(-\lambda t_0)[(\lambda t_0)^k/k!]$.

Hence, number of points in an interval of length t_0 is also a Poisson distributed RV with parameter $\alpha=\lambda t_0$, where λ is the density of points (call arrivals).

If the random process $a(t)$ is the total number of call arrivals until time t, then, the probability of having n new arrivals in $(t, t+\tau)$ is

$P\{ a(t+\tau) - a(t)=n\}= e^{-\lambda\tau} [(\lambda\tau)^n/n!]$

where τ is a time interval.

2. User service time H: Average duration of calls (or *average holding time*).

Service times are modelled by a RV which is exponetially distributed with mean duration of H. The probability that the service time s_n of n^{th} user is less than some call duration s is modelled as

$$P\{s_n < s\} = 1- \exp(-\mu s) \qquad \text{for } s>0$$
$$\approx \mu s \qquad \text{if } \mu s <<1.$$

with μ= 1/H, the *mean service rate*, and

$f(s)= \mu \exp(-\mu s)$.

The duration of every call is independent from the other. Therefore

P{ the duration of at least one call, out of i current calls in the system, is less than s} $= P\{s_1 < s\}+ P\{s_2 < s\}+\ldots+ P\{s_i < s\}$

$$\approx i\mu s \qquad \text{for small } \mu s.$$

3. A trunked system:

The opereation of a trunked system is a *continuous time – discrete state* random process:

Continuous-time: call can arrive at any time
Discrete-state: there can be discrete number of users; state of the system *at any time instant* is the number of current users talking in the system.

This process can be modelled (an approximation) as a continuous time-discrete state *Markov* process.

We can sample this system at every δ seconds by sampling the number of current users. If we take these samples as the state the system, then the system is discretized on time axis also. Operation becomes:

Discrete time- discrete state Markov process.

A further simplification in modelling this operation is possible when δ is decreased. Assume that we keep δ sufficiently small such that, only a single one of the following events can take place within this interval:

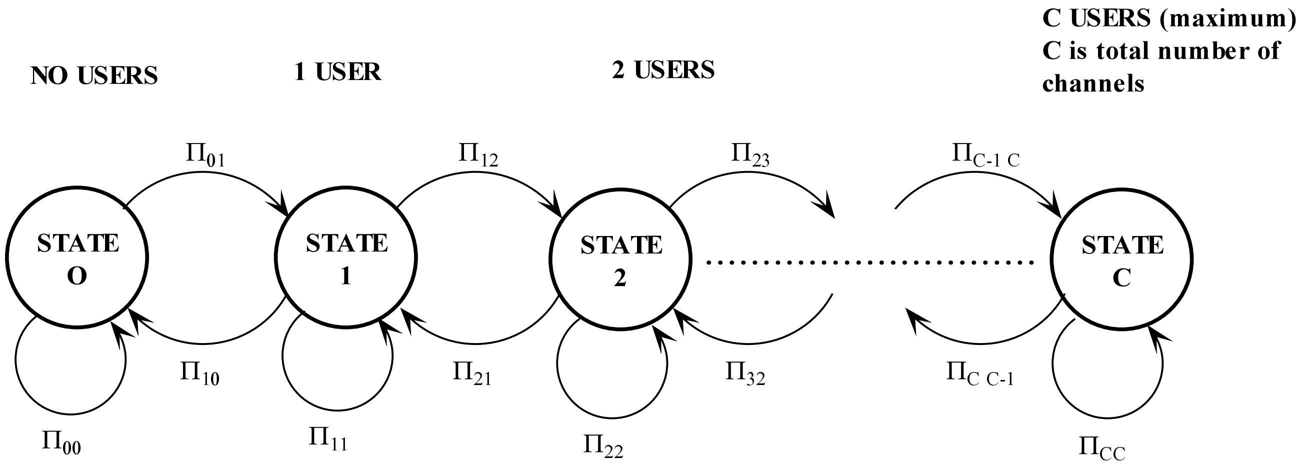
- **A current call can finish, i.e. $N_k = i$, $N_{k+1} = i-1$;**
- **Nothing happens, i.e. $N_k = i$, $N_{k+1} = i$;**
- **A new call arrives, i.e. $N_k = i$, $N_{k+1} = i+1$.**

The next state can be at most one user more or one user less (put in more rigorous terms, the probability

1-P{ next state can be at most one user more or one user less } is insignificant). Now, let

> **$N_k = N(k\delta)$ is the number of occupied channels in the system (or the number of users in the system) at $t = k\delta$,**
>
> **N_{k+1} can only be equal to $N_k - 1$, N_k, $N_k + 1$.**

Graphically,



At any instant, state of the system can only be in any one of the nodes (circles). Π_{ij} are called the *transition probabilities* of the process, where i,j= 0, 1, 2,..., C.

A Markov process in most general sense (continuous-time continuous-state) is a stochastic process whose past has no influence on the future, *if its present state is specified*. A discrete-time *Markov chain* is a discrete Markov process which has *countable number of states*, as in this case. A Markov chain is specified in terms of its *state probabilities*

$p_j[k] = P\{N_k = j\}$ j=0, 1, 2, ..., C (the value of the state)

and the transition probabilities

$\Pi_{ij}[k,l] = P\{ N_k = j \mid N_l = i\}$ i,j=0, 1, 2, ..., C (values of states N_l and N_k) and k, l are time instants t= kδ and t= lδ.

$\Pi_{ij}[k,l] = P\{ N_k = j \mid N_l = i\}$ is the probability of transition from state i to another state. If we sum these up over all states (over j=0, 1, ..., i, ..., C)

$\sum_j \Pi_{ij}[k,l] = 1.$

Also, since

$P\{ N_k = j \mid N_l = i\} P\{N_l = i\} = P\{ N_k = j, N_l = i\},$

$\sum_i p_i[l]\,\Pi_{ij}[k,l] = \sum_i P\{N_l=i\}\, P\{ N_k = j \mid N_l=i\}$

$= \sum_i P\{ N_k = j\, ,\, N_l=i\}$

$= P\{N_k=j\}$

$= p_j[k].$

We must determine the transition probabilities. Recall that

$P\{ a(t+\delta) - a(t)=n\}= e^{-\lambda\delta}\, [(\lambda\delta)^n/n!],$

and

$P\{s_n < \delta\}= 1- e^{-\mu\delta}$ **. Then**

$\Pi_{00}[k+1, k]=$ **P{there are no call arrivals during time interval** $[k\delta, (k+1)\delta]$**,
given that the state was** $N_k=0$ **at** $t= k\delta$**}
is simply equal to the probability of having no call arrivals in** $[k\delta, (k+1)\delta]$**,
or**

$\Pi_{00}[k+1, k] = e^{-\lambda\delta}\, [(\lambda\delta)^0/0!]= e^{-\lambda\delta}$

$\approx 1-\lambda\delta$ **for small** $\lambda\delta$**.**

We shall denote $\Pi_{00}[k+1, k]$ **by** Π_{00} **only from here onwards.**

Similarly,

$\Pi_{01}= \Pi_{01}[k+1, k]= P\{ N_{k+1} =1 \mid N_k=0\}$

$=$ **P{ state was 0 at** $t=k\delta$ **and one call arrived in** $[k\delta, (k+1)\delta]$**}**

$= e^{-\lambda\delta}\, [(\lambda\delta)^1/1!]$

$= (\lambda\delta)e^{-\lambda\delta} \approx (\lambda\delta)(1-\lambda\delta)$

$\approx \lambda\delta$ **for small** $\lambda\delta$**.**

$\Pi_{10}= \Pi_{10}[k+1, k]= P\{ N_{k+1} =0 \mid N_k=1\}$

$=$ **P{state was 1 and *that* call finished in** $[k\delta, (k+1)\delta]$**}**

$= 1- e^{-\mu\delta}$

$\approx \mu\delta$ **for** $\mu\delta \ll 1$**.**

Similarly, around state i,

$\Pi_{i,i+1}= P\{ N_{k+1}=i+1 \mid N_k=i\} \approx \lambda\delta,$

and

$\Pi_{i,i-1}$= P{ N_{k+1}=i-1 | N_k=i}=P{ ***one call out of i calls*** **finished in** $[k\delta, (k+1)\delta]$}

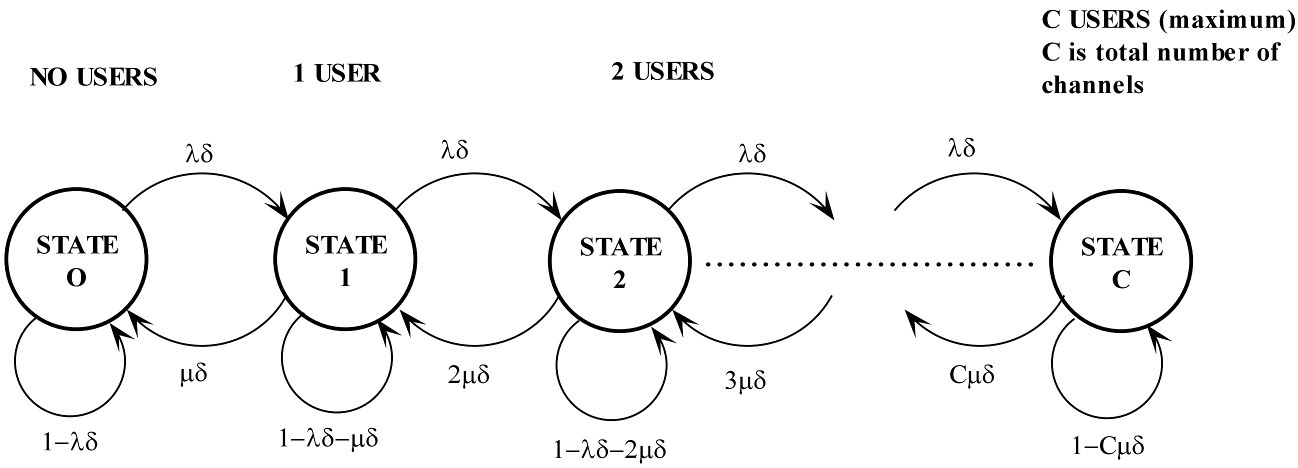
$$= P\{s_1 < \delta\} + P\{s_2 < \delta\} + \ldots + P\{s_i < \delta\}$$
$$= i\,(1 - e^{-\mu\delta})$$
$$\approx i\mu\delta$$

for $\mu\delta$ <<1, since the events are independent.

We know that the total out-going probability from a node must sum up to unity in a Markov chain:

$$\Pi_{i,i+1} + \Pi_{i,i} + \Pi_{i,i-1} = 1 \Rightarrow \Pi_{i,i} = 1 - \Pi_{i,i+1} + \Pi_{i,i-1}$$
$$= 1 - \lambda\delta - i\mu\delta \,.$$

We can now redraw the Markov chain as



But a Markov process is also Markov if the time is reversed:

$$P\{N_{k+1} | N_k\}\, P\{N_k\} = P\{N_{k+1}, N_k\} = P\{N_k | N_{k+1}\}\, P\{N_{k+1}\}.$$

Now, for any $n \leq C$

$$P\{N_k = n-1\}\, P\{N_{k+1} = n | N_k = n-1\} = P\{N_k = n-1\}(\lambda\delta)$$
$$= P\{N_{k+1} = n\}\, P\{N_k = n-1 \mid N_{k+1} = n\} = P\{N_{k+1} = n\}(n\mu\delta)$$

$$\Rightarrow \quad P\{N_k = n-1\}(\lambda\delta) = P\{N_{k+1} = n\}(n\mu\delta), \text{ or}$$
$$P_{n-1}(\lambda\delta) = P_n\,(n\mu\delta)$$

$$\Rightarrow \quad \lambda P_{n-1} = n\mu P_n \,.$$

Thus

$$P_1 = (\lambda/\mu)\, P_0$$

$$P_2 = (\lambda/2\mu)\, P_1 = (1/2)(\lambda/\mu)^2\, P_0$$

.

.

.

$$P_n = (\lambda/n\mu)\, P_{n-1} = (\lambda/\mu)^n\, P_0 / n!$$

We also know that

$$\sum_{i=0}^{C} P_i = 1 \,.$$

Substituting P_i in Σ:

$$\sum_{i=0}^{C} (\lambda/\mu)^i\, P_0 / i! = 1 \quad \Rightarrow \quad P_0 = \{\sum_{i=0}^{C} (\lambda/\mu)^i / i!\}^{-1}$$

Probability of having all channels occupied is:

$$P_C = (\lambda/\mu)^C\, P_0 / C! = \{(\lambda/\mu)^C / C!\} / \{\sum_{i=0}^{C} (\lambda/\mu)^i / i!\}$$

The total offered trafic is $A = \lambda H = \lambda/\mu$ Erlangs. Then

$$P_C = \{(A)^C / C!\} / \{\sum_{i=0}^{C} (A)^i / i!\} = GOS = \text{Probability of blocking}$$

The Erlang B formula!