King Fahd University of Petroleum & Minerals Computer Engineering Dept

COE 540 – Computer Networks

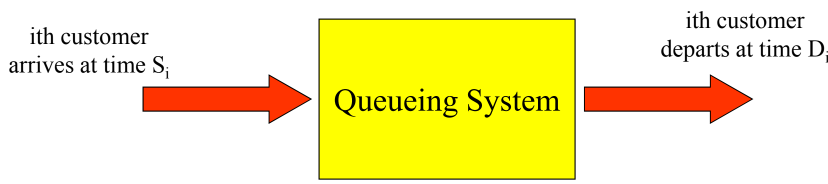
Term 101

Dr. Ashraf S. Hasan Mahmoud
Rm 22-148-3
Ext. 1724
Email: ashraf@kfupm.edu.sa

Queuing Model

- **Consider the following system:**

$A(t)$ $\qquad N(t) = A(t) - D(t)$ $\qquad D(t)$



$T_i = D_i - A_i$ $\qquad W_i = T_i - S_i$

$= D_i - A_i - S_i$

A(t) – number of arrivals in (0, t]
D(t) – number of departures in (0, t]
N(t) – number of customers in system in (0,t]
T_i – duration of time spent in system for ith customer
W_i – duration of time spent waiting for service for ith customer

Example 1: Queueing System

Problem: A data communication line delivers a block of information every 10 microseconds. A decoder check each block for errors and corrects the errors if necessary. It takes 1 microsecond to determine whether the block has any errors. If the block has one error it takes 5 microseconds to correct it and it has more than 1 error it takes 20 microseconds to correct the error. Blocks wait in the queue when the decoder falls behind. Suppose that the decoder is initially empty and that the number of errors in the first 10 blocks are: 0, 1, 3, 1, 0, 4, 0, 1, 0, 0.

a) Plot the number of blocks in the decoder as a function of time.

b) Find the mean number of blocks in the decoder

c) What percent of the time is the decoder empty?

Example 1: Queueing System – cont'd

Solution:

Interarrival time = 10 μsec

Service time = 1 if no errors
1+5 if 1 error
1+20 if more than 1 error

The queue parameters (A, D, S, and W) are shown below:

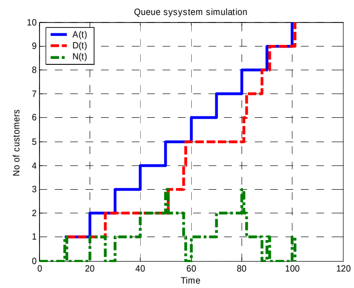
Block #:	1	2	3	4	5	6	7	8	9	10
Arrivals:	10	20	30	40	50	60	70	80	90	100
Errors:	0	1	3	1	0	4	0	1	0	0
Service:	1	6	21	6	1	21	1	6	1	1
Departs:	11	26	51	57	58	81	82	88	91	101
Waiting:	0	0	0	11	7	0	11	2	0	0
Total:	1	6	21	17	8	21	12	8	1	1

Example 1: Queueing System – cont'd

<u>Solution:</u>
Using the previous results and knowing that
N(t) = A(t) – D(t)
One can produce the following results

T_bar = 96/10 = 9.6μsec
Lambda = 10/101
N_bar = Lambda * T_bar
= 96/101 = 0.950

```
Average no of customers in system =   0.950
Average customer waiting time     =   3.100 microsec
Maximum simulation time           = 101.000 microsec
Duration server busy              =  65.000
Server utilization                =  0.6436
Server idle                       =  0.3564
```



The following Matlab code can be used to solve this queue system (Note the code is general – it solves any system provided The Arrivals vector A, and the service vector S)

Example 1: Queueing System – cont'd

```
%
% Problem 9.3 - Leon Garcia's book
clear all
A = [10:10:100];
Errors = [0 1 3 1 0 4 0 1 0 0];
S = zeros(size(A));
D = zeros(size(A));
%
% this loop to computes service times
for i=1:length(A);
    if (Errors(i)==0)  S(i) = 1;
    else
        if (Errors(i)==1) S(i) = 6;
        else
            S(i) = 21;
        end
    end
    %
    % this section computes the departure time for
the ith user
    if (i>1) % this is not the first user
        if (D(i-1) < A(i)) D(i) = A(i) + S(i);
        else
            D(i) = D(i-1) + S(i);
        end
    else
        D(i) = A(i)+S(i);
    end
    %
    % compute waiting time
    W(i) = D(i) - A(i) - S(i);
end
%
```

```
% Compute N(t)
T    = [];  % time axis
T(1) = 0;   % time origin
N    = [];  % number of cutomers
N(1) = 0;   % initial condition
k    = 2;   % place for next insert
A_max = A(length(A)); % last arrival instant
i = 1;      % index for arrivals
j = 1;      % index for departures
t = 0;      % system time

while (t < A_max)
    t = min(A(i), D(j));
    if (t == A(i))
        N(k) = N(k-1) + 1;
        T(k) = t;
        k    = k + 1;
        i    = i + 1;  % get next arrival
    else % departure occurs
        N(k) = N(k-1) - 1;
        T(k) = t;
        k    = k + 1;
        j    = j + 1; % get next departure
    end
end
%
% record remaining departure instants
for i=j:1:length(D)
    t    = D(i);
    N(k) = N(k-1) - 1;
    T(k) = t;
    k    = k + 1;
end

k = k - 1; % decrement k to get real size of N and T
%
% compute means
MeanW              = mean(W);
T_Intervales       = T(2:k)-T(1:k-1);
MeanN              = sum(N(1:k-1).*T_Intervales) / T(k);
IdleDurationsIndex = find(N(1:k-1) ~= 0);
Utilization        = sum(T_Intervales(IdleDurationsIndex))/T(k);
%
```

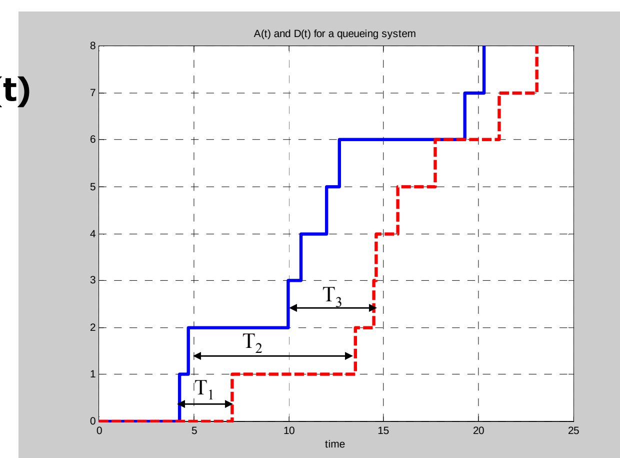
Example 1: Queueing System – cont’d

```
% Display results
fprintf('Block #:  '); fprintf('%3d  ', [1:1:length(A)]); fprintf('\n');
fprintf('Arrivals: '); fprintf('%3d  ', A); fprintf('\n');
fprintf('Errors:   '); fprintf('%3d  ', Errors); fprintf('\n');
fprintf('Service:  '); fprintf('%3d  ', S); fprintf('\n');
fprintf('Departs:  '); fprintf('%3d  ', D); fprintf('\n');
fprintf('Waiting:  '); fprintf('%3d  ', W); fprintf('\n');
fprintf('\n\n');
fprintf('Average no of customers in system = %7.3f\n', MeanN);
fprintf('Average customer waiting time     = %7.3f microsec\n', MeanW);
fprintf('Maximum simulation time           = %7.3f microsec\n', T(k));
fprintf('Duration server busy              = %7.3f microsec\n', ...
         sum(T_Intervales(IdleDurationsIndex)));
fprintf('Server utilization                = %7.4f\n', Utilization);
fprintf('Server idle                       = %7.4f\n',1.0-Utilization);
%
% Plot results
figure(1)
h = stairs(T, N); grid
set(h, 'LineWidth', 3);
xlabel('Time');
ylabel('No of customers in system, N(t)');

figure(2);
[AT, AA] = stairs(A, cumsum(ones(size(A))));
[DT, DD] = stairs(D, cumsum(ones(size(D))));
[NT, NN] = stairs(T, N);
h = plot(AT, AA, '-', DT, DD,'--r', NT, NN,'-.'); grid
set(h, 'LineWidth', 3);
title('Queue sysystem simulation');
ylabel('No of customers');
xlabel('Time');
legend('A(t)', 'D(t)', 'N(t)', 0);

figure(3);
h = stem(W); grid
set(h, 'LineWidth', 3);
ylabel('Waiting time');
xlabel('Customer index');
LegendStr = ['MeanW = ' num2str(MeanW)];
legend(LegendStr, 0);
```


Number of Customers in System

- **Blue curve: A(t)**
- **Red curve: D(t)**

- **Total time spent in the system for all customers = area in between two curves**



Little’s Formula

- **Little’s formula:**

$$E[N] = \lambda E[T]$$

Holds for many service disciplines and for systems with arbitrary number of servers. It holds for many interpretations of the system as well

Example 2:

- **<u>Problem</u>: Let Ns(t) be the number of customers being served at time t, and let τ denote the service time. If we designate the set of servers to be the “system”m then Little’s formula becomes:**

$$E[Ns] = \lambda E[\tau]$$

Where E[Ns] is the average number of busy servers for a system in the steady state.

Example 2: cont'd

Note: for a single server Ns(t) can be either 0 or 1 ➔ E[Ns] represents the portion of time the server is busy. If p_0 = Prob[Ns(t) = 0], then we have

$$1 - p_0 = E[Ns] = \lambda E[\tau], \text{ Or}$$
$$p_0 = 1 - \lambda E[\tau]$$

The quantity $\lambda E[\tau]$ is defined as the utilization for a single server. Usually, it is given the symbol ρ

$$\rho = \lambda E[\tau]$$

For a c-server system, we define the utilization (the fraction of busy servers) to be

$$\rho = \lambda E[\tau] / c$$

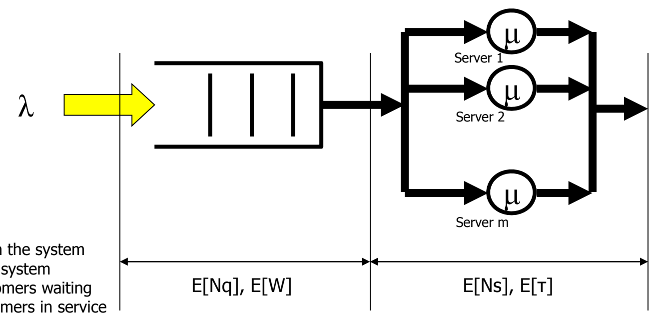
Example 3: Applications on Little's Formula

Refer to the slides for Dr. Waheed.

The slides have 8 good examples!

Queue System and Parameters

- **Queueing system with m servers**
  - **When m = 1 – single server system**
- **Input: arrival statistics (rate λ), service statistics (rate μ), number of customers (m), buffer size**
- **Output: E[N], E[T], E[Nq], E[W], Prob[buffer size = x], Prob[W<w], etc.**



E[N] = mean # of customers in the system
E[T] = mean time spent in the system
E[Nq] = mean number of customers waiting
E[Ns] = mean number of customers in service
E[W] = mean waiting time for a customer
E[τ] = mean service time for a customer

E[N], E[T]
E[N] = E[Nq] + E[Ns],
E[T] = E[W] + E[τ]

The M/M/1 Queue

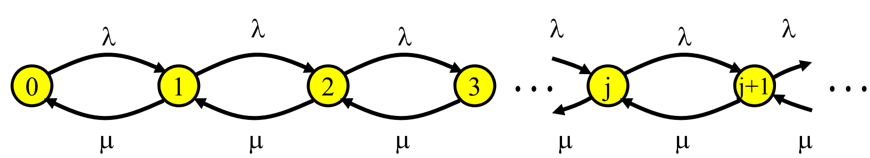
- **Consider m-server system where customers arrive according to a Poisson process of rate λ**
  - **➔ inter-arrival times are iid exponential r.v. with mean 1/λ**
- **Assume the service times are iid exponential r.v. with mean 1/μ**
- **Assume the inter-arrival times and service times are independent**
- **Assume the system can accommodate unlimited number of customers**

The M/M/1 Queue – cont'd

- **What is the steady state pmf of N(t), the number of customers in the system?**
- **What is the PDF of T, the total customer delay in the system?**

The M/M/1 Queue – cont'd

- **Consider the transition rate diagram for M/M/1 system**



- **Note:**
  - **System state – number of customers in systems**
  - **λ is rate of customer arrivals**
  - **μ is rate of customer departure**

The M/M/1 Queue – Distribution of Number of Customers

- **Writing the global balance equations for this Markov chain and solving for Prob[N(t) = j], yields (refer to previous example)**

$$p_j = \text{Prob}[N(t) = j] = (1-\rho)\rho^j$$

for $\rho = \lambda/\mu < 1$

Note that for $\rho = 1$ → arrival rate λ = service rate μ

The M/M/1 Queue – Expected Number of Customers

- **The mean number of customer is given by**

$$E[N] = \sum_j j\ \text{Prob}[N(t) = j]$$

$$= \rho / (1-\rho)$$

The M/M/1 Queue – Mean Customer Delay

- **The mean total customer delay in the system is found using Little's formula**

$$\begin{aligned} E[T] &= E[N]/\lambda \\ &= \rho/[\lambda(1-\rho)] \\ &= 1/\mu(1-\rho) \\ &= 1/(\mu-\lambda) \end{aligned}$$

The M/M/1 Queue – Mean Queueing Time

- **The mean waiting time in queue is given by**

$$\begin{aligned} E[W] &= E[T] - E[\tau] \\ &= \rho/(1-\rho)\quad E[\tau] \end{aligned}$$

The M/M/1 Queue – Mean Number in Queue

- **Again we employ Little's formula:**

$$E[Nq] = \lambda E[W]$$

$$= \rho^2 / (1-\rho)$$

Remember:

server utilization $\rho = \lambda/\mu = 1-p_0$

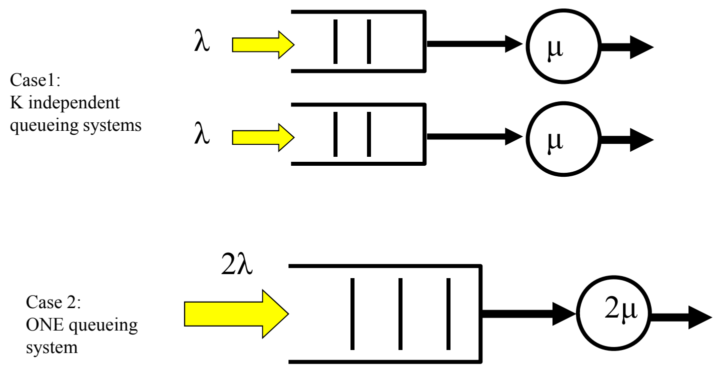
All previous quantities E[N], E[T], E[W], and E[Nq] $\rightarrow \infty$ as $\rho \rightarrow 1$

Scaling Effect for M/M/1 Queues

- **Consider a queue of arrival rate λ whose service rate is μ**
  - $\rho = \lambda/\mu$,
  - **The expected delay E[T] is given by**
    $E[T] = (1/\mu) / (1-\rho)$
- **If the arrival rate increases by a factor of K, then we either**
  1. **Have K queueing systems, each with a server of rate μ**
  2. **Have one queueing system with a server of rate $K\mu$**
- **Which of the two options will perform better?**

Example 4: Scaling Effect for M/M/1 Queues

- **Example: K = 2: M/M/1 and M/M/2 systems with the same arrival rate and the same maximum processing rate**



Example 4: Scaling Effect for M/M/1 Queues – cont'd

- **Case 1: K queueing systems**
  - **Identical systems**
  - **E[T] is the same for all – E[T] = $(1/\mu) / (1-\rho)$**

- **Case 2: 1 queueing system with server of rate $K\mu$**
  - **ρ for this system = $(K\lambda)/(K\mu) = \lambda/\mu$ – same as the original system**
  - **E[T'] = $(1/(K\mu)) / (1-\rho) = (1/K)$ E[T]**

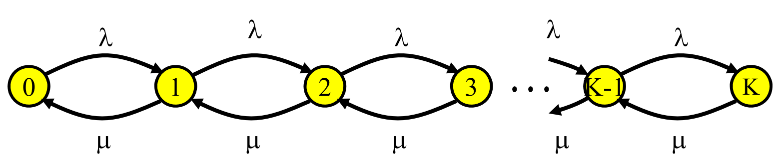
- **Therefore, the second option will provide a less total delay figure – significant delay performance improvement!**

M/M/1/K – Finite Capacity Queue

- **Consider an M/M/1 with finite capacity K < ∞**
- **For this queue – there can be at most K customers in the system**
  - **1 being served**
  - **K-1 waiting**
- **A customer arriving while the system has K customers is BLOCKED (does not wait)!**

M/M/1/K – Finite Capacity Queue – cont'd

- **Transition rate diagram for this queueing system is given by:**
  - **N(t) - A continuous-time Markov chain which takes on the values from the set {0, 1, …, K}**



M/M/1/K – Finite Capacity Queue – cont'd

- **The global balance equations:**

$$\lambda \quad p_0 = \mu p_1$$
$$(\lambda + \mu)p_j = \lambda p_{j-1} + \mu p_{j+1} \qquad \text{for } j=1, 2, \ldots, K-1$$
$$\mu \quad p_K = \lambda p_{K-1}$$

➔ Prob[N(t) = j] = p_j = $(1-\rho)\rho^j/(1-\rho^{K+1})$ j=0,1, …, K; ρ<1

When ρ = 1, p_j = 1/(K+1) (all states are equiprobable)

M/M/1/K – Mean Number of Customers

- **Mean number of customers, E[N] is given by:**

$$E[N] = \sum_{j=0}^{K} j \Pr[N(t) = j]$$

$$= \begin{cases} \dfrac{\rho}{1-\rho} - \dfrac{(K+1)\rho^{K+1}}{1-\rho^{K+1}} & \rho < 1 \\ K/2 & \rho = 1 \end{cases}$$

M/M/1/K – Blocking Rate

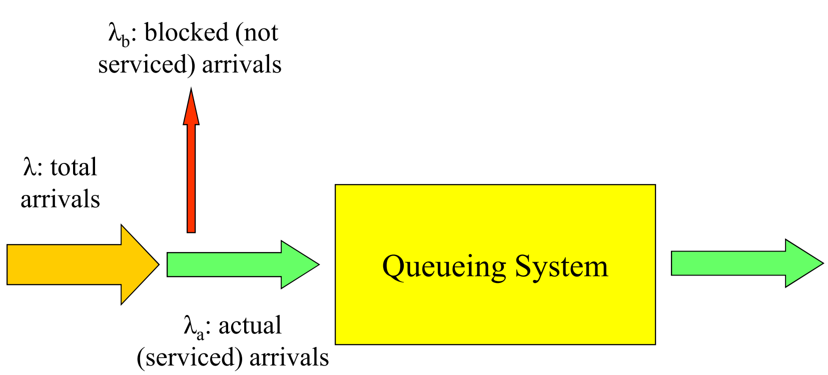
- **A customer arriving while the system is in state K is BLOCKED (does not wait)!**
- **Therefore, rate of blocking, λ_b is given by**

$$\lambda_b = \lambda\, p_K$$

- **The actual arrival rate into the system is λ_a given**

$$\begin{aligned}\lambda_a &= \lambda - \lambda_b \\ &= \lambda(1 - p_K)\end{aligned}$$

M/M/1/K – Blocking Rate – cont'd



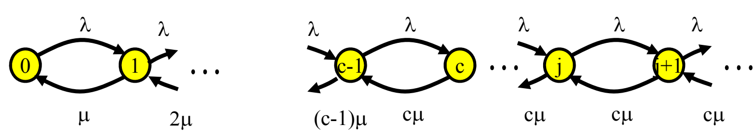
M/M/1/K – Mean Delay

- **The mean total delay E[T] is given by**

$$E[T] = E[N] / \lambda_a$$

Multi-Server Systems: M/M/c

- **The transition rate diagram for a multi-server M/M/c queue is as follows:**
  - **Departure rate = $k\mu$ when k servers are busy**
  - **We can show that the service time for a customer finding k servers busy is exponentially distributed with mean $1/(k\mu)$**



Multi-Server Systems: M/M/c – cont'd

- **Writing the global balance equations:**

$$\lambda p_0 = \mu p_1$$
$$j\mu p_j = \lambda p_{j-1} \quad \text{for } j=1, 2, \ldots, c$$
$$c\mu p_j = \lambda p_{j-1} \quad \text{for } j= c, c+1, \ldots$$

Note this distribution is the same as that for M/M/1 when you set c to 1.

➔

$p_j= a^j/j!\ p_0$ (for $j=1, 2, \ldots, c$) and
$p_j= \rho^{j-c}/c!\ a^c\ p_0$ (for $j=c, c+1, \ldots$)

where $a = \lambda/\mu$ **and** $\rho = a/c$

- **From this we note that the probability of system being in state c, pc, is given by**

$$p_c = a^c/c!\ p_0$$

Multi-Server Systems: M/M/c – cont'd

- **To find** p_0**, we resort to the fact that** $\Sigma\ p_j = 1$

➔
$$p_0 = \left\{ \sum_{j=0}^{c-1} \frac{a^j}{j!} + \frac{a^c}{c!} \frac{1}{1-\rho} \right\}^{-1}$$

- **The probability that an arriving customer has to wait**

$$\text{Prob}[W > 0] = \text{Prob}[N \geq c]$$
$$= p_c + p_{c+1} + p_{c+2} + \ldots$$
$$= p_c/(1-\rho)$$

Erlang-C formula

Question: What is Prob[W>0] for M/M/1 system?

Multi-Server Systems: M/M/c – cont'd

- **The mean number of customers in queue (waiting):**

$$E[N_q] = \sum_{j=c}^{\infty} (j-c) \Pr[N(t) = j]$$
$$= \sum_{j=c}^{\infty} (j-c) \rho^{j-c} p_c$$
$$= \frac{\rho}{(1-\rho)^2} p_c$$
$$= \frac{\rho}{1-\rho} \Pr[W > 0]$$

Multi-Server Systems: M/M/c – cont'd

- **The mean waiting time in queue:**

$$E[W] = E[N_q]/\lambda$$

- **The mean total delay in system:**

$$E[T] = E[W] + E[\tau]$$
$$= E[W] + 1/\mu$$

- **The mean number of customers in system:**

$$E[N] = \lambda E[T]$$
$$= E[N_q] + a$$

Why?

Example 5:

- **A company has a system with four private telephone lines connecting two of its sites. Suppose that requests for these lines arrive according to a Poisson process at rate of one call every 2 minutes, and suppose that call durations are exponentially distributed with mean 4 minutes. When all lines are busy, the system delays (i.e. queues) call requests until a line becomes available.**
- **Find the probability of having to wait for a line.**
- **What is the average waiting time for an incoming call?**

Example 5: cont'd

- **Solution:**

$\lambda = ½$, $1/\mu = 4$, $c = 4 \Rightarrow a = \lambda/\mu = 2$

$\Rightarrow \rho = a/c = ½$

$p_0 = \{1+2+2^2/2!+2^3/3!+2^4/4!\ (1/(1-\rho))\}^{-1}$
$= 3/23$

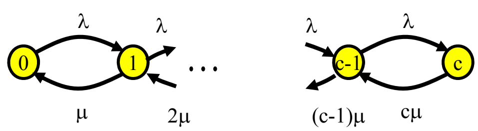
$p_c = a^c/c!\ p0$
$= 2^4/4! \times 3/23$

(1) $Prob[W > 0] = p_c/(1-\rho)$
$= 2^4/4! \times 3/23 \times 1/(1-1/2)$
$= 4/23$
≈ 0.17

(2) To find E[W], find E[Nq] …
$E[Nq] = \rho/(1-\rho) * Prob[W>0] = 0.1739$
$E[W] = E[Nq]/\lambda = 0.35$ min

Multi-Server Systems: M/M/c/c

- **The transition rate diagram for a multi-server with no waiting room (M/M/c/c) queue is as follows:**
    - **Departure rate = kμ when k servers are busy**



PMF for Number of Customers for M/M/c/c

- **Writing the global balance equations, one can show:**

$$p_j = a^j/j!\ p_0 \qquad \text{(for } j=0, 1, \ldots, c)$$

where a = λ/μ (the offered load)

- **To find p_0, we resort to the fact that $\Sigma\, p_j = 1$**

$$p_0 = \left\{ \sum_{j=0}^{c} \frac{a^j}{j!} \right\}^{-1}$$

Erlang-B Formula

- **Erlang-B formula is defined as the probability that all servers are busy:**

$$\Pr[N=c]=p_c$$
$$=\frac{a_c/j!}{1+a+a^2/2!+\ldots+a^c/c!}$$

Expected Number of customers in M/M/c/c

- **The actual arrival rate *into* the system:**

$$\lambda_a=\lambda(1-p_c)$$

- **Average total delay figure:**

$$E[T]=E[\tau]$$

Why?

- **Average number of customers:**

$$E[N]=\lambda_a E[\tau]$$

Example 6:

- **A company has a system with four private telephone lines connecting two of its sites. Suppose that requests for these lines arrive according to a Poisson process at rate of one call every 2 minutes, and suppose that call durations are exponentially distributed with mean 4 minutes. When all lines are busy, the system BLOCKS the incoming call and generates a busy signal.**
- **Find the probability of being blocked.**

Example 6:

- **Solution:**

$\lambda = 1/2,\ 1/\mu = 4,\ c = 4 \Rightarrow a = \lambda/\mu = 2$

$\Rightarrow \rho = a/c = 1/2$

$$p_c = \frac{a^c/c!}{1 + a + a^2/2! + a^3/3! + a^4/4!}$$

$$= \frac{2^4/4!}{1 + 2 + 2^2/2! + 2^3/3! + 2^4/4!} = 9.5\%$$

Therefore, the probability of being blocked is 0.095.

M/G/1 Queues

- **Poisson arrival process (i.e. exponential r.v. interarrival times)**
- **Service time: general distribution $f_\tau(x)$**
  - **For M/M/1, $f_\tau(x) = \mu e^{-\mu x}$ for x > 0**

- **The state of the M/G/1 system at time t is specified by**
  1. **N(t)**
  2. **The remaining (residual) service time of the customer being served**

The Residual Service Time

- **Mean residual time (see example and derivation in handout) is given by**

$$E[R] = \frac{E[\tau^2]}{2E[\tau]}$$

Mean Waiting Time in M/G/1

- **The waiting time of a customer is the sum of the residual service time R′ of the customer (if any) found in service and the Nq(t)= k-1 service time of the customers (if any) found in queue**

$$
\begin{aligned}
E[W] &= E[R'] + E[Nq]\, E[\tau] \\
&= E[R'] + \lambda E[W] E[\tau] \\
&= E[R'] + \rho\, E[W]
\end{aligned}
$$

Mean Waiting Time in M/G/1 – cont'd

- **But residual service time R′ (as observed by an arriving customers) is either**
  - **0 is the server is free**
  - **R if the server is busy**
- **Therefore, mean of R′ is given by**

$$
\begin{aligned}
E[R'] &= 0 \text{ X } Pro[N(t)=0] + E[R](1-Pro[N(t)=0]) \\
&= E[\tau^2]/(2E[\tau]) \text{ X } \rho \\
&= \lambda E[\tau^2]/2
\end{aligned}
$$

Mean Waiting Time in M/G/1 – cont'd

- **Substituting back, yields**

$$E[W] = \frac{\lambda E[\tau^2]}{2(1-\rho)}$$

$$= \frac{\lambda(\delta^2_\tau + E[\tau]^2)}{2(1-\rho)}$$

$$= \frac{\rho\,(1 + C_\tau^2)}{2(1-\rho)} E[\tau]$$

Remember:
- $E[\tau^2] = \delta^2_\tau + E[\tau]^2$
- $C^2_\tau = \delta^2_\tau / E[\tau]^2$

Pollaczek-Khinchin (P-K) Mean Value Formula

Mean Delay in M/G/1 – cont'd

- **The mean waiting time, E[T] is found by adding mean service time to E[W]:**

$$E[T] = E[\tau] + E[W]$$

$$= E[\tau] + \frac{\rho\,(1 + C_\tau^2)}{2(1-\rho)} E[\tau]$$

Example 7:

- **<u>Problem</u>: Compare E[W] for M/M/1 and M/D/1 systems.**

- **<u>Answer</u>:**

M/M/1: service time, τ, is exponential r.v. with parameter μ

➔ $E[\tau] = 1/\mu$, $E[\tau^2] = 2/\mu^2$, $\delta^2_\tau = 1/\mu^2$, $C^2_\tau = 1$

M/D/1: service time, τ, is constant with value $\tau = 1/\mu$

➔ $E[t] = 1/\mu$, $E[\tau^2] = 1/\mu^2$, $\delta^2_\tau = 0$, $C^2_\tau = 0$

Example 7: cont'd

- **<u>Answer</u>: cont'd**

Substitute in P-K mean value formula

M/M/1:

$$E[W_{M/M/1}] = \frac{\lambda E[\tau^2]}{2(1-\rho)} = \frac{\rho}{(1-\rho)} E[\tau]$$

M/D/1:

$$E[W_{M/D/1}] = \frac{\lambda E[\tau^2]}{2(1-\rho)} = \frac{\rho}{2(1-\rho)} E[\tau]$$

$$= \frac{1}{2} E[W_{M/M/1}]$$

The waiting time in an M/D/1 queue is half of that of an M/M/1 system

Example 8:

- **Problem**: Assume traffic is arriving at the input port of a router according to a Poisson arrival process of rate λ = 100 packets/sec. If the traffic distribution is as follows:
  - 30% of packets are 512 Bytes long,
  - 50% of packets are 1024 Bytes long,
  - 20% of packets are 4096 Bytes long

  If the transmit speed of the router output port is 1.5 Mb/s

  a) What is the average packet transmit time?

  b) What is the average packet waiting time before transmit?

  c) What is the average buffer size in the router?

Example 8: cont'd

- **Solution**:

a) Average packet size,

$E[L] = 0.3x512 + 0.5x1024 + 0.2x4096$

$= 1484.8$ Bytes

average transmit time = $E[L]/R = 1484.8x8/1.5x10^6 =$ 0.0079 sec

b) $E[L^2] = 0.3x(512x8)^2 + 0.5x(1024x8)^2 + 0.2x(4096x8)^2 =$ 2.5334e+008 Bits^2

$E[\tau^2] = E[L^2]/R^2 =$ 1.1259e-004 $\sec^2$

$\rho = \lambda E[\tau] = 0.7919$

$E[W] = 0.5\, \lambda\, E[\tau^2] / (1-\rho)$

$= 0.0271$ sec

c) $E[Nq] = \lambda E[W]$

$= 2.705$ packet