ECSE-4730: Computer Communication Networks (CCN)

Network Layer Performance Modeling & Analysis

Shivkumar Kalyanaraman: shivkuma@ecse.rpi.edu

Biplab Sikdar: sikdab@rpi.edu

http://www.ecse.rpi.edu/Homepages/shivkuma

Rensselaer

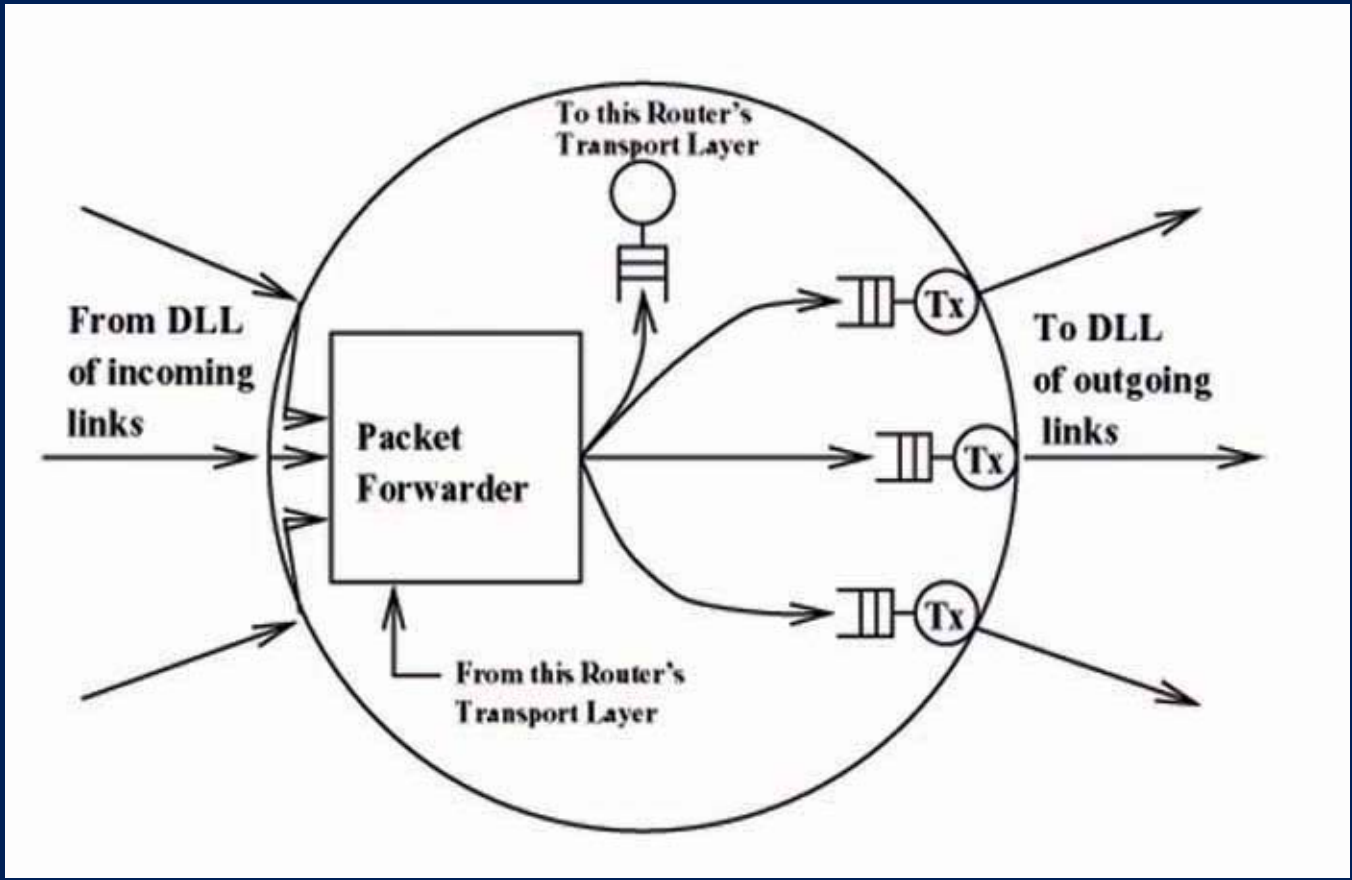
RSVP

- ***Network Layer Performance Modeling & Analysis***
  - **Part I: Essentials of Probability**
  - ***Part II: Inside a Router***
  - **Part III: Network Analysis**

Network Layer Performance Modeling & Analysis: Part II Inside a Router

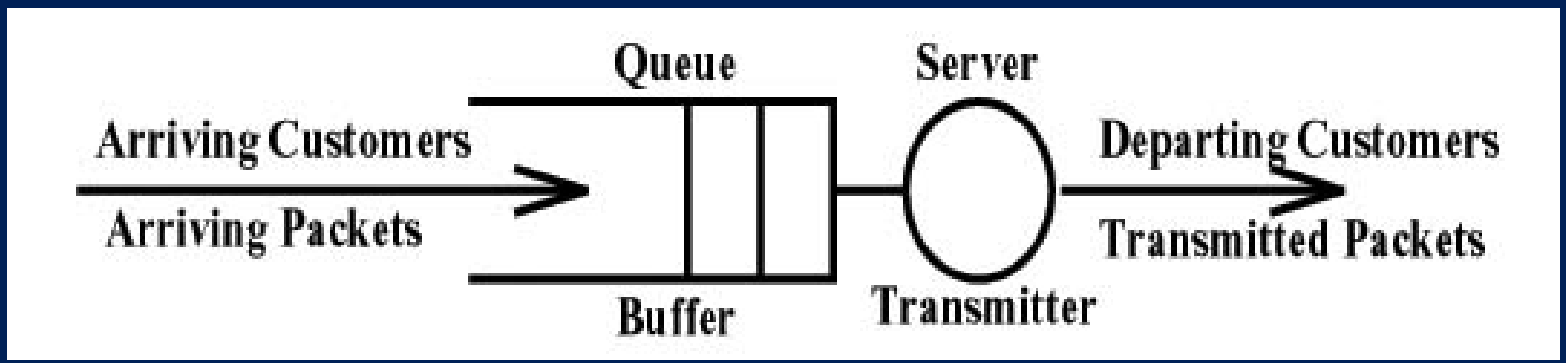
- Basic Single Queue Model
- Poisson Arrival Model
- The M/M/1 Queue
- Read any of the queuing theory references, e.g. Schwartz (Sections 2.1-3), Molloy, Kleinrock.

Queuing in the Network Layer at a Router



Basic Single Queue Model

- Classical queuing theory can be applied to an output link in a router.



Basic Single Queue Model

- For example, a *56 kbps* transmission line can "serve" 1000-bit packets at a rate of

$$\frac{56{,}000 \text{ bits/sec}}{1000 \text{ bits/packet}} = 56 \text{ packets/sec}$$

Applications of Queuing Analysis Outside of Networking

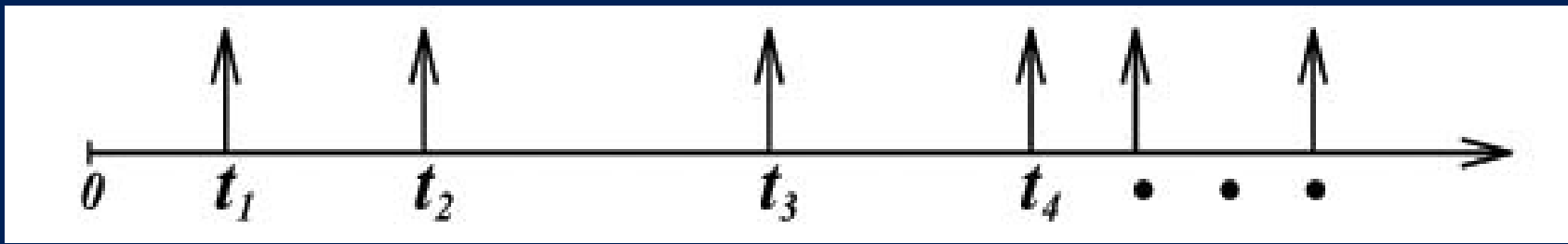
- Checkout line in a supermarket
- Waiting for a teller in a bank
- Batch jobs waiting to be processed by the CPU

Applications of Queuing Analysis Outside of Networking

- "That's the way the whole thing started,
  Silly but it's true,
  Thinking of a sweet romance
  Beginning in a queue."
  -G. Gouldman, "Bus Stop"
  The Hollies

The Poisson Arrival Model

- A Poisson process is a sequence of events "randomly spaced in time"



The Poisson Arrival Model

- Examples
  - Customers arriving to a bank
  - Packets arriving to a buffer
- The rate λ of a Poisson process is the average number of events per unit time (over a long time).

Properties of a Poisson Process

- For a length of time t the probability of n arrivals in t units of time is

τ_4

$$P_n(t) = \frac{(\lambda t)^n}{n!} e^{-\lambda t}$$

Properties of a Poisson Process

- **For 2 disjoint (non-overlapping) intervals, (s_1, s_2) and (s_3, s_4), (i.e. $s_1 < s_2 \leq s_3 < s_4$), the number of arrivals in (s_1, s_2) is independent of the number of arrivals in (s_3, s_4)**

Interarrival Times of a Poisson Process

- Pick an arbitrary starting point in time (call it 0).
- Let τ_1 = the time until the next arrival

$$P(\tau_1 > t) = P_0(t) = e^{-\lambda t}$$

Interarrival Times of a Poisson Process

- So

$$F_{\tau_1}(t) = P(\tau_1 \le t) = 1 - e^{-\lambda t} \quad \textbf{and} \quad f_{\tau_1}(t) = \lambda e^{-\lambda t}$$

τ_1 ,the time until the first arrival, Has an exponential distribution!

Interarrival Times of a Poisson Process

- Let τ_2 = the length of time <u>between</u> the first and second arrival.
- We can show that

$$P(\tau_2 > t \mid \tau_1 = s) = P(\tau_2 > t) = e^{-\lambda t} \quad \text{for any } s,t > 0$$

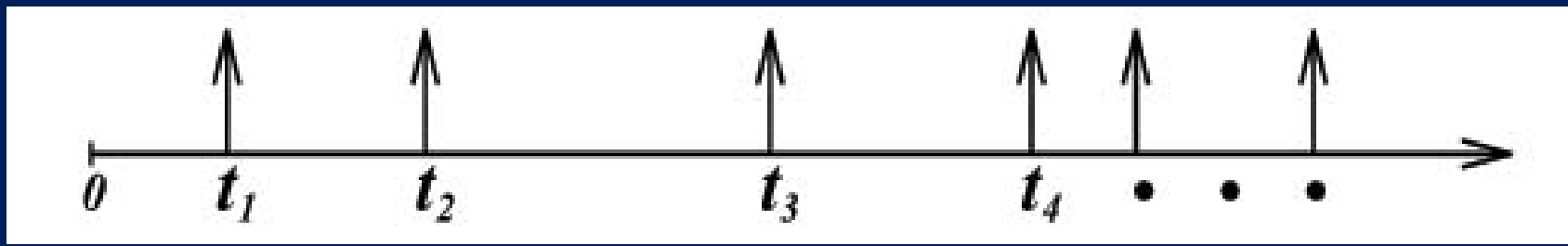
i.e. τ_2 is exponential and independent of τ_1 !

Interarrival Times of a Poisson Process

- Similarly define τ_3 as the time between the second and third arrival; τ_4 as the time between the third and fourth arrival;…
- The random variables $\tau_1, \tau_2, \tau_3,$ … are called the <u>interarrival times</u> of the Poisson process

Interarrival Times of a Poisson Process

- The interarrival time random variables, $\tau_1, \tau_2, \tau_3 \ldots$
  - Are (pair-wise) independent.
  - Each has an exponential distribution with mean $1/\lambda$.



The M/M/1 Queue

- An M/M/1 queue has
  - Poisson arrivals (with rate λ)
  - Exponential service times (with mean 1/μ, so μ is the "service rate").
  - One (1) server
  - An infinite length buffer
- The M/M/1 queue is the most basic and important queuing model.

Queuing Notation

"M/M/1" is a special case of more general (Kendall) notation: X/Y/m/k, where

- **X is a symbol representing the interarrival process**
  - **M = Poisson (exponential interarrival times, τ)**
  - **D = Deterministic (constant τ).**

Queuing Notation

- Y is a symbol representing the service distribution
  - M = exponential, D = deterministic
  - G = General (or arbitrary).
- m = number of servers
- k = number of buffer slots (omitted when k = ∞)

Aside: The D/D/1 Queue

- The D/D/1 queue has
  - Deterministic arrivals (periodic with period = $1/\lambda$).
  - Deterministic service times (each service takes exactly $1/\mu$).
  - As well as 1 server and an infinite length buffer.

Aside: The D/D/1 Queue

- If $\lambda < \mu$ then there is no waiting in a D/D/1 queue.

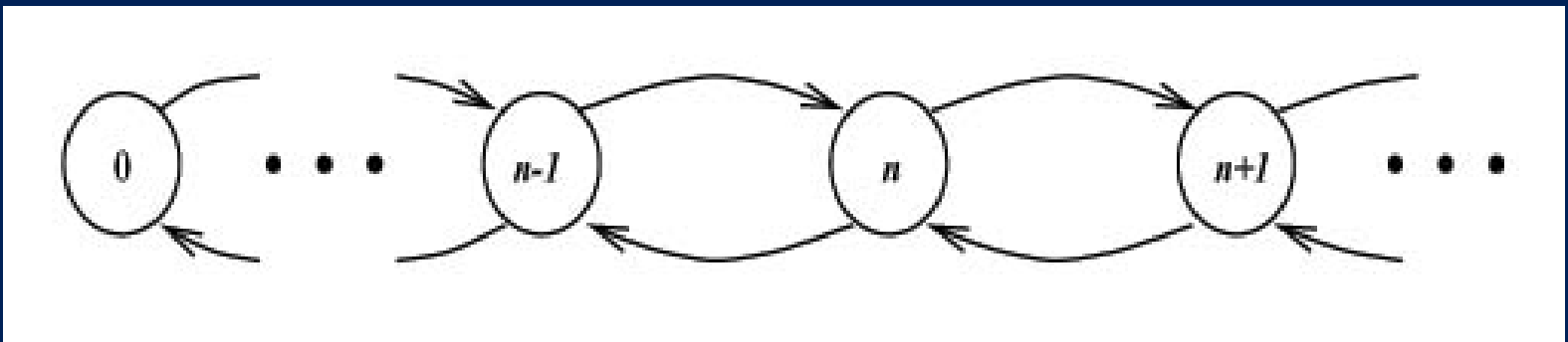
Randomness is a major cause of delay in a network node!

State Analysis of an M/M/1 Queue

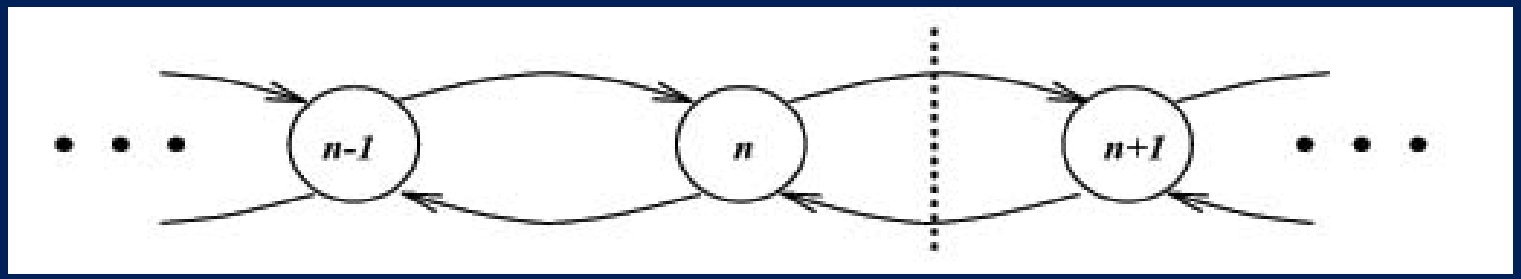
- Let n be the state of the system = the number of packets in the system (including the server).
- Let p_n be the steady state probability of finding n customers waiting in the system (including the server).

State Analysis of an M/M/1 Queue

- How to find p_n? The state diagram:



State Analysis of an M/M/1 Queue



- If the system is stable (i.e. $\neq p_n$ 0 for each n), then in a steady state it will drift back and forth across the dotted line. So,
- the number of transitions from left to right = the number of transitions from right to left.

State Analysis of an M/M/1 Queue

- Thus we obtain the balance equations

$$p_n \lambda = p_{n+1} \mu \quad \text{for each} \quad n \geq 0$$

State Analysis of an M/M/1 Queue

- Lets solve the balance equations: $p_n \lambda = p_{n+1} \mu$
- For $n = 0$ we get

- If we let $\rho = \lambda / \mu$, this becomes

$$p_1 = \rho p_0$$

State Analysis of an M/M/1 Queue

- **Similarly**

$$p_2 = \rho\, p_1 = \rho^2 p_0$$

- **And if general**

$$p_n =$$

State Analysis of an M/M/1 Queue

- We have $p_n = p^n p_0$ for n =1,2,3,...
- We need to solve for p_0 , so we need one more equation. Use

$$\sum_{n=0}^{\infty} p_n = 1$$

- We obtain

$$1 = \sum_{n=0}^{\infty} \rho^n p_0 = p_0 \sum_{n=0}^{\infty} \rho^n = \begin{cases} p_0 \left(\dfrac{1}{1-\rho} \right) & \text{for } \rho < 1 \\ \infty & \text{for } \rho \geq 1 \end{cases}$$

State Analysis of an M/M/1 Queue

- **So we must have** $p_0 = 1 - \rho$

and

$$p_n = (1-\rho)\rho^n \quad \text{for } n = 1, 2, 3, \ldots$$

State Analysis of an M/M/1 Queue

- Note that requiring $\rho < 1$ for stability (i.e. $\lambda < \mu$) makes intuitive sense.
- Also $\rho = 1 - \rho_0$

  = probability that the queuing system is NOT empty

  = probability the server is working

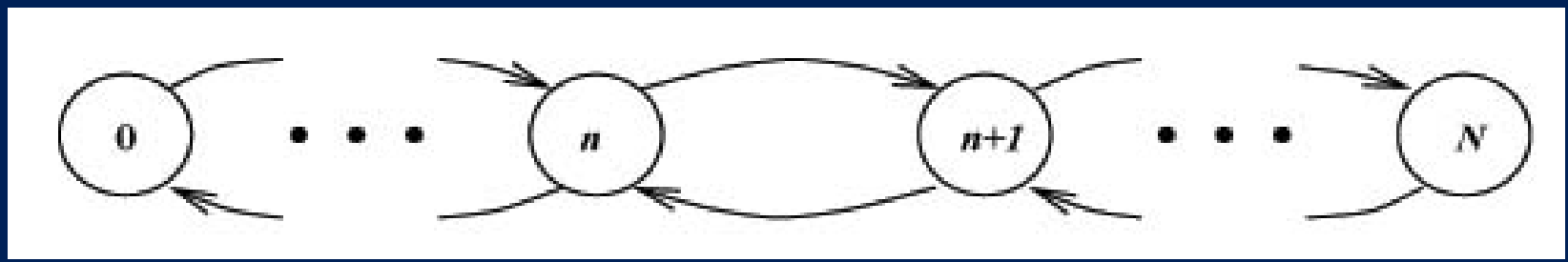
State Analysis of an M/M/1 Queue

So ρ is sometimes called the "server utilization"

- **Finally note that $p_n = (1-\rho)p^n$, $n = 0,1,2,3,\ldots$ is a geometric distribution**

The Finite Buffer Case: M/M/1/N

- Infinite buffer assumption is unrealistic in practice.
- N = total number of buffer slots (including server).
- New state diagram:



The Finite Buffer Case: M/M/1/N

- Get the same balance equations, $p_n\lambda = p_{n+1}\mu$ but now only for n = 0,1,2,…,N - 1 with $N < \infty$. So

$$p_n = \rho\, p_{n-1} = \rho^n p_0 \quad \text{for } n = 0,1,2,...,N$$

as before, but we get a different p_0.

The Finite Buffer Case: M/M/1/N

- From $p_n = \rho^n p_0$ for $n = 0,1,2,\ldots, N < \infty$ and $\sum_{n=0}^{N} p_n = 1$ we get

$$p_0 = 1 - \sum_{n=1}^{N} \rho^n p_0$$

- So

$$p_0 = \frac{1}{1+\sum_{n=1}^{N} \rho^n} = \frac{1}{1+\frac{\rho(1-\rho^N)}{(1-\rho)}} = \ldots = \frac{1-\rho}{1-\rho^{N+1}}$$

The Finite Buffer Case: M/M/1/N

- Note that this holds <u>for any</u> $\rho \geq 0$. No need to assume $\rho < 1$. We always have the stability in the finite buffer case.

Blocking Probability and the Right Size Buffer

- So in the finite buffer case,

$$p_n = \frac{(1-\rho)\rho^n}{1-\rho^{N+1}} \quad \text{for} \quad n = 0,1,2,\ldots,N$$

- Note that P_N is the probability that the buffer is full at an arbitrary point in time.

Blocking Probability and the Right Size Buffer

- Since arrivals are independent of buffer state, we have $P_N = P_B$ = probability an arriving packet is turned away due to a full buffer.
- P_B is called <u>blocking probability.</u>

Blocking Probability and Buffer Size

- P_B is very important!
- We can use P_B to choose the correct buffer size.
- Example: For $\rho = 0.5$, $p_N > 10^{-6}$ for $N \leq 18$, while $p_N < 10^{-6}$ for $N \geq 19$.

Blocking Probability and Buffer Size

- Thus, if we desire a blocking probability less than 10^{-6}, we need a buffer capable of holding 19 or more packets.

Throughput in the Finite Buffer Case

- The <u>throughput</u> γ of any queuing system is the rate at which customers successfully leave the system.
- For the M/M/1 infinite buffer case, $\gamma = \lambda$ if the system is stable. (Everything that arrives must eventually depart.)

Throughput in the Finite Buffer Case

- For the M/M/1/N finite buffer case, $\gamma = \lambda(1 - P_B)$

  (Everything that arrives and is not blocked must eventually depart.)

Throughput in the Finite Buffer Case

Alternate way to compute throughput of M/M/1/N:

Look at the output side.

- P (server is busy) = $1 - p_0$
- When the server is busy, the output rate = μ
- when the sever is idle, the output rate = 0
- So the average rate = $\gamma = \mu(1 - p_0) + 0 p_0$

Aside: Derivation of $P_N = P_B$ Using Throughput

- Equating our two formulas for γ we get

$$\mu(1-p_0) = \lambda(1-P_B)$$

- Solving for P_B we get

$$P_B = 1 - \frac{1-p_0}{\rho} = \ldots = \frac{(1-\rho)\rho^N}{1-\rho^{N+1}} = p_N$$

- Isn't that neat?

Approximation of a Finite Buffer System by the Infinite Buffer Model

- For a infinite buffer, $p_n = (1-\rho)\rho^n$
- For a finite buffer, $p_n = (1-\rho)\rho^n / (1-\rho^{N+1})$
- For $\rho = 0.8$ and $N = 16$ packets, these probabilities differ by less than 2.3%
- For $\rho = 0.8$ and $N = 32$, the difference is only 0.06%

Approximation of a Finite Buffer System by the Infinite Buffer Model

The infinite buffer model is a very good approximation of a finite buffer system.

Even for moderate buffer sizes!

How Long is that Line?

- Lets look again at the M/M/1 queuing system.
- *n* = the number in the system (including the server)
- So the average number in the system is

$$E(n) = \sum_{n=0}^{\infty} np_n = (1-\rho)\sum_{n=0}^{\infty} np^n = (1-\rho)\frac{\rho}{(1-\rho)^2} = \frac{\rho}{1-\rho}$$

Little's Formula and Queuing Delay

- Let T = time spent by a customer in a queuing system (waiting and being served).
- $E(T)$ = the average delay for a customer.

Little's Formula and Queuing Delay

- Little's Formula says $\lambda E(T) = E(n)$

- where λ is the "arrival rate for customers eventually served" (which we called γ)

Little's Formula holds for very general queuing systems (not just M/M/1). Even whole networks!

Little's Formula and Queuing Delay

- Little's Formula is either deep of obvious. Intuition:
- Pick a "typical customer"
- When it arrives to the queuing system, it should find *E(n)* customers waiting.

Little's Formula and Queuing Delay

- When it leaves the system, it has been in the system for $E(T)$. Thus $\lambda E(T)$ customers should have arrived during its time in the system.
- In steady state, the average number of customers left behind on the departure should equal the average number found on the arrival, i.e. $\lambda E(T) = E(n)$

Little's Formula and Queuing Delay

- Let's apply Little to the M/M/1 queue

$$E(T) = \frac{E(n)}{\lambda} = \frac{\rho}{\lambda(1-\rho)} = \frac{1}{\mu - \lambda}$$

- *E(T)* is measured in units of time. Sometimes it is more convenient to consider

$$\mu E(T) = \frac{\mu E(n)}{\lambda} = \frac{\rho}{\rho(1-\rho)} = \frac{1}{1-\rho}$$

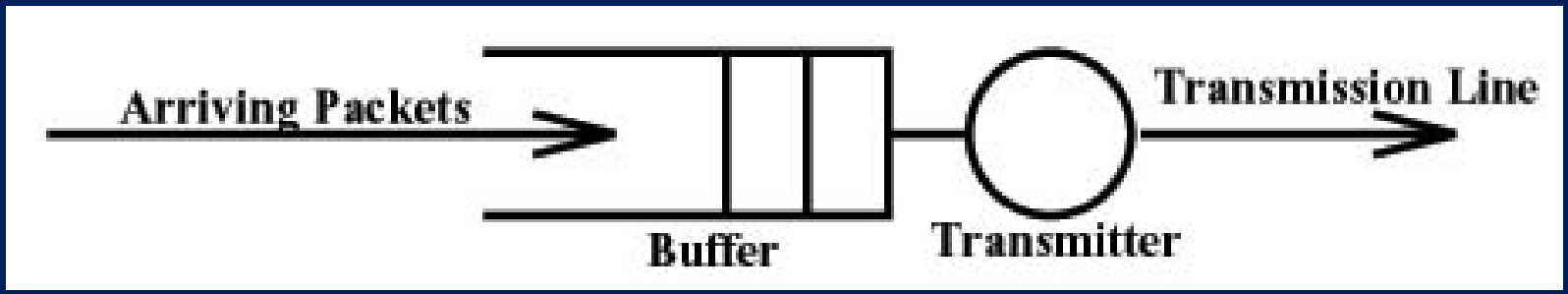
which is unitless

Little's Formula and Queuing Delay

- Sometimes we consider the <u>waiting time</u> W, i.e. the time spent waiting in the queue (not in service). So,

$$E(W) = E(T) - \frac{1}{\mu}$$

Single Link Example



- **Poisson packet arrivals with rate λ = 2000 p/s**
- **Fixed link capacity C = 1.544 Mb / s (T1 Carrier rate).**

Single Link Example

- We approximate the packet length distribution by an exponential with mean L = 515 b/p
- Thus the service time is exponential with mean

$$\frac{1}{\mu} = \frac{L}{C} = \frac{515\ b/p}{1.544\ Mb/s} \approx 0.33 ms/p$$

i.e. packets are served at a rate of

$$\mu = 3000\ p/s$$

Single Link Example

- Using our formulas for an M/M/1 queue

$$\rho = \frac{\lambda}{\mu} = 0.67$$

So

$$E(n) = \frac{\rho}{1-\rho} = 2.0 \text{ packets}$$

and

$$E(T) = \frac{E(n)}{\lambda} = 1.0 \text{ ms}$$

Other Queuing Models

- There are many other important queuing models which are useful in networking.
- M/M/k for k>1. Multiple servers
  - Good model of a link which is made up of multiple channels, either physically of through multiplexing (e.g. a T1 carrier is typically time division multiplexed with k = 24).
  - Has worse performance at lower loads than M/M/1 with same total capacity.

Other Queuing Models

- M/M/k/k for k $\geq$ 1. One or more servers, no buffers (except one in each server).
  - Important model in circuit switched networks.
  - Models a trunk line with k circuits available.

Other Queuing Models

- Any customer (a call) which doesn't get a circuit is blocked (gets a busy signal).
- Blocking probability is given by the Erlang B (or Erlang Loss) Formula

$$P_B = \frac{\rho^k / k!}{\sum_{i=0}^{k} \rho^i / i!} \qquad \rho \geq 0$$

Other Queuing Models

- M/G/1. Arbitrary service (packet length) distribution.
  - Can still compute the mean number in the system via the Pollaczek-Khinchine (P-K) Formula

$$E(n) = \left(\frac{\rho}{1-\rho}\right)\left[1 - \frac{\rho}{2}(1-\mu^2\sigma^2)\right] \quad \rho < 1$$

Other Queuing Models

$$E(n) = \left(\frac{\rho}{1-\rho}\right)\left[1 - \frac{\rho}{2}(1-\mu^2\sigma^2)\right] \quad \rho<1$$

where σ^2 is the variance of the service time distribution. Again, variablility (randomness) causes delay.

- Can apply Little's Formula to get the mean delay

Other Queuing Models

- **M/D/1. Deterministic service times (packet length).**
  - **Special case of M/G/1 with $\sigma^2 = 0$**

$$E(n) = \left(\frac{\rho}{1-\rho}\right)\left(1-\frac{\rho}{2}\right) \qquad \rho < 1$$

  - **Under heavy load ($\rho \approx 1$), M/D/1 has half the delay of an M/M/1**
  - **This is one motivation for a fixed-packet-length system like ATM**

Other Queuing Models

- Can also model and analyze other queuing systems
  - With priority
  - With more general arrival process
  - With "vacations"
  - Many others

Other Queuing Models

- See Schwartz (Ch. 2), Kleinrock (Vol. I & II) or take ECSE-6820/DSES-6820, Queuing (sic) Systems & Applications
- Queuing theory is also used in analysis of Operating Systems, e.g. in CSCI-6140