Threeparts/

Part I/: Essentials of Probability.

Part II/: Inside a Router.

Part III/: Network Analysis.

Read any of the queuing theory references, e.g.

- The M/M/1 Queue
- Poisson Arrival Model
- Basic Single Queue Model

INSIDE A ROUTER

MODELING AND ANALYSIS, PART II:

NETWORK LAYER PERFORMANCE

Part III: Network Analysis.

Part II: Inside a Router.


Three parts.

MODELING AND ANALYSIS

NETWORK LAYER PERFORMANCE
Queueing in the Network Layer at a Router

From DLL of incoming links

To DLL of outgoing links

Packet Forwarder

To this Router's Transport Layer

From this Router's Transport Layer

Basic Single Queue Model

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

- 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 Queueing 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.
- “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.”

\[
0 \quad t_1 \quad t_2 \quad t_3 \quad t_4 \quad \cdots
\]

- Examples
  - Customers arriving to a bank.
  - Packets arriving to a buffer.
- The rate \( \lambda \) 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

$$P_n / P(t) = (t)^n / n!$$

For $2$ disjoint (non-overlapping) intervals, and

$$P_{n_1 + n_2} = P_{n_1} P_{n_2}$$

The time until the first arrival

$$\tau_1 \sim \text{Exp}(\lambda)$$

Picks an arbitrary starting point in time (call it 0).

Interarrival Times of a Poisson Process

Let $\tau_1 = \text{time until the next arrival}$. So

$$\tau_{-\lambda} \sim \text{Exp}(\lambda)$$

For $2$ disjoint (non-overlapping) intervals, and

$$P_{n_1 + n_2} = P_{n_1} P_{n_2}$$

The time until the first arrival

$$\tau_1 \sim \text{Exp}(\lambda)$$

Picks an arbitrary starting point in time (call it 0).
Interarrival Times of a Poisson Process (cont.)

- Let $\tau_2$ = the length of time between 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. $\tau_2$ is exponential and independent of $\tau_1$!
- Similarly define $\tau_3$ as the time between the second and third arrival; $\tau_4$ as the time between the third and fourth arrival; . . .
- The random variables $\tau_1, \tau_2, \tau_3, \ldots, \tau_n, \ldots$ are called the interarrival times of the Poisson process.

Interarrival Times of a Poisson Process (cont.)

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

![Diagram showing interarrival times](image_url)
A queue is a mathematical model for a waiting line. It is denoted as $M/M/1$, where $M$ stands for Poisson (exponential interarrival times), $M$ for exponential (or arbitrary) service distribution, and $1$ for a single server.

Queueing Notation

$X$ is a symbol representing the interarrival process.

$D$ is a symbol representing the service distribution.

$G$ is a symbol representing the general distribution.

$m$ is the number of servers.

$k$ is the number of buffer slots (omitted when $k = \infty$).

- $\lambda$ is the arrival rate.
- $\mu$ is the service rate.

The $M/M/1$ queue is the most basic and important queueing model.

1. Poisson arrivals (with rate $\lambda$).
2. Exponential service times (with mean $1/\mu$, so $\mu$ is the service rate).
3. One (1) server.
4. An infinite length buffer.
Aside: The D/D/1 Queue

- The D/D/1 queue has
  - Deterministic arrivals (periodic with period = 1/λ).
  - Deterministic service times (each service takes exactly 1/μ).
  - As well as 1 server and an infinite length buffer.
- If λ < μ 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 pn be the steady state probability of finding n customers waiting in the system (including the server).
- How to find pn? The state diagram:
State Analysis of an M/M/1 Queue (cont.)

- If this system is stable (i.e. \( p_n \neq 0 \) for each \( n \)), then in 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.
- Thus we obtain the balance equations

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

State Analysis of an M/M/1 Queue (cont.)

- Let's solve the balance equations:  
  \[ p_n \lambda = p_{n+1} \mu \]
- For \( n = 0 \) we get

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

\[
p_1 = \rho p_0
\]

- Similarly

\[
p_2 = \rho p_1 = \rho^2 p_0
\]

- And in general

\[
p_n = \]

State Analysis of an $M/\text{M}/1$ Queue (cont.,)

Finally note that $d = 0$, i.e., $u_d(d - 1) = u_d$

Also, the server utilization

$\mu = \text{probability that the server is working}$

$\rho = \text{probability that the queueing system is NOT empty}$

$\mu = 1 - \rho$

Note that requiring $\mu > 1$ for stability (i.e., $\mu > d$)

$\mu < 1$

makes intuitive sense.

$\mu > \frac{1}{\rho}$

Geometric distribution.

For $i = 1, 2, 3, \ldots$

$u_d(d - 1) = u_d$

and $d - 1 = 0$

So we must have

$1 > d$

$1 < d$

We obtain

$0 = u_d$

We need to solve for $0d$, so we need one more equation.

$u_d(d - 1) = u_d$

Finally note that $\rho = 1$, i.e., $0d = 0d$

$u_d(d - 1) = u_d$

We have

State Analysis of an $M/\text{M}/1$ Queue (cont.,)
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:

\[
\begin{array}{c}
0 \\
\cdots \\
\cdots \\
n \\
\cdots \\
n+1 \\
\cdots \\
N
\end{array}
\]

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

\[
p_n = \rho p_{n-1} = \rho^n p_0 \quad \text{for} \quad n = 0, 1, 2, \ldots, N
\]

as before, but we get a different \( p_0 \).

The Finite Buffer Case: M/M/1/N (cont.)

- 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

\[
p_0 = \frac{1}{1 + \sum_{n=1}^{N} \rho^n} = \frac{1}{1 + \frac{\rho(1-\rho^N)}{(1-\rho)}} = \cdots = \frac{1 - \rho}{1 - \rho^{N+1}}
\]

- Note that this holds for any \( \rho \geq 0 \). No need to assume \( \rho < 1 \). We always have stability in finite buffer case.
Blocking Probability and Buffer Size

Thus, if we desire a blocking probability less than $p_N = 10^{-6}$ for $N \gtrsim 19$, while $N_d \gtrsim 10^{-6}$ for $N \gtrsim 18$.

**Example:** For $d = 0.5$, we can use $p_N = 10^{-6}$ to choose the correct buffer size.

$P_B$ is very important.

$p_B$ is called the blocking probability.

Away due to a full buffer, a packet is turned down.

Since arrivals are independent, we have

- Note that the buffer is full when $N \gtrsim N_d$.
- $P_B = \frac{1}{u}$ for $u \geq N_d$.

So in the finite buffer case, $u = N_d$.

Blocking Probability and the Right Buffer Size
Throughput in the Finite Buffer Case

$$0 \cdot d - 0.1 \cdot (1 - p)$$

So the average rate is:

- When the server is idle, the output rate is 0.
- When the server is busy, the output rate is $p \cdot (server \ is \ busy)$.

Look at the output side.

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

Throughput in the Finite Buffer Case (cont.)
The infinite buffer model is a very good approximation of a finite buffer system.

For a finite buffer, these probabilities differ by less than 2.3%.

\[
\Pr(N = 16 | P_B = 0.8) = \Pr(N = 32 | P_B = 0.8)
\]

Isn't that neat?

\[
Nd = \frac{(1+N^d - 1)}{N^d(d-1)} = \cdots = \frac{d}{0d - 1} - 1 = \beta_d
\]

Solving for we get \( \beta_d \) we get

\[
(\beta_d - 1)^r = (0d - 1)^r
\]

Equating our two formulas for we get

Aside: Derivation of Using Throughput

\[
\beta_d = Nd
\]
even whole networks.}

Little’s formula holds for very general queueing sys-

Let $T$ = times spent by a customer in a queueing

system (waiting and being served).

Little’s formula says

\[ E(T) = E(N) \]

Let $T$ = time spent by a customer in a queueing

delay

\[ \frac{d - 1}{d} = \frac{d - 1}{d} \]

So the average number in the system is

the number in the system (including the server).

Let’s look again at the M/M/1 queueing system.

How long is that line?
Little's Formula and Queueing Delay (cont.)

Little's Formula is either deep or obvious. Intuition:

\[ \frac{n}{t} - (L)E = (\lambda)E \]

Sometimes when we consider the waiting time \( W \), i.e. the time spent waiting in the queue (not in service), then

\[ W = \frac{d - t}{d} = \frac{\chi}{(u)E} = (L)E \]

Sometimes we consider \( (L)E \) is measured in units of time. Sometimes it is

\[ \frac{\chi - n}{t} = \frac{(d - t)\chi}{d} = \frac{\chi}{(u)E} = (L)E \]

Let's apply Little to the \( M/M/1 \) queue

Let \( \text{Found on arrival, i.e. } \lambda \chi = (L)E \).

\( u \) is the average number of customers behind on departure should equal the average number in the system.

\( d \) during its time in the system.

Thus, \( (L)E \) customers should have arrived for \( E \).

When it leaves the system, it has been in the system

\( (u)E \) customers waiting.

When it arrives to the queuing system, it should find

Pick a "typical customer".

Little's Formula is either deep or obvious. Intuition:
Single Link Example

- Poisson packet arrivals with rate $\lambda = 2000 \text{ p/s}$.
- Fixed link capacity $C = 1.544 \text{ Mb/s}$ (T1 Carrier rate).
- We approximate the packet length distribution by an exponential with mean $L = 515 \text{ b/p}$.
- Thus the service time is exponential with mean

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

i.e. packets are served at a rate of $\mu = 3000 \text{ p/s}$.

Single Link Example (cont.)

- 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 Queueing Models (cont.)

\[ 0 < d \quad \frac{\lambda}{\gamma} d^0 \quad \frac{\lambda}{\gamma} d = B \]

- Erlang Loss Formula
  - Blocking probability is given by the Erlang B (or
  - is blocked (gets a busy signal).
- Any customer (a call) which doesn't get a circuit
- Models a trunk line with K circuits available.
- Important model in circuit switched networks.
- Important model in each server.
  - M/M/K/K for K > 1. One or more servers, no buffers

Other Queueing Models (cont.)

with same total capacity.
- Has worse performance at lower loads than M/M/1
- Packed with K = 24.
- (E.g., a T1 carrier is typically time division multiplexing
  channels, either physically or through multiplexing) a link which is made up of multiple
- \[ M/M/K \quad \text{for } K < 1. \quad \text{Multiple servers.} \]
- There are many other important queueing models

Other Queueing Models
Other Queueing Models (cont.)

- This is one motivation for fixed-packet-length sys-
- tems like ATM.
- Under heavy load (\( \mu \approx \sigma \)), \( M/M/1 \) is
  has half the delay

\[
I > d \quad \left( \frac{\bar{X}}{d} - 1 \right) \left( \frac{d - I}{d} \right) = (u) E
\]

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

\( M/D/I \).

Other Queueing Models (cont.)

- Can apply Little's formula to get the mean delay.
- Under heavy load, \( \sigma^2 \) is the variance of the service time dis-
- tribution. Again, variability (randomness) causes

\[
I > d \quad \left[ \left( \frac{\bar{X}}{d} - 1 \right) \frac{\sigma^2}{d} - 1 \right] \left( \frac{d - I}{d} \right) = (u) E
\]

via the Pollaczek-Khintchine (P-K) formula

- Can still compute the mean number in the system.

\( M/G/I \).

- Arbitrary service (packet length) distribution.
Queuing theory is also used in analysis of operating systems, e.g., in CSCL-6140. Queueing theory is also used in analysis of operating systems. Queueing theory is also used in analysis of operating systems. Queueing theory is also used in analysis of operating systems.

Applications.

ECSE-6820/DESE-6820, Queueing (sic) systems &

See Schwartz (Ch. 2), Kleinrock (Vol. I & II) or take

Many others.

With "vacations."

With more general arrival processes.

With priority.

Can also model and analyze other queuing systems

Other Queuing Models (cont.)