18-847F: Special Topics in Computer Systems

Foundations of Cloud and Machine Learning Infrastructure
Lecture 3: Basics of Queueing Theory

Foundations of Cloud and Machine Learning Infrastructure
Announcements

HW1 will be released tonight. Due on Sept 16\textsuperscript{th} on Gradescope.

1 Programming problem on MapReduce – Submit code on Canvas

Problem 1 of the HW is due on Sept 9\textsuperscript{th}, before the effective presentations workshop

TA Office Hours: Tues 11:30 am- 12:30 pm, CIC 4112, Bellefield Room

Piazza open for discussions: Search 18-847F
Queueing Theory
Reference Textbooks

- Performance Modeling and Design of Computer Systems
- Stochastic Processes
Queueing Terminology

Mean Service Time \( E[S] = \frac{1}{\mu} \)
Mean Waiting Time \( E[W] \)
Mean Response Time \( E[T] = E[W] + E[S] \)
Mean # Customers in Queue \( E[N] \)
Server Utilization or Load \( \rho = \frac{\lambda}{\mu} \)
Exercise: First-come first-served Queue

Arrival Rate $\lambda = 0.5$ →  Service rate $\mu = 1.0$

$t = 0$ Yellow job arrives
$t = 2.5$ Blue job leaves
$t = 4$ Green job leaves
$t = 5$ Yellow job leaves at time $t = 5$

Q1: Waiting Time $W$ of the yellow job?
Q2: Service Time $S$ of the yellow job?
Q3: Response time $T$ of the yellow job?
Q4: Load on the system? What happens if $\lambda = 1.1$?
Processor-Sharing Queues

- Service rate $\mu = 1.0$

  - $t = 0$  Blue job arrives
  - $t = 0.5$  Green job arrives
  - $t = 1.5$  Yellow job arrives
  - $t = 1.5$  Blue job leaves
  - $t = 2.5$  Green job leaves
  - $t = 3.0$  Yellow job leaves
First-come First-served vs. Processor-Sharing

Which is better in terms of $E[T]$?

Suppose that all jobs arrive at time $t = 0$, and service time is deterministic, 1 sec per job
First-come First-served vs. Processor-Sharing

Which is better in terms of E[T]?

Suppose that all jobs arrive at time t = 0, and service time is deterministic, 1 sec per job

\[
E[T_{\text{blue}}] = 1.0 \\
E[T_{\text{green}}] = 2.0 \\
E[T_{\text{yellow}}] = 3.0
\]

\[
E[T_{\text{blue}}] = 3.0 \\
E[T_{\text{green}}] = 3.0 \\
E[T_{\text{yellow}}] = 3.0
\]

Then why use processor-sharing?
- To avoid starving small jobs that get stuck behind large ones
- For jobs that interact with each other
We will focus on FCFS jobs in this lecture

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Service Time</td>
<td>( E[S] = \frac{1}{\mu} )</td>
</tr>
<tr>
<td>Mean Waiting Time</td>
<td>( E[W] )</td>
</tr>
<tr>
<td>Mean Response Time</td>
<td>( E[T] = E[W] + E[S] )</td>
</tr>
<tr>
<td>Mean # Customers in Queue</td>
<td>( E[N] )</td>
</tr>
<tr>
<td>Server Utilization or Load</td>
<td>( \rho = \frac{\lambda}{\mu} )</td>
</tr>
</tbody>
</table>
Design Question 1
What if the arrival rate doubles?

Job Arrival Rate $\lambda = 3$  

Service rate $\mu = 5$

Mean Response Time $T = \text{Waiting time in Queue + Service Time}$

Q: If $\lambda$ doubles, do you need a server of 2x rate to achieve the same $E[T]$?
Design Question 2
Many slow, or one fast server?

Choose first idle server

Q: Which of the two systems gives lower $E[T]$?
Design Question 3
Many slow, or one fast server?

Q: Which of the two systems gives lower $E[T]$?
Little’s Law

Theorem: For any ergodic open system we have

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

Very general and hence powerful law

• Any # of servers, scheduling policy, queue size limit

Some Variants

\[ E[N_w] = \lambda \ E[W] \]
\[ \rho = \lambda \ E[S] \]
A professor takes 2 new students in even-numbered years, and 1 new student in odd-numbered years.

If avg. graduation time = 6 yrs, how many students will the professor have on average?
A professor takes 2 new students in even-numbered years, and 1 new student in odd-numbered years.

If avg. graduation time = 6 yrs, how many students will the professor have on average?

\[ E[N] = \lambda E[T] \]
\[ = 1.5 \times 6 \]
\[ = 9 \]
Kendall’s Notation

Arrival
Rate $\lambda$

Service rate $\mu$

ARRIVAL DIST.
M: Poisson
$E_k$: Erlang
G: General

A / S / n

SERVICE DIST.
M: Exponential
D: Deterministic
G: General

Number of servers
Kendall’s Notation

Arrival Rate $\lambda$ → Service rate $\mu$

ARRIVAL DIST.
M: Poisson
$E_k$: Erlang
G: General

A/S/n/c

SERVICE DIST.
M: Exponential
D: Deterministic
G: General

Max # users allowed in queue
Number of servers

Max # users allowed in queue
Exercise: What are the distributions of Poisson and Exponential random variables?

Arrival Rate \( \lambda \)  \rightarrow  \text{SERVICES}  \rightarrow  \text{A/S/n/c}  \rightarrow  \text{Max # users allowed in queue}

ARRIVAL DIST.  
M: Poisson  
\( E_k \): Erlang  
G: General

SERVICE DIST.  
M: Exponential  
D: Deterministic  
G: General

Number of servers
M/M/1 Queue

WANT TO FIND

1. Mean Response Time $E[T]$
2. Mean Waiting Time $E[W]$
M/M/1: Markov Model

\[ \pi_i = \rho^i (1 - \rho) \]
\[ \pi_0 = (1 - \rho) \]

where \( \rho = \frac{\lambda}{\mu} \)

\[ \mathbb{E}[N] = \sum_{i=0}^{\infty} i\pi_i = \rho (1 - \rho) \sum_{i=1}^{\infty} i \rho^{i-1} = \frac{\rho}{1 - \rho} \]
M/M/1: Mean Response Time

\[ \mathbb{E}[N] = \sum_{i=0}^{\infty} i \pi_i = \rho (1 - \rho) \sum_{i=1}^{\infty} i \rho^{i-1} = \frac{\rho}{1 - \rho} \]

\[ \mathbb{E}[T] = \frac{\mathbb{E}[N]}{\lambda} = \frac{\rho}{\lambda (1 - \rho)} = \frac{1}{\mu - \lambda} \]

\[ \mathbb{E}[W] = \frac{1}{\mu - \lambda} - \frac{1}{\mu} = \frac{\rho}{\mu - \lambda} \]
Exercise: Design Question 1
What if the arrival rate doubles?

Job Arrival Rate $\lambda = 3$ → Service rate $\mu = 5$

Mean Response Time $T = $ Waiting time in Queue + Service Time

Q: If $\lambda$ doubles, do you need a server of 2x rate to achieve the same $E[T]$?

A: Service rate $6 + 2 = 8$ is sufficient
Exercise: M/M/1 Queue
What if the service rate doubles?

Q: Is the first queue twice (or more) longer than the second?

What is $E[W^{(A)}] / E[W^{(B)}]$ as a function of $\rho = \lambda/\mu$?
Exercise: M/M/1 Queue

What if the service rate doubles?

Q: Is the first queue twice (or more) longer than the second?

What is $E[W^{(A)}] / E[W^{(B)}]$ as a function of $\rho = \lambda/\mu$?

**Answer:** \[ \frac{2(2 - \rho)}{1 - \rho} \]
M/M/n Queue

WANT TO FIND

1. Mean Response Time $E[T]$
2. Mean Waiting Time $E[W]$
M/M/n Queue

\[ P_Q = \sum_{i=n}^{\infty} \pi_i \]

\[ = \pi_0 \frac{n^n}{n!} \sum_{i=n}^{\infty} \rho^i \]

where \( \pi_0 = \left[ \sum_{i=0}^{n-1} \frac{(n\rho)^i}{i!} + \frac{(n\rho)^n}{n!(1 - \rho)} \right]^{-1} \]

\[ = \frac{(n\rho)^n \pi_0}{n!(1 - \rho)} \]

Erlang-C Formula

Used in call centers to determine number of agents required
M/M/n Queue

\[ \mathbb{E}[N_w] = \sum_{i=n}^{\infty} \pi_i (i - n) \]
\[ = \pi_0 \sum_{i=n}^{\infty} \frac{\rho^i n^n}{n!} (i - n) \]
\[ = P_Q \frac{\rho}{1 - \rho} \]

\[ \mathbb{E}[W] = \frac{\mathbb{E}[N_w]}{\lambda} = P_Q \frac{\rho}{\lambda (1 - \rho)} \]
\[ \mathbb{E}[T] = P_Q \frac{\rho}{\lambda (1 - \rho)} + \frac{1}{\mu} \]
Design Question 2
Many slow, or one fast server?

Q: Which of the two systems gives lower $E[T]$?
Design Question 2
Many slow, or one fast server?

Choose first idle server

\[ \mathbb{E}[T]^{M/M/n} = PQ \frac{\rho}{\lambda(1 - \rho)} + \frac{1}{\mu} \]

\[ \mathbb{E}[T]^{M/M/1} = \frac{\rho}{\lambda(1 - \rho)} \]

System Load \( \rho = \frac{\lambda}{3\mu} \)
Design Question 3
Many slow, or one fast server?

Choose first idle server

M/M/n is n times slower when \( \rho \to 0 \)

\[
\frac{\mathbb{E}[T]_{M/M/n}}{\mathbb{E}[T]_{M/M/1}} = P_Q + n(1 - \rho)
\]

M/M/n and M/M/1 are almost equal when \( \rho \to 1 \)
Design Question 3
Many slow, or one fast server?

Freq. Division Multiplexing (FDM)

M/M/1
Design Question 3
Many slow, or one fast server?

\[ E[T]^{FDM} = \frac{n}{n\mu - \lambda} \]

\[ E[T]^{M/M/1} = \frac{1}{n\mu - \lambda} \]

FDM is \( n \) times slower than M/M/1
M/G/1 Queue
Pollaczek-Khinchine Formula

\[ \mathbb{E}[T] = \mathbb{E}[X] + \frac{\mathbb{E}[X^2]}{2(1 - \lambda \mathbb{E}[X])} \]

Cannot use Markov chain analysis
Proof of PK formula

\[
\mathbb{E}[T_w] = \mathbb{E}[N_w] \cdot \mathbb{E}[X] + E[R]
\]

\[
= \lambda \mathbb{E}[T_w] \cdot \mathbb{E}[X] + \frac{\mathbb{E}[X^2]}{2}
\]

\[
= \frac{\mathbb{E}[X^2]}{2(1 - \lambda \mathbb{E}[X])}
\]
M/G/n Queue

Choose first empty queue

Service dist $F_X$

Service dist $F_X$

Service dist $F_X$

Cannot use Markov chain analysis

$$E[T] \approx E[X] + \frac{E[X^2]}{2E[X]} \cdot E[W^{M/M/n}]$$
Choose first idle server

System 1

1 \[ \mu \]

2 \[ \mu \]

k \[ \mu \]

System 2

\[ \frac{\lambda}{k} \]

\[ \frac{\lambda}{k} \]

\[ \frac{\lambda}{k} \]

System 3

\[ \lambda \]

\[ \lambda \]

\[ \lambda \]
HW1 picture