CS 4215: Quantitative Performance Evaluation for Computing Systems

Lydia Y. Chen
y.chen-10@tudelft.nl

Motivation

Course Overivew

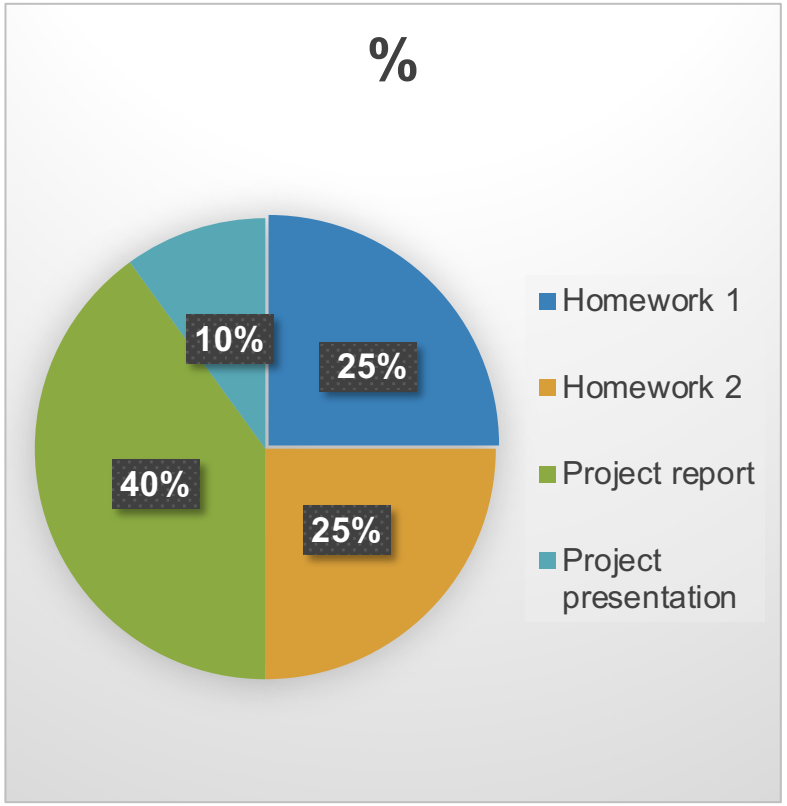
Overview - schedule

- Overview on quantitative methods and basics mathematics tools. (week 1)
- Queueing methods for modeling computing systems (week 1 and 2)
- Scheduling and load balancing (week 3)
- Machine learning methods for modeling computing systems (week 4)
- Design of experiments and simulation (week 5)
- System security and scalability analysis (week 6)
- Optimization and resource management (week 7)

Overview – assessment

- Homework (50%): 2 individual homework due in week 4 and 7. Each homework accounts 25% of the grade and cover 3 weeks material. Homework will be given in week 1 and 4. Students have three weeks time to complete them.

- Group project (50%): group project report (40%) and presentation (10%). There will be 7 predefined projects students can choose from. There will be an interim discussion with each team in week 6. The final report will be due in week 9, and 20 minutes presentation in week 9 as well.



Performance Evaluation Overview

Key metrics

- Performance metrics:
  - Average (tail) response times
  - Maximum throughput
  - Availability (failure rate)
  - Scalability ()
- Workload metrics
  - Arrival rates
  - Service demands

Architecture of computer systems

- Multiple -level programming
- Network protocol

Operational Laws & Modification Analysis

Operational laws

Utilization Law

Forced Flow Law

Little's Law

General Response Time Law

Interactive Response Time Law

Bottleneck Law

Operational Laws

- Relationships that do not require any assumptions about the distribution of service times or inter-arrival times.
- **Operational** $\Rightarrow$ Directly measured.
- **Operationally testable assumptions**
  $\Rightarrow$ assumptions that can be verified by measurements.
  - For example, whether number of arrivals is equal to the number of completions?
  - This assumption, called job flow balance, is operationally testable.
  - A set of observed service times is or is not a sequence of independent random variables is not is not operationally testable.

Terminologies

Quantities that can be directly measured during a finite observation period.

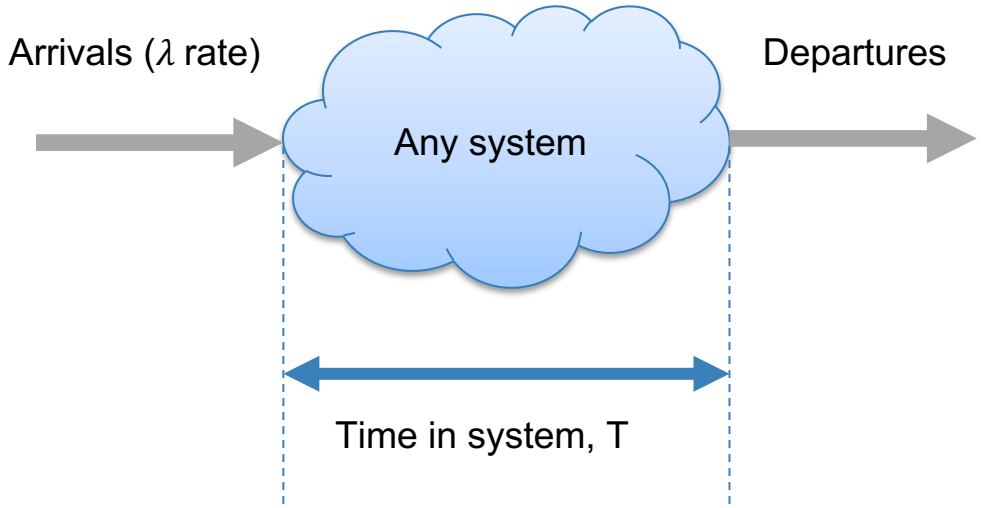
- t = Observation interval
- A_i = number of arrivals
- C_i = number of completions
- B_i = busy time B_i

Arrival rate $\lambda = \frac{A(t)}{t}$

Throughput $\mathrm{X} = \frac{C(t)}{t}$

Utilization $\mathrm{U} = \frac{B(t)}{t}$

Service time $\mathrm{S} = \frac{B}{C}$



Open v.s. Closed systems

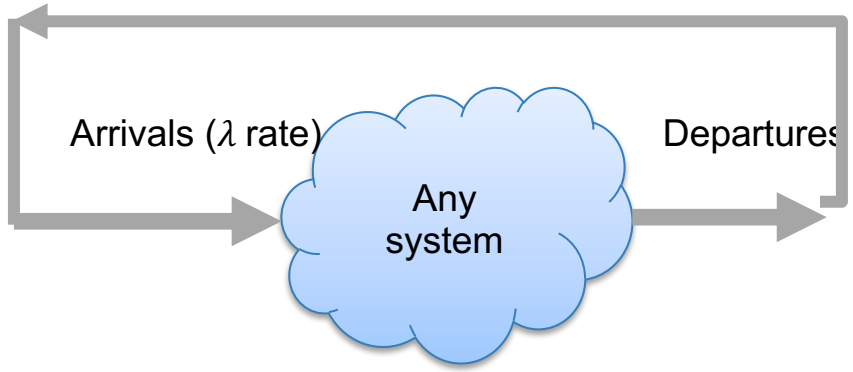
Open systems

- ▷ **External arrivals**
- ▷ **Eg., web serve requests**

.

Closed systems

- ▷ **Fix number of clients**
- ▷ **Eg., thread pool**

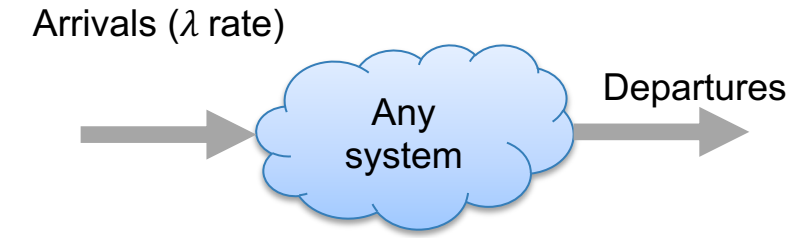


The average number of jobs in the system =
(the average arrival rate) X
(the average time a job spends in the systems)

Little's Laws

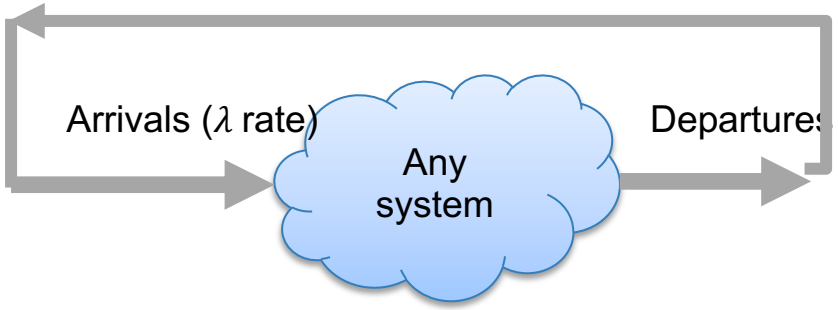
- For open systems

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

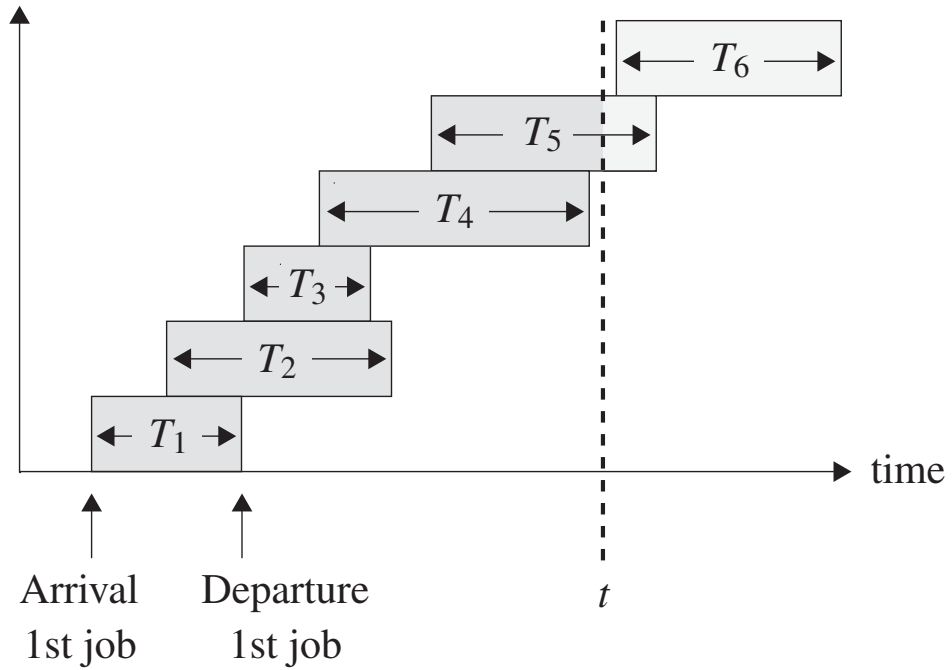


- For closed systems

$$N = XE[T]$$



Little's Law for open systems: proof sketch



Ti: time of ith job spend int the system

C(t): number of job completion till time t

A(t): number of job arrivals till time t

$$\sum_{i \in C(t)} T_i \leq \mathcal{A} \leq \sum_{i \in A(t)} T_i$$

$$\mathcal{A} = \int_0^t N(s)ds.$$

$$\sum_{i \in C(t)} T_i \leq \int_0^t N(s)ds \leq \sum_{i \in A(t)} T_i.$$

Proof sketch for Little's Law

$$\frac{\sum_{i \in C(t)} T_i}{t} \leq \frac{\int_0^t N(s)ds}{t} \leq \frac{\sum_{i \in A(t)} T_i}{t}$$

Taking limits as $t \to \infty$,

$$\lim_{t\to\infty} \frac{\sum_{i \in C(t)} T_i}{C(t)} \cdot \lim_{t\to\infty} \frac{C(t)}{t} \leq \overline{N}^{\text{Time Avg}} \leq \lim_{t\to\infty} \frac{\sum_{i \in A(t)} T_i}{A(t)} \cdot \lim_{t\to\infty} \frac{A(t)}{t}$$

$$\Rightarrow \overline{T}^{\text{Time Avg}} \cdot X \leq \overline{N}^{\text{Time Avg}} \leq \overline{T}^{\text{Time Avg}} \cdot \lambda.$$

Little's Laws

- For open systems

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

Q. Are we assuming FCFS service order?

Q. Can the system have multiple servers

- For closed systems

$$N = XE[T]$$

Question: if we are are only interested in "red" type of jobs, can we apply little's law?

Little's Laws for Time in Queue

$$\bar{N}_Q^{Time\ Avg} = \lambda \bar{T}_Q^{Time\ Avg}$$

Q. How to prove it?

$\bar{N}_Q$: the number of jobs in queue in the system

$\bar{T}_Q$: the time jobs spend in queue

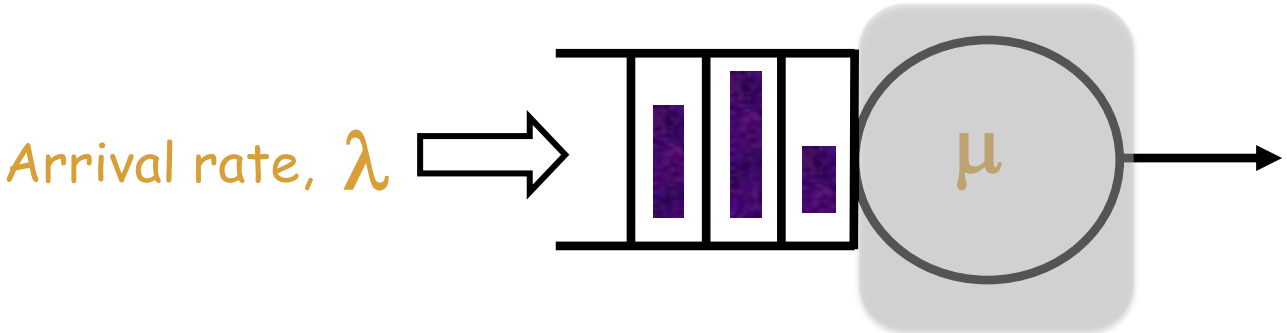
Utilization Law

$$\rho_i = \frac{\lambda_i}{\mu_i}$$

Q. How to prove it?

$\boldsymbol{\rho_i}$
=Expected number of jobs in service
= Arrival rate into service facility ·
Mean time in service facility for device i
= $\lambda_i \cdot E[Service\ time\ at\ device\ i]$
= $\lambda_i \cdot \frac{1}{\mu_i}$

Hint: let the system consist of just the service facility without the associated queue



Utilization Law

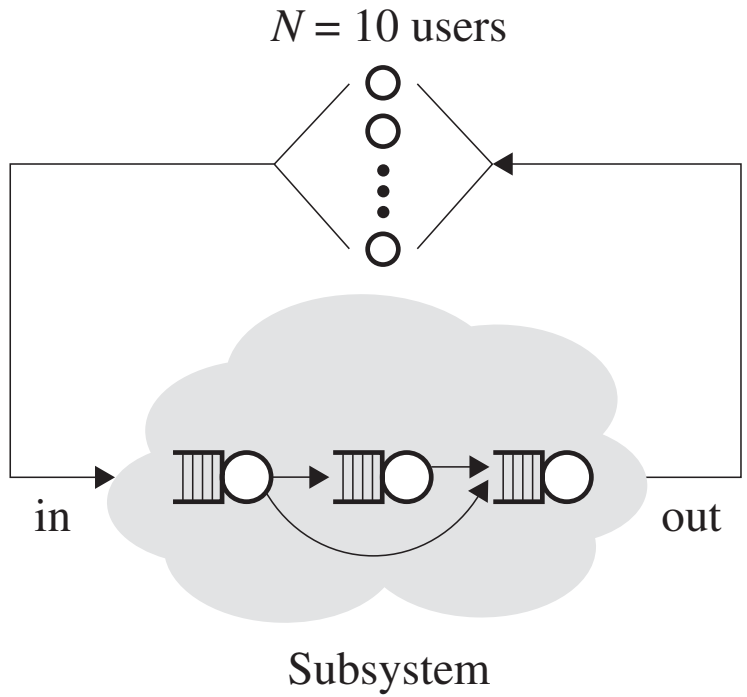
$$\rho_i = \frac{\lambda_i}{\mu_i}$$

$$\rho_i = \lambda_i E[S_i] = \lambda_i E[S_i]$$

Example 1

Interactive systems with N=10 users, shown below. The expected think time is E[Z]=5 seconds and that the expected response time is E[R]=**15** seconds, measuring from in to out.

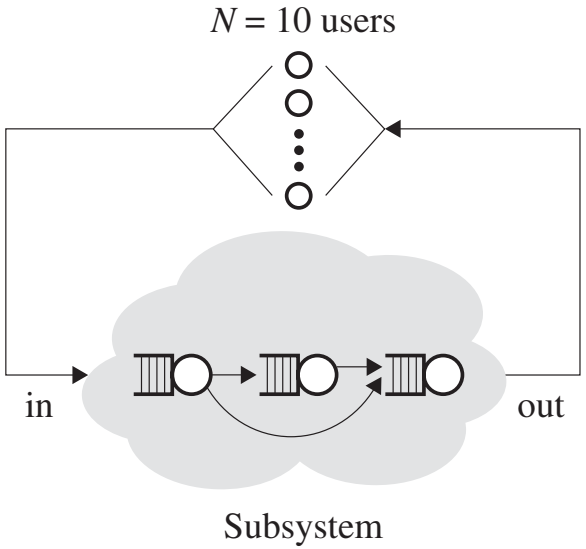
Q. What is the throughput?



Example 2

A single FCFS queue with a capacity limit of 7 jobs. Arrivals that find a full buffer are dropped?

Q. What does Littles Law look like for this systems



Forced Flow Law

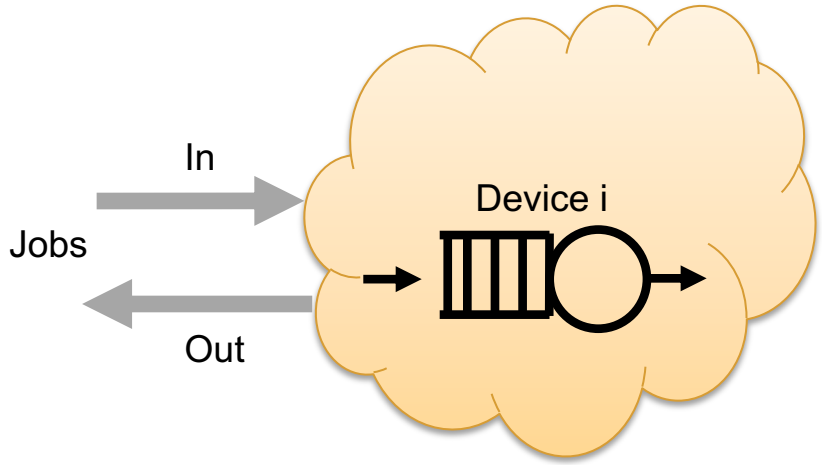
- Relates the system throughput to individual device throughputs.

$$X_i = E[V_i] \cdot X$$

V_i (visit ratio) denotes the number of visit to device i per job

Q. Intuitively make sense?

$$X_i = \frac{C_i}{t} = \frac{C_i}{C_0}\frac{C_0}{t}$$



Bottleneck Law

$$\rho_i = X \cdot E[D_i]$$

$D_i = V_i S_i$ is the total service demand on the device for all visits of a job.

- Proof: by combining the forced flow law and the utilization law, we get:

$$\rho_i = X_i \cdot E[S_i] = X \cdot E[V_i] \cdot E[S_i] = X \cdot E[D_i]$$

- The device with the highest D_i has the highest utilization and is the **bottleneck device**.

Bottleneck analysis

- Improving the bottleneck device will provide the highest payoff in terms of system throughput.
- Improving other devices will have little effect on the system performance.
- Identifying the bottleneck device should be the first step in any performance improvement project.

In-class exercise

An interactive system with following characteristics

- 25 terminals (N=25)
- 18 seconds average think time
- 20 visits to a specific SSD per interaction on average
- 30 % utilization of that SSD
- 0.025 seconds average service time per visit to that disk

Question: what is the mean response time

$$E[R] = \frac{N}{X} \cdot E[Z]$$

$$X = \frac{X_{disk}}{E[V_{disk}]} = \frac{\frac{\rho_{disk}}{E[S_{disk}]}}{E[V_{disk}]}$$

Asymptotic bounds for closed systems

$$X \leq \min\left(\frac{N}{D + E[Z]}, \frac{1}{D_{max}}\right)$$

$$E(R) \geq \max(D, N\,D_{max} - E[Z])$$

Intuitive meaning?

- Here, D is the sum of total service demands on all devices except terminals.

Proof for asymptotic bounds

- The asymptotic bounds are based on the following observations:
- The utilization of any device cannot exceed one. This puts a limit on the maximum obtainable throughput.
- The response time of the system with *N* users cannot be less than a system with just one user. This puts a limit on the minimum response time.

Proof (Cont)

- The interactive response time formula can be used to convert the bound on throughput to that on response time and vice versa.
- For the bottleneck device *b* we have:

$$U_b = X \cdot D_{max}$$

- Since U_b cannot be more than one, we have:

$$U_b = X \cdot D_{max} \leq 1$$

$$X \leq \frac{1}{D_{max}}$$

Proof (Cont)

- With just one job in the system, there is no queueing and the system response time is simply the sum of the service demands:

$$R(1) = D_1 + D_2 + \cdots + D_M = D$$

- Here, *D* is defined as the sum of all service demands.
- With more than one user there may be some queueing and so the response time will be higher. That is:

Proof (Cont)

- Applying the interactive response time law to the bounds:

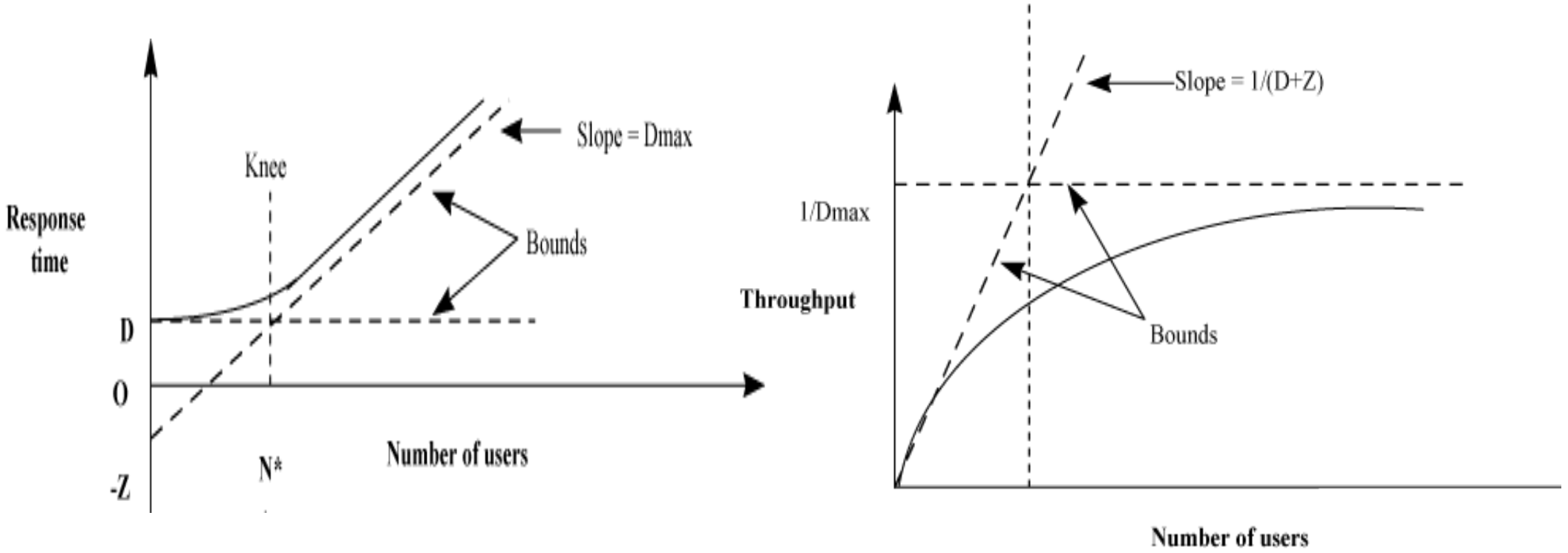
$$R(N) \geq D$$

- Combining these bounds we get the asymptotic bounds.

$$R(N) = \frac{N}{X(N)} - Z \geq ND_{max} - Z$$

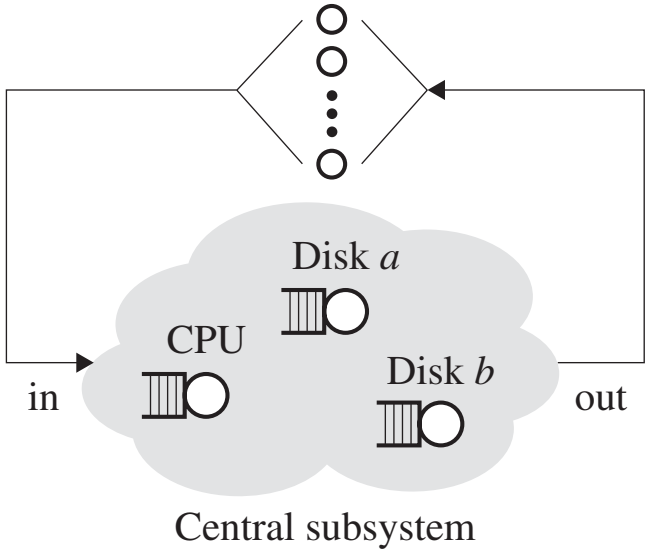
$$X(N) = \frac{N}{R(N) + Z} \leq \frac{N}{D + Z}$$

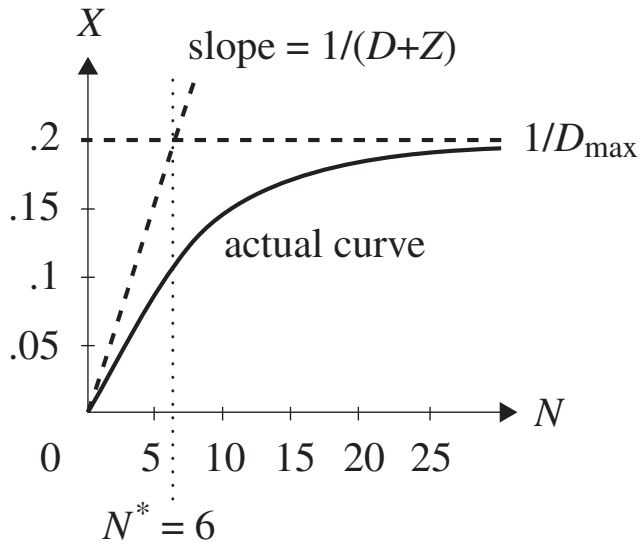
Typical Asymptotic Bounds



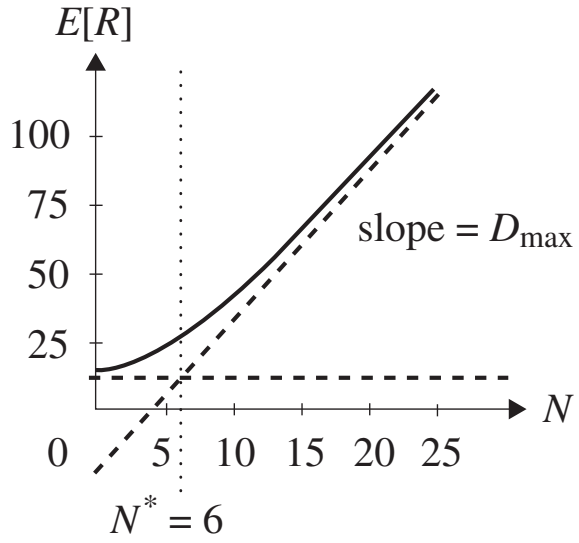
Asymptotic example

- E[Z]=18
- $E[D_{cpu}] = 5$
- $E[D_{disk\ a}] = 4$
- $E[D_{disk\ b}] = 3$





(a) X versus N



(b) **E** $[R]$ versus N

Typical Asymptotic Bounds (Cont)

- The number of jobs N^* at the knee is given by:

$$D = N^* D_{max} - Z$$

$$N^* = \frac{D + Z}{D_{max}}$$

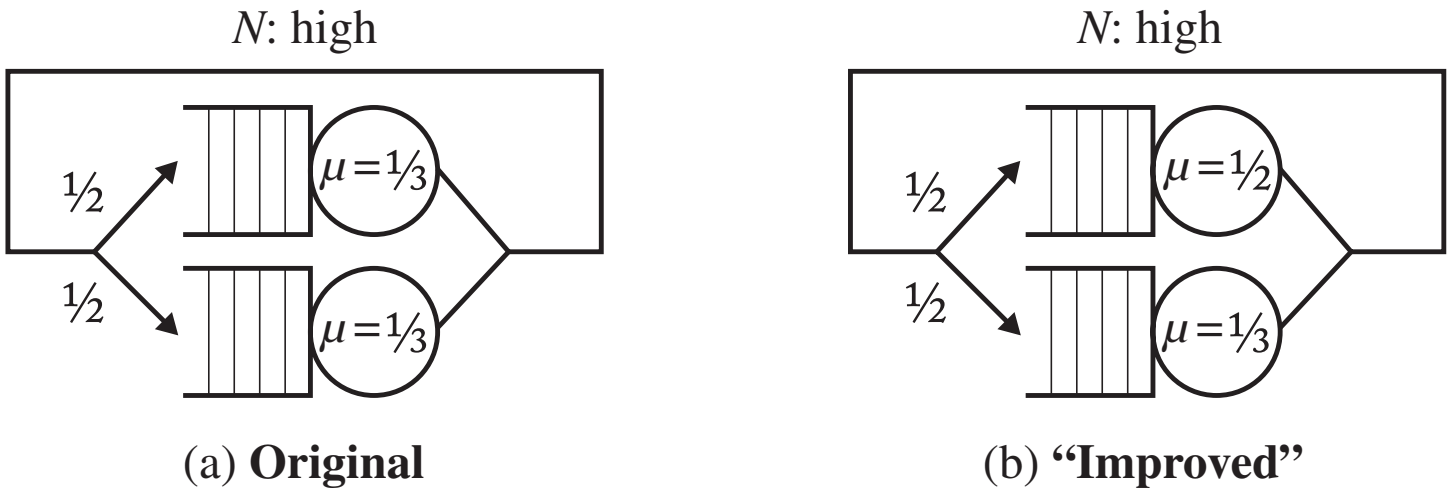
- If the number of jobs is more than N^*, then we can say with certainty that there is queueing somewhere in the system.
- The asymptotic bounds can be easily explained to people who do not have any background in queueing theory or performance analysis.

Question

- How much does the throughput improve from (a) to (b)? How about the response time?

No Change! Why?

Bottleneck remains



(a) **Original** (b) **"Improved"**

More on modification analysis

- T=650 seconds
- $B_{cpu} = 400$ seconds
- $B_{slow\ disk} = 100\ seconds$
- $B_{fast\ disk} = 600\ seconds$
- $C = C_{cpu} = 200\ jobs$
- $C_{slow\ disk} = 2000\ jobs$
- $C_{fast\ disk} = 20000\ jobs$
- $E[Z] = 15\ seconds$
- N=20 users

Faster CPU?
Replace it with twice faster one

Balancing slow and fast disks?

Second fast disk?

More on modification analysis

Faster CPU?
Replace it with twice faster one

- T=650 seconds
- $B_{cpu} = 400$ seconds
- $B_{slow\ disk} = 100\ seconds$
- $B_{fast\ disk} = 600\ seconds$
- $C = C_{cpu} = 200\ jobs$
- $C_{slow\ disk} = 2000\ jobs$
- $C_{fast\ disk} = 20000\ jobs$
- $E[Z] = 15\ seconds$
- N=20 users

- $D_{cpu} = 2\ sec/job$
- $D_{slow\ disk} = 0.5\ sec/job$
- $D_{fast\ disk} = 3sec/job$
- $E[V_{cpu}] = 1\ visit/\ job$
- $E[V_{slow\ disk}] = 10\ visit/\ job$
- $E[V_{fast\ disk}] = 100\ visit/\ job$
- $E[S_{cpu}] = 2\ sec/\ vist$
- $E[S_{\text{slow}\ disk}] = 0.5sec/\ vist$
- $E[S_{fast\ disk}] = .03sec/\ vist$

More on modification analysis

- T=650 seconds
- $B_{cpu} = 400$ seconds
- $B_{slow\ disk} = 100\ seconds$
- $B_{fast\ disk} = 600\ seconds$
- $C = C_{cpu} = 200\ jobs$
- $C_{slow\ disk} = 2000\ jobs$
- $C_{fast\ disk} = 20000\ jobs$
- $E[Z] = 15\ seconds$
- N=20 users

- $D_{cpu} = 2\ sec/job$
- $D_{slow\ disk} = 0.5\ sec/job$
- $D_{fast\ disk} = 3sec/job$
- $E[V_{cpu}] = 1\ visit/\ job$
- $E[V_{slow\ disk}] = 10\ visit/\ job$
- $E[V_{fast\ disk}] = 100\ visit/\ job$
- $E[S_{cpu}] = 2\ sec/\ vist$
- $E[S_{\text{slow } disk}] = 0.5sec/\ vist$
- $E[S_{fast\ disk}] = .03sec/\ vist$

Faster CPU?
Replace it with twice faster one

More on modification analysis

- T=650 seconds
- $B_{cpu} = 400$ seconds
- $B_{slow\ disk} = 100\ seconds$
- $B_{fast\ disk} = 600\ seconds$
- $C = C_{cpu} = 200\ jobs$
- $C_{slow\ disk} = 2000\ jobs$
- $C_{fast\ disk} = 20000\ jobs$
- $E[Z] = 15\ seconds$
- N=20 users

- $D_{cpu} = 2\ sec/job$
- $D_{slow\ disk} = 0.5\ sec/job$
- $D_{fast\ disk} = 3sec/job$
- $E[V_{cpu}] = 1\ visit/\ job$
- $E[V_{slow\ disk}] = 10\ visit/\ job$
- $E[V_{fast\ disk}] = 100\ visit/\ job$
- $E[S_{cpu}] = 2\ sec/\ vist$
- $E[S_{\text{slow } disk}] = 0.5sec/\ vist$
- $E[S_{fast\ disk}] = .03sec/\ vist$

Balancing slow and fast disks?

More on modification analysis

- T=650 seconds
- $B_{cpu} = 400$ seconds
- $B_{slow\ disk} = 100\ seconds$
- $B_{fast\ disk} = 600\ seconds$
- $C = C_{cpu} = 200\ jobs$
- $C_{slow\ disk} = 2000\ jobs$
- $C_{fast\ disk} = 20000\ jobs$
- $E[Z] = 15\ seconds$
- N=20 users

- $D_{cpu} = 2\ sec/job$
- $D_{slow\ disk} = 0.5\ sec/job$
- $D_{fast\ disk} = 3sec/job$
- $E[V_{cpu}] = 1\ visit/\ job$
- $E[V_{slow\ disk}] = 10\ visit/\ job$
- $E[V_{fast\ disk}] = 100\ visit/\ job$
- $E[S_{cpu}] = 2\ sec/\ vist$
- $E[S_{\text{slow } disk}] = 0.5sec/\ vist$
- $E[S_{fast\ disk}] = .03sec/\ vist$

Second fast disk?

Summary

- Little's Law
- Utilization Law
- Forced Flow Law
- Asymptotic bounds

$$\mathrm{E}[N] = \lambda E[T]$$

$$\rho_i = X \cdot E[D_i]$$

$$X_i = E[V_i] \cdot X$$

$$X \leq \min\left(\frac{N}{D + E[Z]}, \frac{1}{D_{max}}\right)$$

$$E(R) \geq max(D, N\, D_{max} - E[Z])$$

Related homework

Exercise 6.4, 7.4

Next lecture:

Thanks!

Any questions?

y.chen-10@tudelf.nl
lydiaychen@ieee.org

Presentation design

This presentations uses the following typographies and colors:

- Titles: **Raleway**
- Body copy: **Lato**

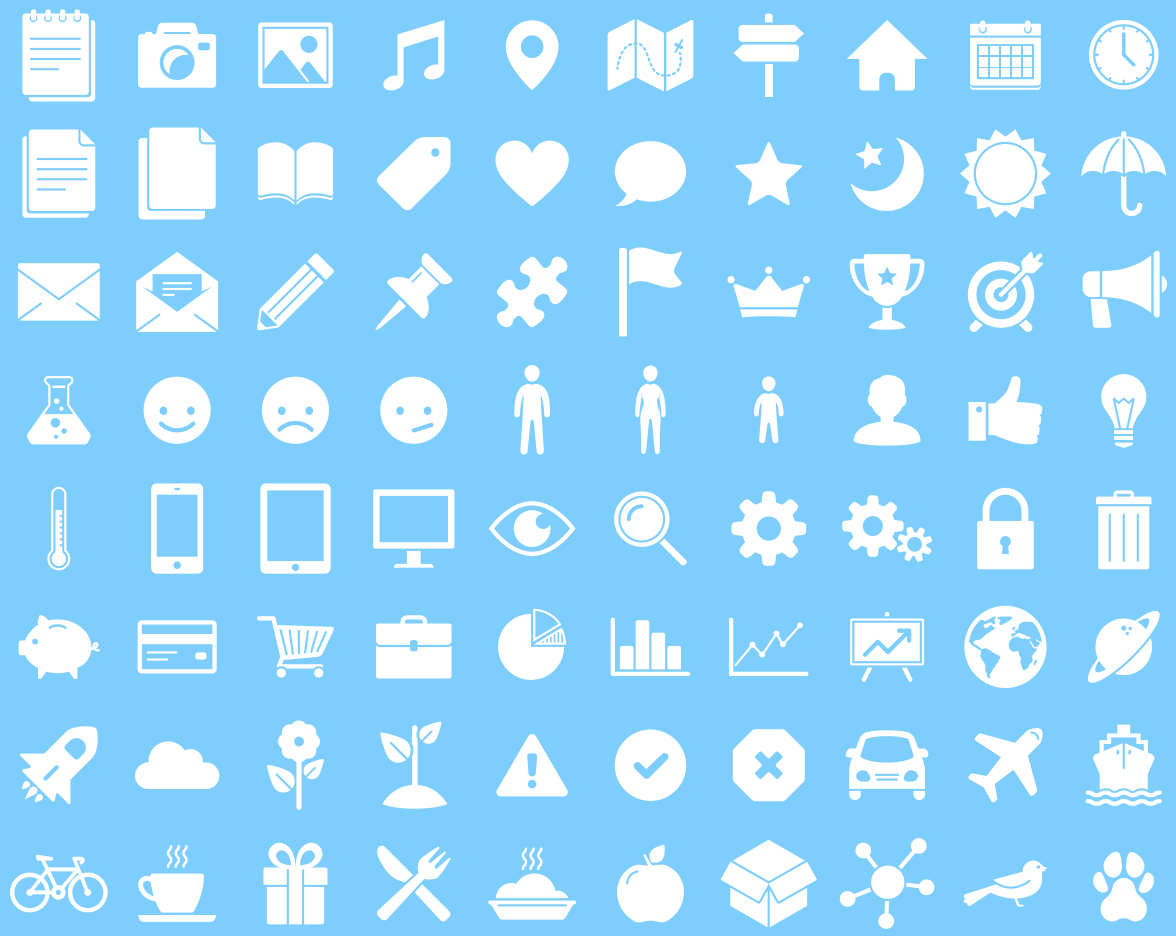
You can download the fonts on these pages:

https://www.fontsquirrel.com/fonts/raleway

https://www.fontsquirrel.com/fonts/lato

- Dark blue **#2185c5**
- Light blue **#7ecefd**
- Yellow **#ff9715**
- Magenta **#f20253**
- Dark gray **#677480**
- Light gray **#97abbc**

You don't need to keep this slide in your presentation. It's only here to serve you as a design guide if you need to create new slides or download the fonts to edit the presentation in PowerPoint®



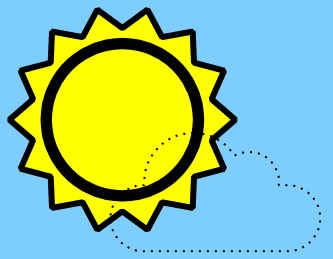
SlidesCarnival icons are editable shapes.

This means that you can:

- Resize them without losing quality.
- Change fill color and opacity.
- Change line color, width and style.

Isn't that nice? :)

Examples:



Now you can use any emoji as an icon!
And of course it resizes without losing quality and you can change the color.

How? Follow Google instructions
https://twitter.com/googledocs/status/730087240156643328

and many more...