Operating systems

Scheduling 2: queueing theory

Last time…

- CPU scheduler
- simple scheduling policies
  - FCFS
  - SJF
  - RR
- more realistic
  - multi-level, multi-level with feedback
  - Linux scheduler

Today…

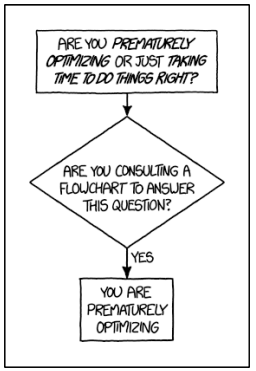
- scheduling criteria
- how to evaluate algorithms
- **queueing theory**

Scheduling criteria

- waiting time
- response time (or delay)
- throughput
- predictability
- scheduling overhead
- starvation

How to evaluate a scheduling algorithm?

1. deterministic modeling
2. queueing models
3. simulations
4. implementation

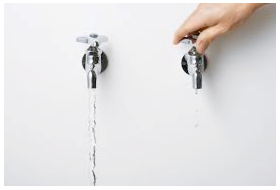


https://xkcd.com/1691/

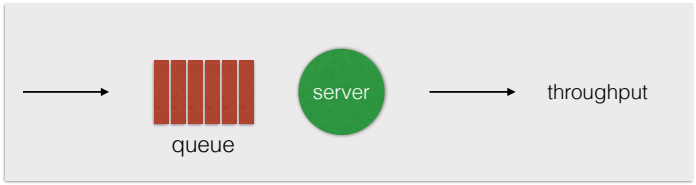
queuing theory

We will assume

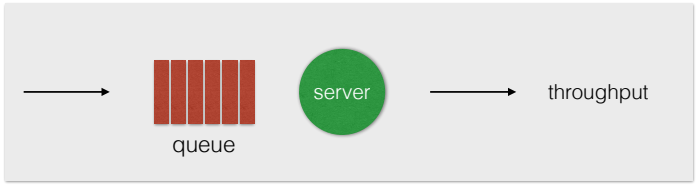
- FIFO scheduling
- all tasks that arrive will eventually be serviced



our system
is assumed to be
work-conserving



queueing delay + service time = response time

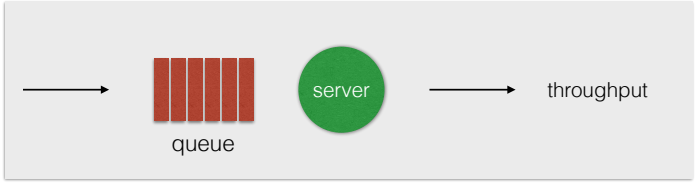


arrival rate

$$U = \min\left(1, \frac{\lambda}{\mu}\right)$$

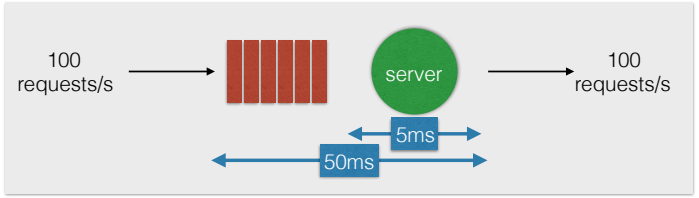
utilization

service rate = $\frac{1}{\text{service time}}$



$$U = \min\left(1, \frac{\lambda}{\mu}\right)$$

$$X = U\mu$$



What is the queueing delay?

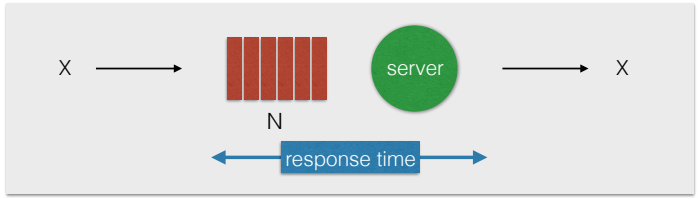
Little's law

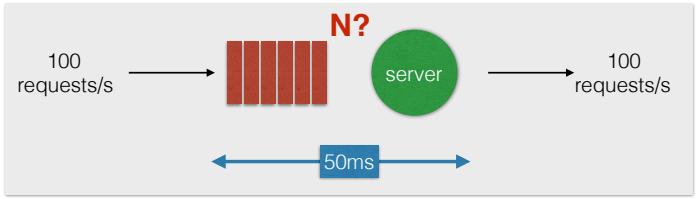
$$N = XR$$

number of tasks in the system

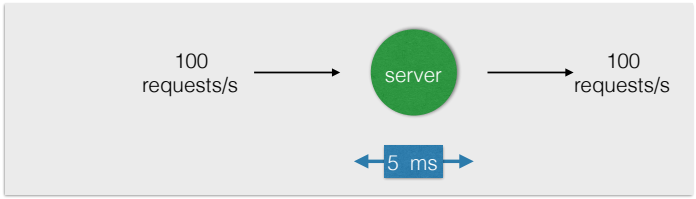
throughput

response time



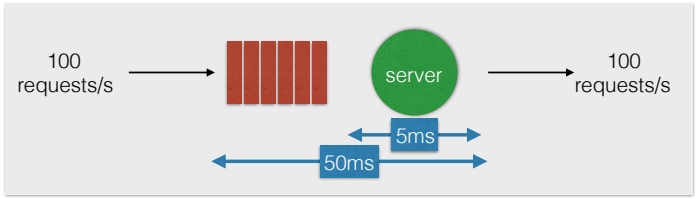


$$N = XR$$



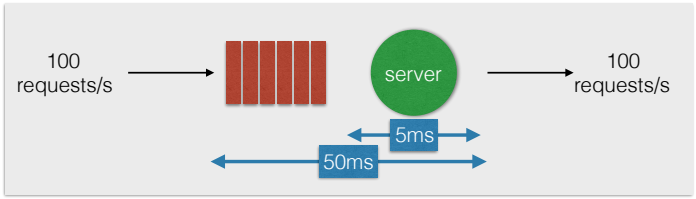
What is the average utilization of the server?

$$N = XR$$



1. How long does an average request spend in the queue?
2. How many requests are in the queue?

$$N = XR$$

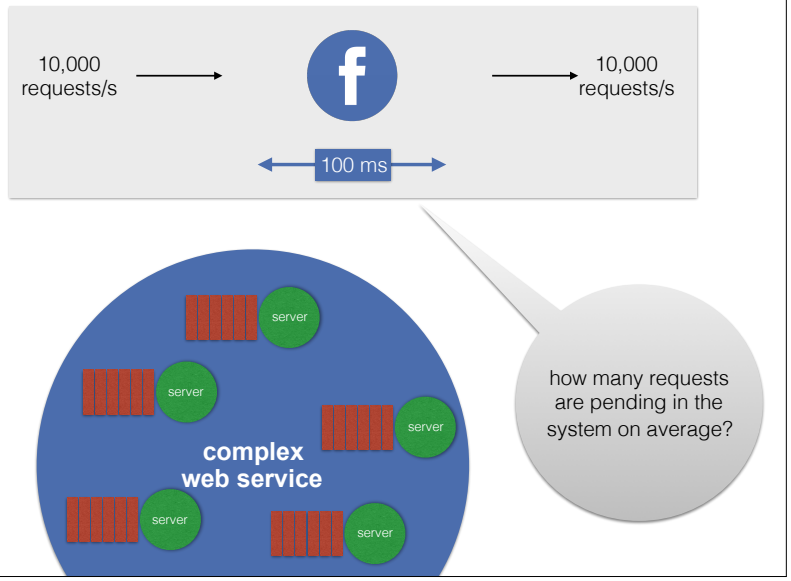


Average number of tasks in queue: 4.5

execution time: 5 ms

Why is queueing delay 4.5 ms and not 4.5 x 5 ms = 22.5 ms?

$$N = XR$$



Scheduling criteria

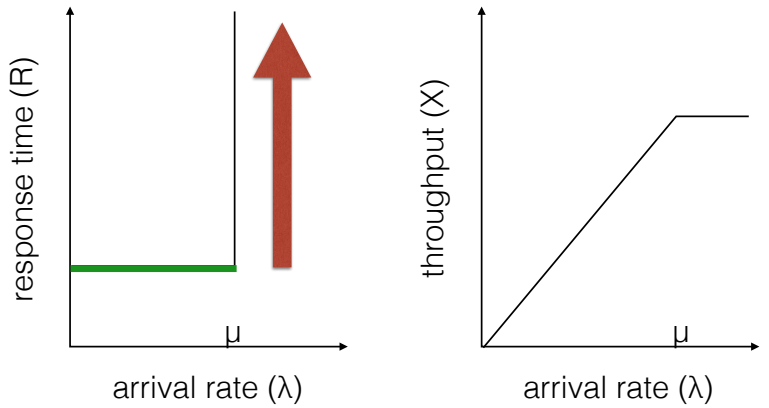
- waiting time
- response time (or delay)
- throughput
- predictability
- scheduling overhead
- starvation

what about
utilization?

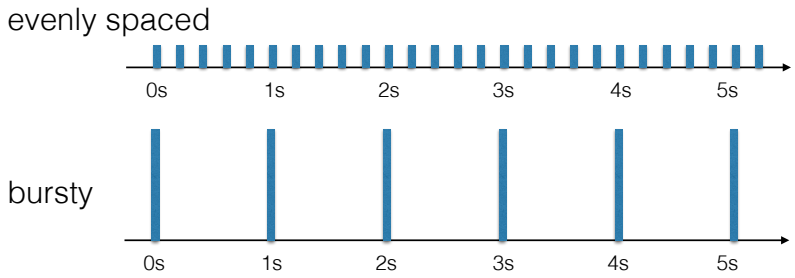
no free lunch

getting real
close to an
unstable system

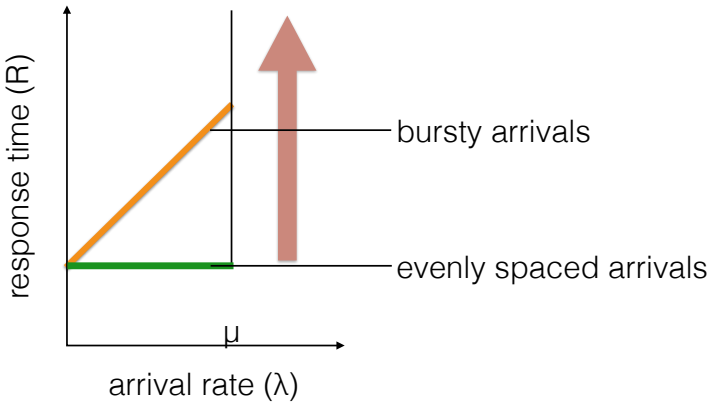
extreme scenario 1: evenly spaced arrivals



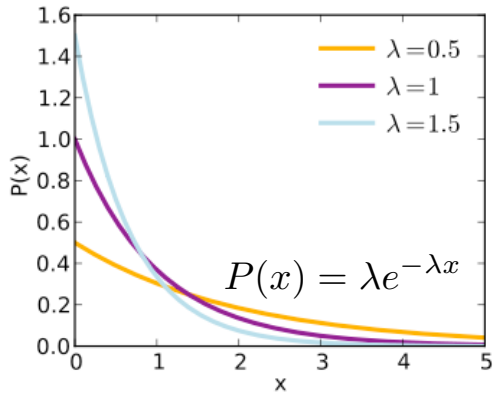
example: λ = 5 requests/s



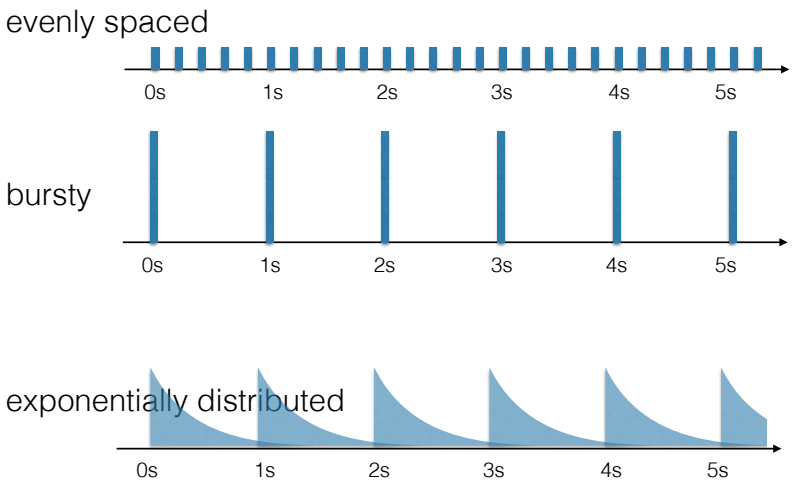
extreme scenario 2: bursty arrivals



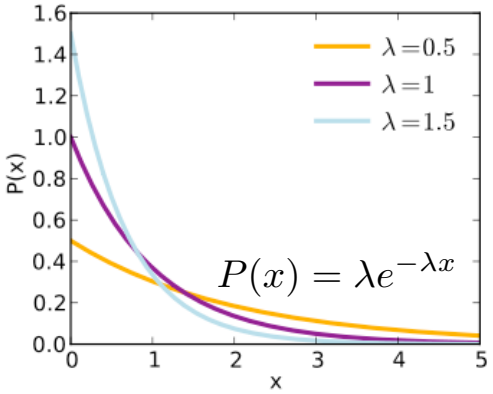
Exponential distribution



example: λ = 5 requests/s



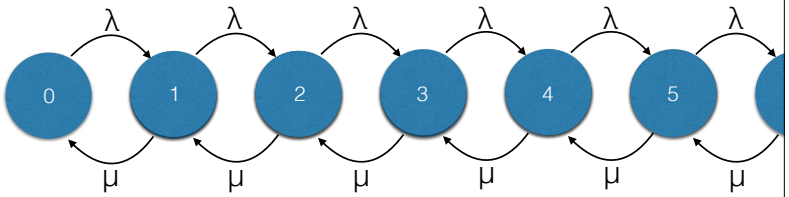
Exponential distribution



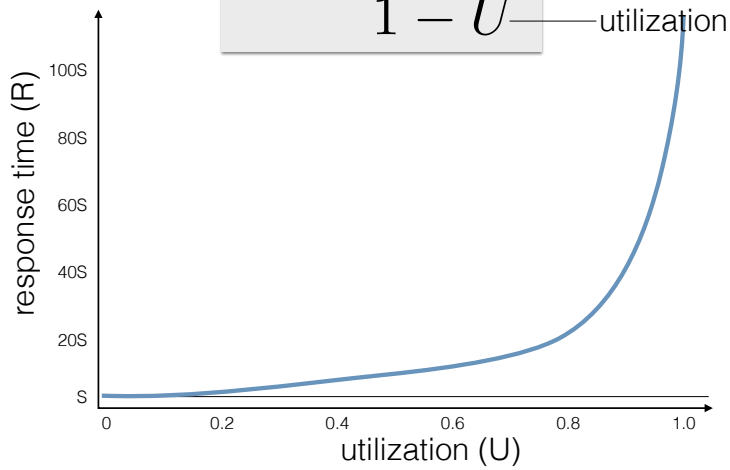
- memoryless
- light-tailed

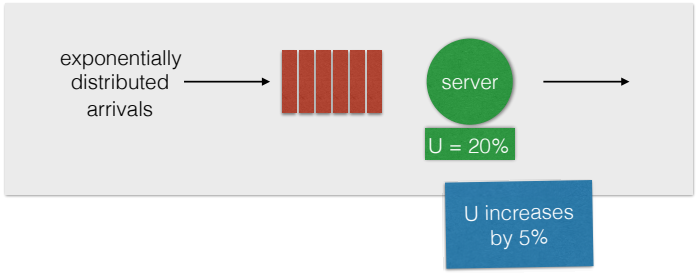
arrival time:
memory-full or
memoryless
probability distribution?

if the arrival process is a memoryless distribution



response time — $R = \frac{S}{1-U}$ — service time; utilization

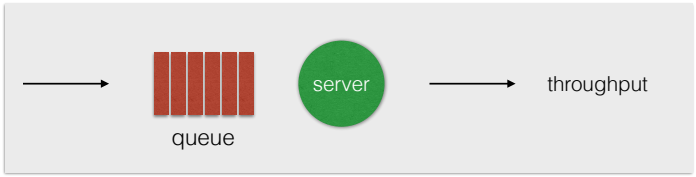




1. by how much does the response time increase?
2. how does that increase compare to the case when utilization goes from 90% to 95%?

$$R = \frac{S}{1-U}$$

in summary…



Little's law $N = XR$

response time ⟷ utilization