if $n \leq k$, then all n are served

\# customers in line

\# servers

n competing clocks rate $n\lambda$

if $n > k$, then only k are being served

k competing clocks rate $k\lambda$

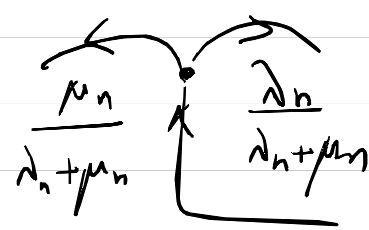
Called M/M/k

- M/M/∞: ∞ly many servers

then $\lambda_n = \lambda \quad \mu_n = n\mu \quad \forall n \geq 0$

Lec 34

Discrete time version:



clock rate $\lambda_n + \mu_n$

in all examples: $\lambda_n + \mu_n \geq \lambda > 0$

so clocks can't keep slowing down

Then clocks don't affect rec. v.s. trans.

$$\boxed{\text{Transient} \Leftrightarrow \sum_{n=1}^{\infty} \frac{\mu_1 \cdots \mu_{n-1}}{\lambda_1 \cdots \lambda_{n-1}} < \infty}$$

Null v.s. pos. recurrence?

In principle, fast clocks may improve return time! (Though they shouldn't since rates are all bounded away from 0)

So we'll just redo things and not just blindly use the discrete time case.

Inv. measure: $\pi A = 0$

$$\mu_{x+1}\pi(x+1) + \lambda_{x-1}\pi(x-1) - (\lambda_x + \mu_x)\pi(x) = 0$$

$$x \geq 1$$

$$\mu_1 \pi(1) - \lambda_0 \pi(0) = 0 \quad \text{so} \quad \pi(1) = \frac{\lambda_0}{\mu_1}\pi(0)$$

$$\mu_{x+1}\pi(x+1) - \mu_x \pi(x) = \lambda_x \pi(x) - \lambda_{x-1}\pi(x-1)$$

Sum $x=1$ to k

$$\mu_{k+1}\pi(k+1) - \mu_1 \pi(1) = \lambda_k \pi(k) - \lambda_0 \pi(0)$$

$=$!!

So $\pi(k+1) = \dfrac{\lambda_k}{\mu_{k+1}} \pi(k)$

$n \geq 1$: $\pi(n) = \dfrac{\lambda_{n-1} \cdots \lambda_1}{\mu_n \cdots \mu_2} \pi(1) = \dfrac{\lambda_0 \cdots \lambda_{n-1}}{\mu_1 \cdots \mu_n} \pi(0)$

So $\boxed{\text{pos. rec.} \Leftrightarrow \underbrace{\sum_{n=1}^{\infty} \frac{\lambda_0 \cdots \lambda_{n-1}}{\mu_1 \cdots \mu_n}}_{C} < \infty}$

$\pi(0)(1+C) = 1$ so $\pi(0) = \dfrac{1}{1+C}$

$\boxed{\pi(n) = \dfrac{\frac{\lambda_0 \cdots \lambda_{n-1}}{\mu_1 \cdots \mu_n}}{\sum_{k=0}^{\infty} \frac{\lambda_0 \cdots \lambda_{n-1}}{\mu_1 \cdots \mu_n}}}$

(recall: empty product is 1)

$(1-\alpha) \sum_{n=0}^{\infty} n \alpha^n$

Applications

Poisson(λ): $\mu_n = 0$, $\lambda_n = \lambda$ so $\sum \prod \frac{\mu's}{\lambda's} = 0 < \infty$

transient ✓

Queuing:

$M/M/1$: $\lambda_n = \lambda > 0 \; \forall n \geq 0$ $\mu_0 = 0$ $\mu_n = \mu > 0 \; \forall n \geq 1$

so $\mu < \lambda \Rightarrow \sum_{n=0}^{\infty} \left(\frac{\mu}{\lambda}\right)^n < \infty$ transient $\pi(n) = \left(1 - \frac{\lambda}{\mu}\right)\left(\frac{\lambda}{\mu}\right)^n$ $n \geq 0$

$\mu > \lambda \Rightarrow \sum_{n=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^n < \infty$ pos. rec. avg. queue length $\dfrac{\lambda/\mu}{1-\lambda/\mu} = \dfrac{\lambda}{\mu - \lambda}$

$\mu = \lambda \Rightarrow$ both give $\sum 1 = \infty$ so null rec.

$M/M/k$: $\sum_{n=0}^{\infty} \frac{\mu_1 \cdots \mu_n}{\lambda_1 \cdots \lambda_n} < \infty$ or $= \infty$?

enough to look at $\sum_{n=k}^{\infty} \frac{\mu_1 \cdots \mu_n}{\lambda_1 \cdots \lambda_n}$

if $n \geq k$ then $\frac{\mu_1 \cdots \mu_n}{\lambda_1 \cdots \lambda_n} = \frac{k! \mu^k k^{n-k} \mu^{n-k}}{\lambda^n}$

$$= \frac{k!}{k^k} \cdot \left(\frac{k\mu}{\lambda}\right)^n$$

So $\sum \frac{\mu_1 \cdots \mu_n}{\lambda_1 \cdots \lambda_n} < \infty \iff k\mu < \lambda$

$k\mu < \lambda \Rightarrow$ transient

$k\mu \geq \lambda \Rightarrow$ recurrent

Now: $\sum_{n=0}^{\infty} \frac{\lambda_0 \cdots \lambda_n}{\mu_1 \cdots \mu_n} < \infty$ or $= \infty$?

again, enough to look at $n \geq k$.

Then $\frac{\lambda_0 \cdots \lambda_n}{\mu_1 \cdots \mu_n} = \frac{\lambda^n}{k! \mu^k k^{n-k} \mu^{n-k}} = \frac{k^k}{k!} \left(\frac{\lambda}{k\mu}\right)^n$

so $k\mu > \lambda \Rightarrow$ pos. rec.

$k\mu = \lambda \Rightarrow$ null rec.

Basically, like having service at rate $k\mu$.