Motivating example for CTMC

Example :- n- Server queue system.

Service rate

$\mu \quad \mu \quad \mu \quad \mu$

$(S_1) \quad (S_2) \quad \cdots \quad (S_k) \quad \cdots \quad (S_n)$

$\uparrow$ λ (Poisson arrivals)

arrival rate.

Defn ①:- for $t \geq 0$ $N(t)$ is a Poisson process w/ rate $\lambda > 0$; then following are true

(i) $N(0) = 0$

(ii) $N(t)$ has independent increments

(iii) $N(\Delta) \sim \text{Poisson}(\Delta\lambda)$

This means no. of arrivals in Δ units of time span.

Additionally,

Any arrival finding all servers busy, leaves immediately without service.

Q) If an arrival finds all servers busy; find Expected no. of busy servers found by the next arriving customer?

Soln:-

T_K := expected no. of busy servers found by the nxt. arrival if there are currently K busy servers.

Generated by CamScanner

current time
(k busy servers)

time of nxt arrival

The fact that there are atleast $(n-k)$ idle servers can be ignored b/c of "memoryless" or (Markovian) property of exponential service times & exponential inter-arrival times.

this means that T_k (as defined earlier) is equivalent to :-

$T_k \equiv$ expected no. of busy servers found by nxt arrival for a k-server system when there are currently k busy servers.

Why?? B/c of this

Boundary condition $T_0 = 0$ is obviously evident.

Note that the "memorylessness" or "Markovian" property implies that time to nxt arrival $\sim exp(\lambda)$ and service time $\sim exp(\mu)$

So if we were to find T_1; by the definition of Expected value of a RV we would have.

$$T_1 = (1)\,\text{Prob}\begin{pmatrix}\text{nxt. arrival}\\ \text{finds one}\\ \text{busy server}\end{pmatrix} + (0)\,\text{Prob}\begin{pmatrix}\text{Nxt. arrival}\\ \text{finds}\\ 0\\ \text{busy}\\ \text{servers}\end{pmatrix}$$

$$= (1)\frac{\lambda}{\lambda+\mu} + (0)\frac{\mu}{\lambda+\mu} = \frac{\lambda}{\lambda+\mu}$$

why ?? B/c of following:-

Let $X \sim \exp(\lambda)$, $Y \sim \exp(\mu)$ $\Big\}$ independent RVs.

$$P(X < Y)$$

why $X < Y$??
B/c if nxt inter-arrival time, X is less than the service time of the 1 busy server; then nxt. arrival will find 1 busy server.

$$= \int_0^\infty \int_0^y f_{XY}(x,y)\,dx\,dy$$

$f_{xy}(x,y) = \lambda e^{-\lambda x}\mu e^{-\mu y}$ due to independence

$$\mu\lambda \int_0^\infty \int_0^y \left(e^{-\mu y} e^{-\lambda x}\right) dx\,dy$$

$$= \mu\lambda \int_0^\infty e^{-\mu y}\left(\frac{e^{-\lambda x}}{-\lambda}\right)_0^y dy = \mu \int_0^\infty \mu e^{-\mu y}\left(1 - e^{-\lambda y}\right) dy$$

$$= \dots = 1 - \frac{\mu}{\lambda+\mu} = \frac{\lambda}{\lambda+\mu}$$

Generated by CamScanner

Okay so $T_1 = \frac{\lambda}{\lambda+\mu}$.

Now, for the general case when there are K - busy servers; we can obtain the expression for T_K by conditioning upon what happens next.

i.e. w/ K busy servers → K exp(μ) alarm clocks.
→ 1 exp(λ) alarm clk.

and,
We will condition upon the following 2 disjoint events:-

(I) a service completion happens first.

OR

(II) an arrival happens first

We know,
Time till nxt. service completion ~ exp($K\mu$)

Why?? B/c for one of the K exp(μ) alarm clks. to "go off" we need to find the D^n of $\min(Y_1, Y_2, \ldots Y_K)$ where $Y_i \sim \exp(\mu)$.

This D^n turns out to be exp($K\mu$).

Why?? Do this yourself as it was already discussed in class.

Generated by CamScanner

So this enables us to find:-

$$\text{Prob}\begin{pmatrix}\text{nxt. thing to happen first} \\ \text{is a service completion}\end{pmatrix} = \frac{K\mu}{\lambda + K\mu}$$

$$\text{Prob}\begin{pmatrix}\text{nxt thing to happen first} \\ \text{is an arrival}\end{pmatrix} = \frac{\lambda}{\lambda + K\mu}$$

then using Law of total expectation:-

$$T_K = E\left(\begin{matrix}\text{no. of} \\ \text{busy} \\ \text{servers}\end{matrix} \middle| \begin{matrix}\text{nxt thing} \\ \text{to happen} \\ \text{first is a} \\ \text{service completion} \\ \text{\& there are} \\ \text{currently } K \text{ busy} \\ \text{servers}\end{matrix}\right) \times P\left(\begin{matrix}\text{nxt. thing} \\ \text{to happen} \\ \text{first is a} \\ \text{service} \\ \text{completion}\end{matrix} \middle| \begin{matrix}\text{currently} \\ K \text{ busy} \\ \text{servers}\end{matrix}\right)$$

$$+ E\left(\begin{matrix}\text{no. of} \\ \text{busy} \\ \text{servers}\end{matrix} \middle| \begin{matrix}\text{nxt. thing} \\ \text{to happen} \\ \text{first is an} \\ \text{arrival \&} \\ \text{there are} \\ \text{currently } K \text{ - busy} \\ \text{servers}\end{matrix}\right) \times P\left(\begin{matrix}\text{nxt thing} \\ \text{to happen} \\ \text{first is} \\ \text{an} \\ \text{arrival}\end{matrix} \middle| \begin{matrix}\text{currently} \\ K \text{ -} \\ \text{busy} \\ \text{servers}\end{matrix}\right)$$

$$= T_{K-1}\left(\frac{K\mu}{\lambda + K\mu}\right) + K\left(\frac{\lambda}{K\mu + \lambda}\right)$$

Why K? B/c in a K- server system, if an arrival happens before a service completion then he leaves w/o being served.

Plugging $k = 2, 3 \dots$

$$\therefore T_2 = T_1 \frac{2\mu}{2\mu+\lambda} + \frac{2\lambda}{2\mu+\lambda}$$

$$= \left(\frac{\lambda}{\lambda+\mu}\right)\left(\frac{2\mu}{2\mu+\lambda}\right) + \left(\frac{2\lambda}{2\mu+\lambda}\right)$$

$$T_3 = T_2\left(\frac{3\mu}{3\mu+\lambda}\right) + \frac{3\lambda}{3\mu+\lambda}$$

$$= \left(\frac{\lambda}{\lambda+\mu}\right)\left(\frac{2\mu}{2\mu+\lambda}\right)\left(\frac{3\mu}{3\mu+\lambda}\right) + \left(\frac{2\lambda}{2\mu+\lambda}\right)\left(\frac{3\mu}{3\mu+\lambda}\right) + \frac{3\lambda}{3\mu+\lambda}$$

And in general,

$$T_n = \frac{n\lambda}{n\mu+\lambda} + \sum_{i=1}^{n-1}\left(\frac{i\lambda}{i\mu+\lambda}\right)\prod_{j=i+1}^{n}\frac{j\mu}{j\mu+\lambda}$$

#

Now we will formally introduce CTMC!

The above is an example of a continuous time stochastic process & actually corresponds to CTMC if we think of the rates

$$q_{i,i+1} = \lambda_i = \lambda$$
$$q_{i,i-1} = \mu_i = \mu$$

And the states space $S = \{0, 1, 2, \dots n\}$ as no. of people in the system (or no. of busy servers)

Generated by CamScanner