Queueing Theory and Simulation, lecture 2

Nicky van Foreest

June 22, 2021

Contents

1 Simulations

1.1 Discrete time, Section 2.1

$$d_k = \min\{L_{k-1}, c_k\} \tag{1}$$
$$L_k = L_{k-1} + a_k - d_k \tag{2}$$

- Time chopped up in periods.
- Often $a_k \sim \text{Poi}(\lambda)$, c.f. Section 2.2.
- Handy to compute queue length behavior over time
- How to estimate waiting times? See exercise in the book. Not entirely simple

1.2 Continuous time, Section 2.3

$$X_k = \text{interarrival time between job } k-1 \text{ and job } k, \tag{3}$$
$$A_k = A_{k-1} + X_k \tag{4}$$
$$D_k = \min\{A_k, D_{k-1}\} + S_k, \tag{5}$$
$$J_k = D_k - A_k, \quad \text{sojourn time} \tag{6}$$
$$W_k = J_k - S_k, \quad \text{sojourn time} \tag{7}$$

- Also a very elegant recursive procedure.
- Now the waiting is easy to find, but not the queue length (try it is you like)
- Simple to use simulation to estimate fraction of jobs that waiting longer than 10 (or whatever number you like).

1.3 Another continuous construction, 2.3

$$\begin{aligned} W_k &= \max\{W_{k-1} + S_{k-1} - X_k, 0\} && (8) \\ &= [W_{k-1} + S_{k-1} - X_k]^+, && (9) \\ J_k &= W_k + S_k && (10) \\ D_k &= A_k + J_k && (11) \end{aligned}$$

1.4 Multiserver queue

- Job k sees, upon arrival, a waiting time $w_{k,i}$ at queue i.
- I represents here a vector $(1, 1, \dots, 1)$.

$$\begin{aligned} s_k &= \arg\min_i \{w_{k,i}\} && (12) \\ w_{k+1} &= [w_k + S_k e_{s_k} - X_{k+1} I]^+. && (13) \end{aligned}$$

2 Exponential distribution, 2.4

2.1 Origin

- A customer goes to a shop, every day uniformly distributed between 12 and 13.
- Superimpose many, N say, of such customers that go to a shop, every day uniformly distributed between 12 and 13.
- The interarrival times of these customers (as seen by the shop) has mean $1/N$.
- Very quickly (for $N > 10$ or so), the interarrival times appear to be $\sim \text{Exp}(\lambda)$.

2.2 Relation to Poisson

Crucial ideas:

- Random variables that are exponentially distributed have the memoryless property, look it up in the book
- $X_k \sim \text{Exp}(\lambda) \iff N[0, t] \sim \text{Poi}(\lambda t)$.

3 My programming environment, only for the interested

3.1 emacs

3.2 literate programming

3.3 org mode