Lecture 21: Burke's Theorem and Jackson Networks

Concepts checklist

At the end of this lecture, you should be able to:

- *state* Burke's Theorem and *explain its importance*, and *use it* to analyse equilibrium behaviour of particular queueing systems;
- *define* an Open Jackson Network; and,
- *state* Jackson's Theorem.

Theorem 19 (Burke's Theorem.). *Consider a queue with a Poisson arrival process of rate λ and exponential service time distribution with parameter $\mu > \lambda$. In equilibrium,*

(*i*) *the departure process from this queue is a Poisson process with parameter λ,*

(*ii*) *the number in the queue at any time t is independent of the departure process prior to t.*

Proof. (*i*) Recall that the queue-length process of a birth-and-death process is reversible. This implies that the reverse process is a continuous-time Markov chain with the same transition rates as the forward time process.

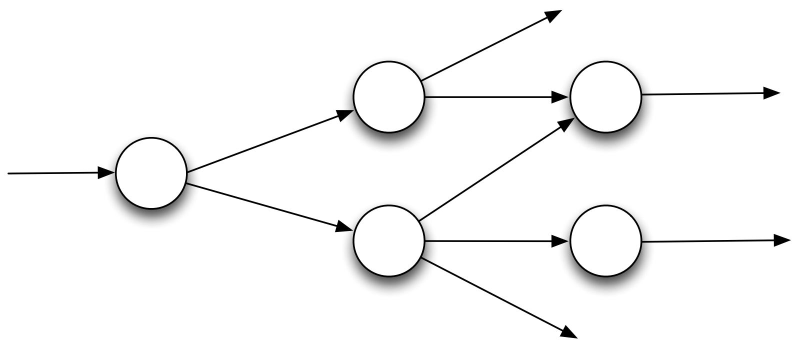
In forward time, arrivals occur in a Poisson process; since the reversed-time Markov chain has the same transition rates, the reversed-time "arrival process" is also a Poisson process. Hence, we have that the forward time departure process is also a Poisson process.

(*ii*) Furthermore, in forward time the state of the queue at time t is independent of future arrivals, which implies that the state is also independent of the past departure process in the reversed-time Markov chain. Because the process is reversible, it is also true for the forward time process that the state is independent of past departure process. □

Note: it is not surprising that the departure rate is λ — because what enters must leave *in equilibrium* for a continuous-time Markov chain to be stable — but what is surprising is the fact that the departure process *in equilibrium* is Poisson. We might have expected a more complicated description of the departure process. (For example, that it is Poisson of rate μ (the service rate) when the queue is busy and of rate 0 when the queue is empty.)

Burke's Theorem is important because it

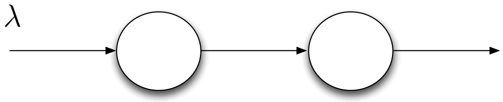
- allows us to split up the output of one queue and feed it to other queues, where the departure process from one queue is the arrival process to other queues,
- and tells us that
  1. the arrival process of downstream queues will be Poisson,
  2. the state of downstream queues at time t
     - depends on the departure process of upstream queues before time t, but
     - is independent of the state of the upstream queues at time t.



Essentially, this means that if a network of n-server queues can be ordered from 1 to J in such a way that customers leaving queue $j \in \{1, 2, \ldots, J\}$ are fed into queues $j+1, \ldots, J$ or leave the system, then this network has a product form equilibrium distribution

$$\pi(\boldsymbol{n}) = \pi(n_1, n_2, \ldots, n_J) = \pi_1(n_1)\pi_2(n_2)\ldots\pi_J(n_J).$$

Example 15. Tandem of single-server queues (Feed-Forward)



Let $\boldsymbol{\pi}(n_1, n_2)$ be the equilibrium distribution, where n_i is the level of occupancy of the ith single-server queue in the tandem.

By Burke's Theorem, the states n_1 and n_2 are independent, so $\boldsymbol{\pi}(n_1, n_2) = \pi_1(n_1)\pi_2(n_2)$, where $\pi_i(n_i)$ is the equilibrium distribution of the ith single-server queue.

Since $\pi_i(n_i) = \left(1 - \frac{\lambda}{\mu_i}\right)\left(\frac{\lambda}{\mu_i}\right)^{n_i}$ for $\lambda < \mu_i$ and $i \in \{1, 2\}$,

$$\pi(n_1, n_2) = \pi_1(n_1)\pi_2(n_2) = \left(1 - \frac{\lambda}{\mu_1}\right)\left(\frac{\lambda}{\mu_1}\right)^{n_1}\left(1 - \frac{\lambda}{\mu_2}\right)\left(\frac{\lambda}{\mu_2}\right)^{n_2}$$

iff $\lambda < \min\{\mu_1, \mu_2\}$.

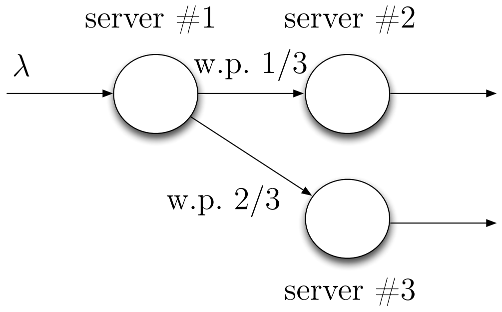
Example 16. Three single-server queues (Feed-forward)

Here, the departure process from queue 1 is probabilistically split: 1/3 to queue 2 and 2/3 to queue 3.

Using Burke's Theorem again, we have

$$\begin{aligned}\pi(\boldsymbol{n}) &= \pi(n_1, n_2, n_3) = \pi_1(n_1)\pi_2(n_2)\pi_3(n_3),\\ &= \left(1 - \frac{\lambda}{\mu_1}\right)\left(\frac{\lambda}{\mu_1}\right)^{n_1}\left(1 - \frac{\frac{1}{3}\lambda}{\mu_2}\right)\left(\frac{\frac{1}{3}\lambda}{\mu_2}\right)^{n_2}\left(1 - \frac{\frac{2}{3}\lambda}{\mu_3}\right)\left(\frac{\frac{2}{3}\lambda}{\mu_3}\right)^{n_3}\end{aligned}$$

iff $\lambda < \min\left\{\mu_1, 3\mu_2, \frac{3\mu_3}{2}\right\}$.



We have used Burke's Theorem to show that feed forward queueing networks have a product form equilibrium distribution. We can extend this to networks which have feedback. When output streams are fed back into earlier queues, the input streams to queues are generally non-Poisson and the logic we have used collapses. However the **independence** result survives, as we will see.

The breakthrough dealing with feedback networks has been generally credited to Jackson in 1957, who considered an open network of N s_i-server queues for $1 \leq i \leq N$. He proved that the joint equilibrium distribution for the network is a product over the queues of the equilibrium distributions of the individual queues.

We will call such networks Open Jackson Networks. Jackson generalised this idea further by allowing the arrival rate at the ith queue to be an arbitrary function $\lambda_i(\boldsymbol{n})$ of the total number of customers in the network.

Definition 19 (Open Jackson Network.)**.** *An Open Jackson Network consists of a network of N queues, where at node i of that network,*

- *arrivals come from outside of the network at rate λ_i,*
- *the service rate is $\mu_i(n_i)$ when there are n_i customers in that queue, and*
- *a customer upon completing service will either*
  - *move to queue j with probability γ_{ij}, or*
  - *leave the network with probability $\beta_i = 1 - \sum_j \gamma_{ij}$.*

The state space of the network records the number of customers at a queue but does not distinguish between customers when a service period ends and a customer is removed from the queue.

Theorem 20 (Jackson's Theorem.)**.** *An Open Jackson Network has the following product form equilibrium distribution (provided it can be normalised):*

$$\pi(\boldsymbol{n}) = \pi(n_1, n_2, \ldots, n_N) = \prod_{i=1}^{N} \pi_i(n_i),$$

$$\textit{where} \quad \pi_i(n_i) = \pi_i(0) \prod_{\ell=1}^{n_i} \frac{y_i}{\mu_i(\ell)} \text{ is the equilibrium of the } i\text{th queue}$$

and y_i is the average arrival rate to queue i, given by the traffic equations

$$y_i = \lambda_i + \sum_{j=1}^{N} y_j \gamma_{ji}.$$

Note:

- Normalisation depends on whether the constants $\pi_i(0)$ can be found for each $i \in \{1, 2, \ldots, N\}$ such that

$$\sum_{n_i=0}^{\infty} \pi_i(n_i) = 1.$$

- We tacitly assume that the rate into each queue must be the same as the rate out (this implies stability). That is, y_i is both the arrival rate and departure rate from queue i.

The form

$$Q_i(n_i) = \prod_{\ell=1}^{n_i} \frac{y_i}{\mu_i(\ell)}$$

is an invariant measure for the number of customers at queue i if the queue is fed with a Poisson arrival stream of rate y_i.

In fact, y_i is the total *average* arrival rate to queue i but it is, in general, not a Poisson stream and yet the result is as if it is a Poisson stream.