

Queuing Networks

Florence Perronnin

Polytech'Grenoble - UGA

March 23, 2017

Outline

- 1 Introduction to Queuing Networks
- 2 Refresher: M/M/1 queue
- 3 Reversibility
- 4 Open Queueing Networks
- 5 Closed queueing networks
- 6 Multiclass networks
- 7 Other product-form networks

Introduction to Queuing Networks

- Single queues have simple results
- They are quite robust to slight model variations
- We may have multiple contention resources to model:
 - ▶ servers
 - ▶ communication links
 - ▶ databases

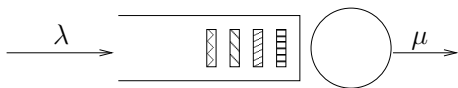
with various routing structures.

Queuing networks are direct results for interaction of classical single queues with probabilistic or static routing.

Outline

- 1 Introduction to Queuing Networks
- 2 Refresher: M/M/1 queue**
- 3 Reversibility
- 4 Open Queueing Networks
- 5 Closed queueing networks
- 6 Multiclass networks
- 7 Other product-form networks

Refresher: M/M/1

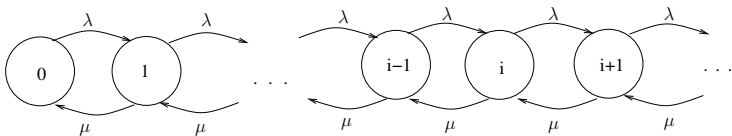


M/M/1 queue

- Infinite capacity
- Poisson(λ) arrivals
- Exp(μ) service times
- FIFO discipline

Definition

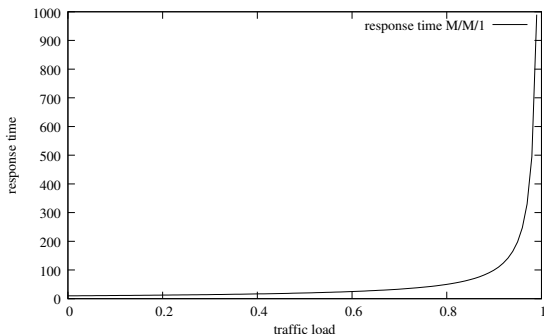
$\rho = \frac{\lambda}{\mu}$ is the **traffic intensity** of the queueing system.



Number of clients $X(t)$ in the system follows a **birth and death** process.

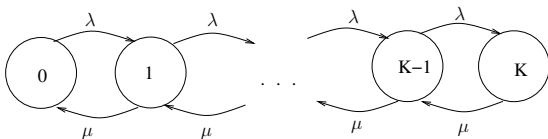
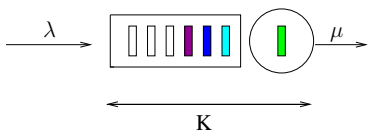
Results for M/M/1 queue

- 1 Stable if and only if $\rho < 1$
- 2 Clients follow a geometric distribution $\forall i \in \mathbb{N}, \pi_i = (1 - \rho)\rho^i$
- 3 Mean number of clients $\mathbb{E}[X] = \frac{\rho}{1-\rho}$
- 4 Average response time $\mathbb{E}[T] = \frac{1}{\mu-\lambda}$



M/M/1/K

In reality, buffers are finite: M/M/1/K is a queueing system with **blocking**.

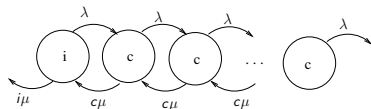
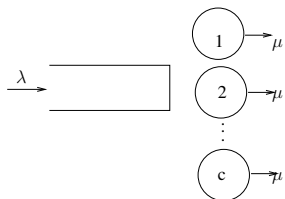


Results for M/M/1/K queue

Geometric distribution with finite state space

$$\pi(i) = \frac{(1 - \rho)\rho^i}{1 - \rho^{K+1}}$$

M/M/c



Results for M/M/c queue

Stability condition $\lambda < c\mu$.

where $\rho = \frac{\lambda}{\mu}$ and with

$$\pi(i) = \begin{cases} C_{\rho,c} \frac{\rho^i}{i!} & \text{if } i \leq c \\ C_{\rho,c} \frac{1}{c!} \left(\frac{\rho}{c}\right)^i & \text{if } i > c \end{cases}$$

$$C_{\rho,c} = \frac{1}{\sum_{i=0}^{c-1} \frac{\rho^i}{i!} + \frac{\rho^c}{c!} \frac{1}{1-\rho/c}}$$

Outline

- 1 Introduction to Queuing Networks
- 2 Refresher: M/M/1 queue
- 3 Reversibility**
- 4 Open Queueing Networks
- 5 Closed queueing networks
- 6 Multiclass networks
- 7 Other product-form networks

Reversibility

Definition

A continuous-time Markov chain $X(t)$ is time-reversible if and only if there exist a probability distribution π such that $\forall (i, j) \in \mathcal{E}^2, \pi_i q_{ij} = \pi_j q_{ji}$

Theorem

the π distribution is also the stationary distribution.

Proof

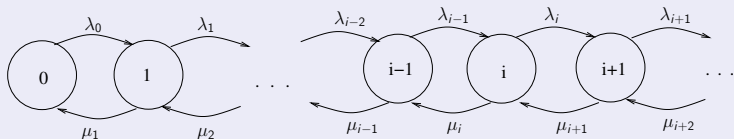
Check the balance equations.

Reversibility

Proposition

An ergodic birth and death process is time-reversible.

Proof



By induction:

- ① $\pi_0 \lambda_0 = \pi_1 \mu_1$
- ② Suppose $\pi_{i-1} \lambda_{i-1} = \pi_i \mu_i$. Then
 $\pi_i (\lambda_i + \mu_i) = \pi_{i+1} \mu_{i+1} + \pi_{i-1} \lambda_{i-1}$
 Which gives $\pi_i \lambda_i = \pi_{i+1} \mu_{i+1}$.

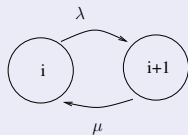
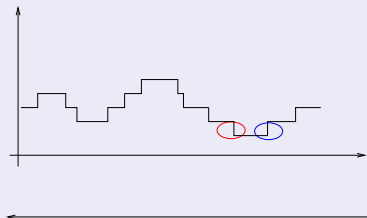


Burke's theorem

Theorem

The output process of an $M/M/s$ queue is a Poisson process that is *independent* of the number of customers in the queue.

Sketch of Proof.

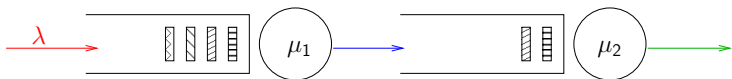


$X(t)$ increases by 1 at rate $\lambda\pi_i$ (Poisson process λ). Reverse process increases by 1 at rate $\mu\pi_{i+1} = \lambda\pi_i$ by reversibility. □

Outline

- 1 Introduction to Queuing Networks
- 2 Refresher: M/M/1 queue
- 3 Reversibility
- 4 Open Queueing Networks**
 - Tandem queues
 - Acyclic networks
 - Backfeeding
 - Jackson networks
 - Open networks of M/M/c queues
- 5 Closed queueing networks
- 6 Multiclass networks

Tandem queues



Let X_1 and X_2 denote the number of clients in queues 1 and 2 respectively.

Lemma

X_1 and X_2 are independent rv's.

Proof

Arrival process at queue 1 is Poisson(λ) so future arrivals are independent of $X_1(t)$.

By time reversibility $X_1(t)$ is independent of past departures.

Since these departures are the arrival process of queue 2, $X_1(t)$ and $X_2(t)$ are independent. \square

Tandem queues

Theorem

The number of clients at server 1 and 2 are independent and

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

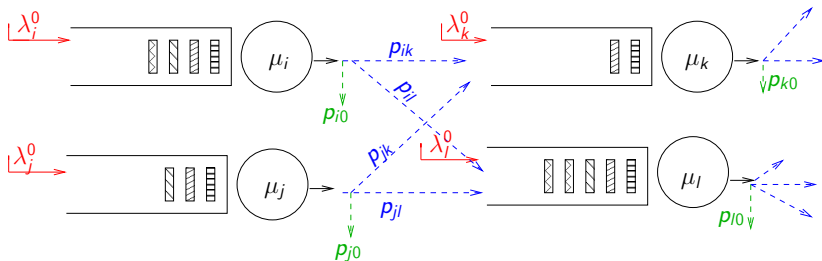
Proof

By independence of X_1 and X_2 the joint probability is the product of M/M/1 distributions. □

This result is called a **product-form** result for the tandem queue.
This product form also appears in more general **networks** of queues.

Acyclic networks

Example of a feed-forward network:



- Exponential service times
- output of i is routed to j with probability p_{ij}
- external traffic arrives at i with rate λ_i^0
- packets exiting queue i leave the system with probability p_{i0} .

Routing matrix:

$$R = \begin{pmatrix} 0 & p_{ij} & p_{ik} & p_{il} \\ p_{ji} & 0 & p_{jk} & p_{jl} \\ p_{ki} & p_{kj} & 0 & p_{kl} \\ p_{li} & p_{lj} & p_{lk} & 0 \end{pmatrix}$$

Decomposition of a Poisson Process

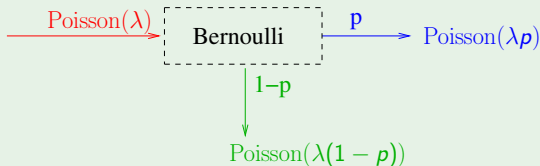
Problem

- $N(t)$ Poisson process with rate λ
- $Z(n)$ sequence of iid rv's $\sim \text{Bernoulli}(p)$ independent of N .

Suppose the n th trial is performed at the n th arrival of the Poisson process.

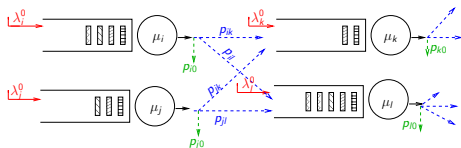
Result

- Resulting process $M(t)$ of successes is $\text{Poisson}(\lambda p)$.
- Process of failures $L(t)$ is $\text{Poisson}(\lambda(1-p))$ and is independent of $M(t)$.



Acyclic networks

Total arrival rate at node i : λ_i
 ($1 \leq i \leq K$).



No feedback: using Burke theorem, all internal flows are Poisson!
 Thus we can consider K independent M/M/1 queues with Poisson arrivals with rate λ_i , where

$$\lambda_i = \lambda_i^0 + \sum_{j=0}^K \lambda_j p_{ji} \quad \text{i.e. } \vec{\lambda} = \vec{\lambda}^0 + \vec{\lambda} \mathbf{R} \quad \text{in matrix notation.}$$

Stability condition

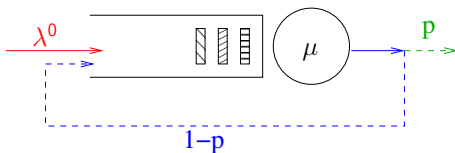
All queues must be stable independently: $\lambda_i < \mu_i, \quad \forall i = 1, 2, \dots, K.$

Backfeeding

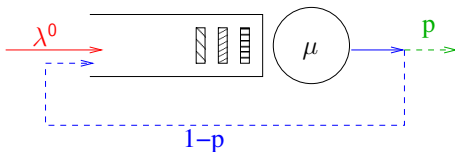
Example

Switch transmitting frames with random errors. A NACK is sent instantaneously if the frame is incorrect.

- Frame success probability is p .
- Arrivals $\sim \text{Poisson}(\lambda^0)$
- Frame transmission times $\sim \text{Exp}(\mu)$



Backfeeding



Remark

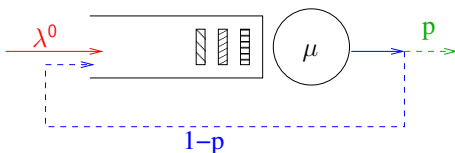
Arrivals are not Poisson anymore!

Result

The departure process is still Poisson with rate λp .

Proof in [Walrand, An Introduction to Queueing Networks, 1988].

Backfeeding: steady-state



Balance equations:

$$\pi(0)\lambda^0 = \mu p \pi(1)$$

$$\pi(n)(\lambda^0 + p\mu) = \lambda^0 \pi(n-1) + \mu p \pi(n+1), \quad n > 0$$

Actual arrival rate $\lambda = \lambda^0 + (1-p)\lambda$, so $\lambda^0 = \lambda p$ which gives

$$\pi(0)\lambda = \mu \pi(1)$$

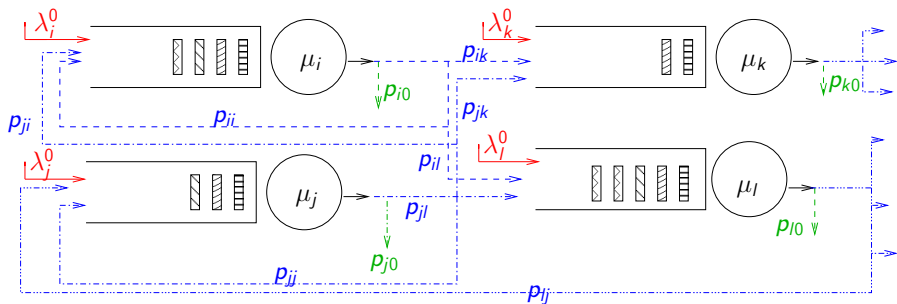
$$\pi(n)(\lambda + \mu) = \lambda \pi(n-1) + \mu \pi(n+1), \quad n > 0 \quad \text{M/M/1!}$$

The unique solution is:

$$\pi(n) = \left(1 - \frac{\lambda}{\mu}\right) \left(\frac{\lambda}{\mu}\right)^n = \left(1 - \frac{\lambda^0}{p\mu}\right) \left(\frac{\lambda^0}{p\mu}\right)^n$$



Jackson networks: example



Backfeeding allowed.

Jackson networks

Theorem (Jackson, 1957)

If $\lambda_i < \mu_i$ (stability condition), $\forall i = 1, 2, \dots, K$ then

$$\pi(\vec{n}) = \prod_{i=1}^K \left(1 - \frac{\lambda_i}{\mu_i}\right) \left(\frac{\lambda_i}{\mu_i}\right)^{n_i} \quad \forall \vec{n} = (n_1, \dots, n_K) \in \mathbb{N}^K.$$

where $\lambda_1, \dots, \lambda_K$ are the *unique* solution of the system

$$\lambda_i = \lambda_i^0 + \sum_{j=0}^K \lambda_j p_{ji}$$

Product form even with backfeeding!

Jackson networks: sketch of proof

Derive balance equations:

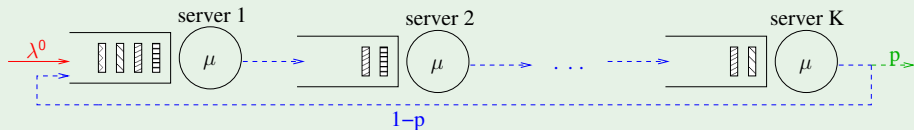
$$\begin{aligned}
 \pi(\vec{n}) \left(\sum_{i=1}^K \lambda_i^0 + \sum_{i=1}^K \mathbb{1}_{[n_i > 0]} \mu_i \right) &= \sum_{i=1}^K \mathbb{1}_{[n_i > 0]} \lambda_i^0 \pi(\vec{n} - \vec{e}_i) \\
 &+ \sum_{i=1}^K p_{i0} \mu_i \pi(\vec{n} + \vec{e}_i) \\
 &+ \sum_{i=1}^K \sum_{j=1}^K \mathbb{1}_{[n_j > 0]} p_{ij} \mu_j \pi(\vec{n} + \vec{e}_i - \vec{e}_j)
 \end{aligned}$$

Then check that $\pi(\vec{n}) = \prod_{i=1}^K \left(1 - \frac{\lambda_i}{\mu_i}\right) \left(\frac{\lambda_i}{\mu_i}\right)^{n_i}$ satisfies the balance equations with $\lambda_i = \lambda_i^0 + \sum_{j=0}^K \lambda_j p_{ji}$. □

Jackson networks: example

Example

Switches transmitting frames with random errors.



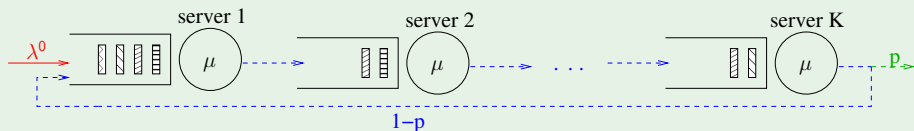
Traffic equations give $\lambda_i = \lambda_{i-1}$ for $i \geq 2$ and $\lambda_1 = \lambda^0 + (1-p)\lambda_K$. The unique solution is clearly $\lambda_i = \frac{\lambda^0}{p}$ for $1 \leq i \leq K$. Apply Jackson's theorem:

$$\pi(\vec{n}) = \left(1 - \frac{\lambda^0}{p\mu}\right)^K \left(\frac{\lambda^0}{p\mu}\right)^{n_1 + \dots + n_K} \quad \forall \vec{n} = (n_1, \dots, n_K) \in \mathbb{N}^K.$$

Jackson networks: example (Continued)

Example

Switches transmitting frames with random errors.



Using M/M/1 results for each queue we get the mean number of frames at each queue $\mathbb{E}[X_i] = \frac{\lambda^0}{\rho\mu - \lambda^0}$

The expected transmission time of a frame is therefore (Little)

$$\mathbb{E}[T] = \frac{1}{\lambda^0} \mathbb{E}[X] = \frac{1}{\lambda^0} \sum_{i=1}^K \mathbb{E}[X_i] = \frac{K}{\rho\mu - \lambda^0}$$

Networks of M/M/c queues

Theorem

Consider an open network of K M/M/ c_i queues. Let $\mu_i(n) = \mu_i \min(n, c_i)$ and $\rho_i = \frac{\lambda_i}{\mu_i}$.

Then **if** $\rho_i < c_i$ for all $1 \leq i \leq K$ then

$$\pi(\vec{n}) = \prod_{i=1}^K C_i \left(\frac{\lambda_i^{n_i}}{\prod_{m=1}^{n_i} \mu_i(m)} \right) \quad \forall \vec{n} = (n_1, \dots, n_K) \in \mathbb{N}^K$$

where $(\lambda_1, \dots, \lambda_K)$ is the **unique positive solution** of the traffic equations

$$\lambda_i = \lambda_i^0 + \sum_{j=0}^K \lambda_j p_{ji}, \quad \text{and where } C_i = \left(\sum_{m=1}^{c_i-1} \frac{\rho_i^m}{m!} + \frac{\rho_i^{c_i}}{c_i!(1 - \rho_i/c_i)} \right)^{-1}$$

Outline

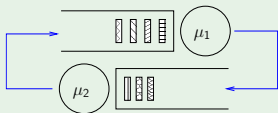
- 1 Introduction to Queuing Networks
- 2 Refresher: M/M/1 queue
- 3 Reversibility
- 4 Open Queueing Networks
- 5 Closed queueing networks**
- 6 Multiclass networks
- 7 Other product-form networks

Closed Queueing Networks

Definition

A **closed** system is a system in which the number of clients is a **constant** variable.

Example



The **traffic equations are linearly dependent!**

$$\lambda_i = \sum_{j=0}^K \lambda_j p_{ji}, \quad 1 \leq i \leq K$$

Therefore the **previous Jackson theorem cannot be applied** and does not yield the correct result.

Jackson theorem for closed networks

Consider a closed queueing system with K queues and N clients. Define by $S(N, K)$ the set of vectors $\vec{n} = (n_1, \dots, n_K) \in \mathbb{N}^K$ such that $n_1 + \dots + n_K = N$.

Theorem (Closed Jackson networks)

Let $(\lambda_1, \dots, \lambda_K)$ be an *arbitrary non-zero* solution of the traffic equation $\lambda_i = \sum_{j=0}^K \lambda_j p_{ji}$, $1 \leq i \leq K$. Then for all $\vec{n} \in S(N, K)$,

$$\pi(\vec{n}) = \frac{1}{C(N, K)} \prod_{i=1}^K \left(\frac{\lambda_i}{\mu_i} \right)^{n_i}, \quad C(N, K) = \sum_{\vec{n} \in S(N, K)} \prod_{i=1}^K \left(\frac{\lambda_i}{\mu_i} \right)^{n_i}$$

Not a product-form!

Performance indexes

A typical performance metric is the **expected queue length** $\mathbb{E}[X_i]$ at node i .

Expected queue length

$$\mathbb{E}[X_i] = \sum_{k=1}^N \left(\frac{\lambda_i}{\mu_i} \right)^k \frac{C(N-k, K)}{C(N, K)}$$

Sketch of Proof:

For a \mathbb{N} -valued r.v we have: $\mathbb{E}[X_i] = \sum_{k \geq 1} P(X_i \geq k)$

$$\begin{aligned} P(X_i \geq k) &= \sum_{\vec{n} \in S(N, K), n_i \geq k} \pi(\vec{n}) = \sum_{\vec{n} \in S(N, K), n_i \geq k} \dots \\ &= \left(\frac{\lambda_i}{\mu_i} \right)^k \frac{C(N-k, K)}{C(N, K)} \end{aligned} \quad (1)$$

Performance indexes

One may also be interested in the **utilization** at node i , i.e. the probability that node i is non-empty.

Utilization

$$U_i = 1 - P(X_i = 0) = \frac{\lambda_i}{\mu_i} \frac{C(N - K, K)}{C(N, K)}$$

Proof

Note that $P(X_i = k) = P(X_i \geq k) - P(X_i \geq k - 1)$ and apply (1) with $k = 0$.

Convolution algorithm

Computing the normalization factor $C(N, K)$ is a heavy task!

$$\begin{aligned}
 C(n, k) &= \sum_{\vec{n} \in S(n, k)} \prod_{i=1}^k \left(\frac{\lambda_i}{\mu_i} \right)^{n_i} = \sum_{m=0}^n \sum_{\substack{\vec{n} \in S(n, k) \\ n_k = m}} \prod_{i=1}^k \left(\frac{\lambda_i}{\mu_i} \right)^{n_i} \\
 &= \sum_{m=0}^n \left(\frac{\lambda_k}{\mu_k} \right)^m \sum_{\vec{n} \in S(n-m, k-1)} \prod_{i=1}^{k-1} \left(\frac{\lambda_i}{\mu_i} \right)^{n_i}
 \end{aligned}$$

Convolution algorithm (Buzen, 1973)

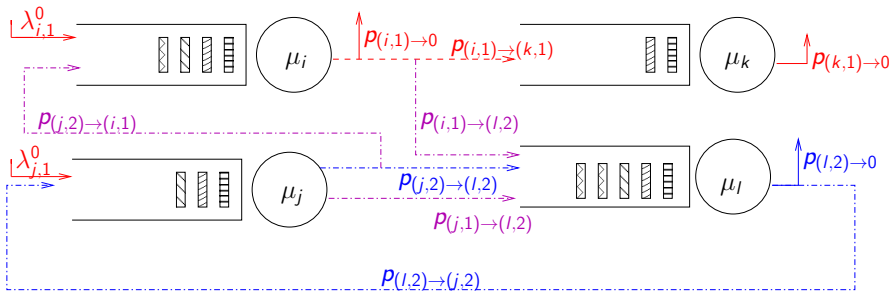
$$C(n, k) = \sum_{m=0}^n \left(\frac{\lambda_k}{\mu_k} \right)^m C(n-m, k-1) \quad \text{and} \quad \begin{cases} C(n, 1) = \left(\frac{\lambda_1}{\mu_1} \right)^n \\ C(0, k) = 1, \forall 1 \leq k \leq K \end{cases}$$

Outline

- 1 Introduction to Queuing Networks
- 2 Refresher: M/M/1 queue
- 3 Reversibility
- 4 Open Queueing Networks
- 5 Closed queueing networks
- 6 Multiclass networks**
- 7 Other product-form networks

Multiclass Networks

Definition



Multiclass Networks

Definition

- $K < \infty$ nodes and $R < \infty$ classes
- Customer at node i in class r will go to node j with class s with probability $p_{(i,r):(j,s)}$
- (i, r) and (j, s) belong to the same **subchain** if $p_{(i,r):(j,s)} > 0$
- FIFO discipline and exponential service times

Definition

A subchain is **open** iff there exist one pair (i, r) for which $\lambda_{(i,r)}^0 > 0$.

Definition

A **mixed** system contains at least one open subchain and one closed subchain.

Multiclass Networks

Definition

The state of a multiclass network may be characterized by the number of customers of each class at each node

$$\vec{Q}(t) = (\vec{Q}_1(t), \vec{Q}_2(t), \dots, \vec{Q}_K(t)) \text{ with } \vec{Q}_i(t) = (Q_{i1}(t), \dots, Q_{iR}(t))$$

Problem

$\vec{Q}(t)$ is not a CMTC!

To see why, consider the FIFO discipline: how do you know the class of the next customer?

Multiclass Networks

Definition

Define $\vec{X}_i(t) = (l_{i1}(t), \dots, l_{iQ_i(t)}(t))$ with $l_{ij}(t)$ the **class of the j th customer at node i** .

Proposition

$\vec{X}(t)$ is a CMTC!

Solving the balance equations for X gives a **product-form** solution. The steady-state distribution of $\vec{X}(t)$ also gives the distribution of $\vec{Q}(t)$ by aggregation of states.

Outline

1 Introduction to Queuing Networks

2 Refresher: M/M/1 queue

3 Reversibility

4 Open Queueing Networks

5 Closed queueing networks

6 Multiclass networks

7 Other product-form networks

- Other service disciplines
- BCMP networks

Other queueing networks

Limitations of Jackson networks

Jackson networks imply

- FIFO discipline
- probabilistic routing

These assumptions can be relaxed using BCMP and Kelly networks.

BCMP networks

[Baskett, Chandy, Muntz and Palacios 1975]

Definition

BCMP networks are multiclass networks with exponential service times and c_i servers at node i .

Service disciplines may be:

- FCFS
- Processor Sharing
- Infinite Server
- LCFS

BCMP networks also have **product-form** solution!

BCMP networks

Definitions

Consider an open/closed/mixed BCMP network with K nodes and R classes in which each node is either FIFO, PS, LIFO or IS. Define

- $\rho_{ir} = \frac{\lambda_{ir}}{\mu_{ir}}$ for LIFO, IS and PS nodes
- $\rho_{ir} = \frac{\lambda_{ir}}{\mu_i}$ for FIFO nodes
- $\lambda_{ir} = \lambda_{ir}^0 + \sum_{(j,s) \in E_k} \lambda_{js} p_{(i,r);(j,s)}$ for any (i,r) of each open subchain E_k
- $\lambda_{ir} = \sum_{(j,s) \in E_m} \lambda_{js} p_{(i,r);(j,s)}$ for any (i,r) of each closed subchain E_m

BCMP networks

Main result

Theorem

The steady-state distribution is given by: for all \vec{n} in state space \mathcal{S} ,

$$\pi(\vec{n}) = \frac{1}{G} \prod_{i=1}^K f_i(\vec{n}_i) \quad \text{with } G = \sum_{\vec{n} \in \mathcal{S}} \prod_{i=1}^K f_i(\vec{n}_i)$$

with $\vec{n} = (\vec{n}_1, \dots, \vec{n}_K) \in \mathcal{S}$ and $\vec{n}_i = (n_{i1}, \dots, n_{iR})$, **if and only if** (stability condition for open subchains)
$$\sum_{r:(i,r) \in \text{any open } E_k} \rho_{ir} < 1, \quad \forall 1 \leq i \leq K.$$

Moreover, $f_i(\vec{n}_i)$ has an explicit expression for each service discipline.

BCMP networks

Main result

$$\text{FIFO } f_i(\vec{n}_i) = |n_i|! \prod_{j=1}^{|n_i|} \frac{1}{\alpha_j(j)} \prod_{r=1}^R \frac{\rho_{ir}^{n_{ir}}}{n_{ir}!} \text{ with } \alpha_j(j) = \min(c_i, j).$$

$$\text{PS or LIFO } f_i(\vec{n}_i) = |n_i|! \prod_{r=1}^R \frac{\rho_{ir}^{n_{ir}}}{n_{ir}!}$$

$$\text{IS } f_i(\vec{n}_i) = \prod_{r=1}^R \frac{\rho_{ir}^{n_{ir}}}{n_{ir}!}$$

Extensions

the BCMP product form result may be extended to the following cases:

- state-dependent routing probabilities
- arrivals depending on the number of customers in the corresponding subchain