Multi-class Queues and Stochastic Networks

Detailed content:
1. reversibility, stationarity, basic queues, output theorem, feedforward networks
2. partial balance, Jackson network, Kelly-Whittle network, arrival theorem
3. quasi-reversibility, customer types, BCMP networks, bandwidth sharing networks
4. blocking, aggregation, decomposition
5. loss networks, bandwidth sharing networks, insensitivity via supplementary variables
6. sojourn time distribution in networks
7. MVA, AMVA, QNA
8. fluid queues, basic models
9. feedback fluid queues, networks of fluid queues
Last time on MQSN ….

• Kelly Whittle network
• Partial balance
Internet of Things

- Optimal routing in Jackson network

- Jobs arrive at outside nodes with given destination
- Each node single server queue
- Minimize sojourn time
- Optimal route selection
- Alternative route
Multi-class Queues and Stochastic Networks

Today (lecture 3): queue length based on Kelly’s lemma

Nelson, sec 10.4—10.7, Kelly, chapter 3

- Kelly Whittle network
- Alternative proof
- Quasi reversibility
- Queue disciplines
- Network of quasi reversible queues
- BCMP networks
Kelly Whittle network

\[ q(n, n-e_j + e_k) = \frac{\psi(n-e_j)}{\phi(n)} \mu_j p_{jk} \]

\[ q(n, n-e_j) = \frac{\psi(n-e_j)}{\phi(n)} \mu_j p_{j0} \]

\[ q(n, n+e_k) = \frac{\psi(n)}{\phi(n)} \lambda_k \]

Theorem: The equilibrium distribution for the Kelly Whittle network is

\[ \pi(n) = B \phi(n) \prod_{j=1}^{J} \rho_j^{n_j} \quad \rho_j = \gamma_j / \mu_j \quad n \in S \]

where

\[ \gamma_j = \lambda_j + \sum_{k} \gamma_k p_{kj} \]

and \( \pi \) satisfies partial balance

\[ \sum_{k=0}^{J} \pi(n)q(n, n-e_j + e_k) = \sum_{k=0}^{J} \pi(n-e_j + e_k)q(n-e_j + e_k, n) \]
Insert equilibrium distribution and rates in **partial balance**

\[
\sum_{k=0}^{J} \pi(n)q(n, n-e_j + e_k) = \sum_{k=0}^{J} \pi(n-e_j + e_k)q(n-e_j + e_k, n)
\]

\[
\sum_{k=0}^{J} \phi(n) \prod_{s=1}^{J} \rho_s^{n_s} \frac{\psi(n-e_j)}{\phi(n)} \mu_j p_{jk}
\]

\[= \sum_{k=0}^{J} \phi(n-e_j + e_k) \prod_{s=1}^{J} \rho_s^{n_s} \frac{\rho_k}{\rho_j} \frac{\psi(n-e_j)}{\phi(n-e_j + e_k)} \mu_k p_{kj}
\]

\[
\psi(n-e_j) \prod_{s=1}^{J} \rho_s^{n_s-\delta_j} \sum_{k=0}^{J} \rho_j \mu_j p_{jk}
\]

\[= \psi(n-e_j) \prod_{s=1}^{J} \rho_s^{n_s-\delta_j} \sum_{k=0}^{J} \rho_k \mu_k p_{kj}
\]

This is the beauty of partial balance! **rate in = rate out**
Multi-class Queues and Stochastic Networks

Today (lecture 3): queue length based on Kelly’s lemma

Nelson, sec 10.4—10.7, Kelly, chapter 3

- Kelly Whittle network
- Alternative proof
- Quasi reversibility
- Queue disciplines
- Network of quasi reversible queues
- BCMP networks
Time reversed process

Theorem: If $X(t)$ is a stationary Markov process with transition rates $q(j,k)$, and equilibrium distribution $\pi(j)$, $j \in S$, then the reversed process $X(\tau - t)$ is a stationary Markov process with transition rates

$$q'(j,k) = \frac{\pi(k)q(k,j)}{\pi(j)} \quad j,k \in S$$

and the same equilibrium distribution.

Theorem: Kelly’s lemma
Let $X(t)$ be a stationary Markov process with transition rates $q(j,k)$. If we can find a collection of numbers $q'(j,k)$ such that $q'(j) = q(j)$, $j \in S$, and a collection of positive numbers $\pi(j)$, $j \in S$, summing to unity, such that

$$\pi(j)q(j,k) = \pi(k)q'(k,j) \quad j,k \in S$$

then $q'(j,k)$ are the transition rates of the time-reversed process, and $\pi(j)$, $j \in S$, is the equilibrium distribution of both processes.
Alternative proof: use Kelly’s lemma

Forward rates

\[ q(n, n-e_j + e_k) = \frac{\psi(n-e_j)}{\phi(n)} \mu_j p_{jk}, \quad j, k = 0, \ldots, J \]

Guess backward rates

\[ q^r(n, n-e_j + e_k) = \frac{\psi(n-e_j)}{\phi(n)} \mu_j p^r_{jk}, \quad j, k = 0, \ldots, J \]

\[ p^r_{jk} = \frac{\rho_k \mu_k}{\rho_j \mu_j} p_{kj} \]

\[ \sum_{j, k} q(n, n-e_j + e_k) = \sum_{j, k} \frac{\psi(n-e_j)}{\phi(n)} \mu_j p_{jk} = \sum_{j} \frac{\psi(n-e_j)}{\phi(n)} \mu_j \]

Check conditions

\[ \sum_{j, k} q^r(n, n-e_j + e_k) = \sum_{j, k} \frac{\psi(n-e_j)}{\phi(n)} \mu_j \frac{\rho_k \mu_k}{\rho_j \mu_j} p_{kj} \]

\[ = \sum_{j, k} \frac{\psi(n-e_j)}{\phi(n)} \mu_j \frac{\rho_j \mu_j}{\rho_j \mu_j} p_{jk} = \sum_{j} \frac{\psi(n-e_j)}{\phi(n)} \mu_j \]
Alternative proof: use Kelly’s lemma

Guess for equilibrium distribution

\[ \pi(n) = B \phi(n) \prod_{j=1}^{J} \rho_{j}^{n_{j}} \quad \rho_{j} = \gamma_{j} / \mu_{j} \quad n \in S \]

insert \quad \pi(n)q^{r}(n, n-e_{j}+e_{k}) = \pi(n-e_{j}+e_{k})q(n-e_{j}+e_{k}, n)

\[ \phi(n) \prod_{s=1}^{J} \rho_{s}^{n_{s}} \frac{\psi(n-e_{j})}{\phi(n)} \mu_{j} p_{jk}^{r} \]

\[ = \phi(n-e_{j}+e_{k}) \frac{\rho_{k}}{\rho_{j}} \prod_{s=1}^{J} \rho_{s}^{n_{s}} \frac{\psi(n-e_{j})}{\phi(n-e_{j}+e_{k})} \mu_{k} p_{kj} \]
Multi-class Queues and Stochastic Networks

Today (lecture 3): queue length based on Kelly’s lemma

Nelson, sec 10.4—10.7, Kelly, chapter 3

• Kelly Whittle network
• Alternative proof
• Quasi reversibility
• Queue disciplines
• Network of quasi reversible queues
• BCMP networks
Queue disciplines

- Operation of the queue $j$:
  1. Each job requires exponential(1) amount of service.
  2. Total service effort supplied at rate $\phi_j(n_j)$
  3. Proportion $\gamma_j(k,n_j)$ of this effort directed to job in position $k$, $k=1,\ldots, n_j$; when this job leaves, his service is completed, jobs in positions $k+1,\ldots, n_j$ move to positions $k,\ldots, n_j-1$.
  4. When a job arrives at queue $j$ he moves into position $k$ with probability $\delta_j(k,n_j+1)$, $k=1,\ldots, n_j+1$; jobs previously in positions $k,\ldots, n_j$ move to positions $k+1,\ldots, n_j+1$.

\[
\sum_{k=1}^{n} \gamma_j(k,n) = 1 \quad \sum_{k=1}^{n} \delta_j(k,n) = 1 \quad \phi_j(n) > 0 \text{ if } n > 0
\]
Queue disciplines

- Operation of the queue j:
  (i) Each job requires exponential(1) amount of service.
  (ii) Total service effort supplied at rate $\varphi_j(n_j)$
  (iii) Proportion $\gamma_j(k,n_j)$ of this effort directed to job in position $k$,
  (iv) job arriving at queue $j$ moves into position $k$ with prob. $\delta_j(k,n_j + 1)$

- Examples:
  FCFS
  LCFS
  PS
  infinite server queue

- BCMP network
Network of queues

• Multiclass queueing network, type or class $i=1,\ldots,I$
• $J$ queues
• Customer type identifies route
• Poisson arrival rate per type $\nu(i)$, $i=1,\ldots,I$
• Route $r(i,1), r(i,2), \ldots, r(i,S(i))$
• Fixed number of visits; cannot use Markov routing
• 1, 2, or 3 visits to queue: use 3 types

• Service requirements for each job at each queue $\exp(1)$

• Type $i$ at stage $s$ in queue $r(i,s)$
• $t_j(l)$: type of customer in position $l$ in queue $j$
• $s_j(l)$: stage of this customer along his route
• $c_j(l) = (t_j(l), s_j(l))$: class of this customer
• $c_j = (c_j(1), \ldots, c_j(n_j))$: state of queue $j$
• $C = (c_1, \ldots, c_J)$: Markov process representing states of the system
Network of queues: product form

- Arrival rate class $i$ to network $\nu(i)$, so also for each stage, say
  \[ \alpha_j(i, s) = \nu(i) \]

- Transition rates
  \[
  q(C, C') = \begin{cases} 
  \phi_j(n_j)\gamma_j(l, n_j) & \text{c pos l in q j is in last stage and leaves net} \\
  \phi_j(n_j)\gamma_j(l, n_j)\delta_k(m, n_k + 1) & \text{c pos l leaves q j routes to q k in pos m} \\
  \nu(i)\delta_k(m, n_k + 1) & \text{c type i enters in q k in pos m, } k = r(i, 1)
  \end{cases}
  \]
Network of queues: product form

- Arrival rate class \(i\) to network \(\nu(i)\), so also for each stage, say \(\alpha_j(i, s) = \nu(i)\)
- Transition rates

\[
q(C, C') = \begin{cases} 
\phi_j(n_j)\gamma_j(l, n_j) & \text{c pos l in q j is in last stage and leaves net} \\
\phi_j(n_j)\gamma_j(l, n_j)\delta_k(m, n_k + 1) & \text{c pos l leaves q j routes to q k in pos m} \\
\nu(i)\delta_k(m, n_k + 1) & \text{c type i enters in q k in pos m, } k = r(i, 1)
\end{cases}
\]

- Equilibrium distribution queue \(j\)

\[
\pi_j(c_j) = b_j \prod_{i=1}^{n_j} \frac{\alpha_j(t_j(l), s_j(l))}{\phi_j(l)}
\]

- Proof: reversed process for queue \(j\): \(q^r(C, C')\)
Network of queues: product form

- Arrival rate class $i$ to network $\nu(i)$, so also for each stage, say $\alpha_j(i,s) = \nu(i)$
- Transition rates
  \[
  q(C,C') = \begin{cases}
  \phi_j(n_j)\gamma_j(l,n_j) & \text{c pos l in q j is in last stage and leaves network} \\
  \phi_j(n_j)\gamma_j(l,n_j)\delta_k(m,n_k + 1) & \text{c pos l leaves q j routes to q k in pos m} \\
  \nu(i)\delta_k(m,n_k + 1) & \text{c type i enters in q k in pos m, } k = r(i,1)
  \end{cases}
  \]
- Equilibrium distribution queue $j$
  \[
  \pi_j(c_j) = b_j \prod_{i=1}^{n_j} \frac{\alpha_j(t_j(l),s_j(l))}{\phi_j(l)}
  \]
- Theorem: equilibrium distribution for open network (closed):
  \[
  \pi(C) = \prod_{j=1}^{J} \pi_j(c_j)
  \]
  Proof: reversed process $q^r(C,C')$
Multi-class Queues and Stochastic Networks

Today (lecture 3): queue length based on Kelly’s lemma

Nelson, sec 10.4—10.7, Kelly, chapter 3

- Kelly Whittle network
- Alternative proof
- **Quasi reversibility**
- Queue disciplines
- Network of quasi reversible queues
- BCMP networks
Quasi-reversibility

• Multi class queueing network, class $c \in C$

• A queue is quasi-reversible if its state $x(t)$ is a stationary Markov process with the property that the state of the queue at time $t_0$, $x(t_0)$, is independent of
  (i) arrival times of class $c$ customers subsequent to time $t_0$
  (ii) departure times of class $c$ customers prior to time $t_0$.

• Theorem
  If a queue is QR then
  (i) arrival times of class $c$ customers form independent Poisson processes
  (ii) departure times of class $c$ customers form independent Poisson processes.
Quasi-reversibility

• $S(c, x)$ set of states queue contains one more class $c$ than in state $x$
• Arrival rate class $c$ customer independent of state $x$
  \[ \alpha(c) = \sum_{x' \in S(c, x)} q(x, x') \]
• Thus arrival rate independent of prior events, and has constant rate (stationary and independent increments) \( \Rightarrow \) Poisson process
**Quasi-reversibility**

- Multi class queueing network, class \( c \in C \)
- \( S(c, x) \) set of states in which queue contains one more class \( c \) than in state \( x \)
- Arrival rate class \( c \) customer independent of state \( x \)
- Departure rate class \( c \) customer independent of state \( x \)

\[
\alpha(c) = \sum_{x' \in S(c, x)} q(x, x')
\]

\[
\pi(x)q^r(x, x') = \pi(x')q(x', x)
\]

\[
\alpha(c) = \sum_{x' \in S(c, x)} q^r(x, x')
\]

- Queue in reversed time is also quasi-reversible.
- Thus departure rate independent of prior events, and has constant rate (stationary and independent increments) \( \rightarrow \) Poisson process
Quasi-reversibility

- Multi class queueing network, class \( c \in C \)
- \( S(c, x) \) set of states in which queue contains one more class \( c \) than in state \( x \)
- Arrival rate class \( c \) customer independent of state \( x \)
  \[ \alpha(c) = \sum_{x' \in S(c, x)} q(x, x') \]
- Departure rate class \( c \) customer independent of state \( x \)
  \[ \pi(x)q^r(x, x') = \pi(x')q(x', x) \]
  \[ \alpha(c) = \sum_{x' \in S(c, x)} q^r(x, x') \]
- Operational definition QR
  \[ \alpha(c) = \sum_{x' \in S(c, x)} q(x, x') = \sum_{x' \in S(c, x)} q^r(x', x) \]
Quasi-reversibility

- Multi class queueing network, class \( c \in C \)
- \( S(c,x) \) set of states in which queue contains one more class \( c \) than in state \( x \)
- Arrival rate class \( c \) customer independent of state \( x \)
- Departure rate class \( c \) customer independent of state \( x \)

\[
\alpha(c) = \sum_{x' \in S(c,x)} q(x, x')
\]

\[
\pi(x)q^r(x, x') = \pi(x')q(x', x)
\]

\[
\alpha(c) = \sum_{x' \in S(c,x)} q^r(x, x')
\]

- Characterise QR, combine

\[
\sum_{x' \in S(c,x)} \pi(x)q(x, x') = \sum_{x' \in S(c,x)} \pi(x')q(x', x)
\]

- A form of partial balance
Quasi-reversibility and reversibility

• Birth death process with state dependent arrivals is reversible but not quasi reversible
• Ex 10.14 quasi-reversible is not reversible

• Note QR is sum of detailed balance for a subset of the transitions:

\[ \sum_{x' \in S(c,x)} \pi(x)q(x,x') = \sum_{x' \in S(c,x)} \pi(x')q(x',x) \]

• A form of partial balance
Multi-class Queues and Stochastic Networks
Today (lecture 3): queue length based on Kelly’s lemma

Nelson, sec 10.4—10.7, Kelly, chapter 3

• Kelly Whittle network
• Alternative proof
• Quasi reversibility
• Queue disciplines
• Network of quasi reversible queues
• BCMP networks
Queue disciplines

- Operation of the queue $j$:
  (i) Each job requires exponential(1) amount of service.
  (ii) Total service effort supplied at rate $\varphi_j(n_j)$
  (iii) Proportion $\gamma_j(k,n_j)$ of this effort directed to job in position $k$, $k=1,..., n_j$; when this job leaves, his service is completed, jobs in positions $k+1,..., n_j$ move to positions $k,..., n_j - 1$.
  (iv) When a job arrives at queue $j$ he moves into position $k$ with probability $\delta_j(k,n_j + 1)$, $k=1,..., n_j + 1$; jobs previously in positions $k,..., n_j$ move to positions $k+1,..., n_j + 1$.

$$\sum_{k=1}^{n} \gamma_j(k,n) = 1 \quad \sum_{k=1}^{n} \delta_j(k,n) = 1 \quad \phi_j(n) > 0 \text{ if } n > 0$$
Queue disciplines

- Operation of the queue j:
  (i) Each job requires exponential(1) amount of service.
  (ii) Total service effort supplied at rate $\phi_j(n_j)$
  (iii) Proportion $\gamma_j(k,n_j)$ of this effort directed to job in position k,
  (iv) job arriving at queue j moves into position k with prob. $\delta_j(k,n_j + 1)$

- Examples:
  FCFS
  LCFS
  PS
  infinite server queue

- BCMP network
Multi-class quasi-reversible M/M/1-FIFO queue

- Operation of the queue with C customer classes:
- (i) Each job requires exponential($\mu$) amount of service.
- (ii) Total service effort supplied at rate $\varphi(n) = 1(n > 0)$
- (iii) Proportion $\gamma(k,n) = 1(k = 1)$ of this effort to job in position $k$,
- (iv) job arriving moves into pos $k$ w.p. $\delta(k,n) = 1(k = n)$
- State of queue $c = (c_1, \ldots, c_n)$
- Arrival rate $\lambda = \Sigma_i \lambda_i$
- Utilization system: $\rho = \lambda/\mu$ and of class $c$: $\rho_c = \lambda_c/\mu$
- Equilibrium distribution:
  $$\pi(c) = (1 - \rho)\rho^n \prod_{i=1}^{n} \varphi_{c_i}$$

- Where prob part. customer is class $c$ $\varphi_c = \frac{\rho_c}{\rho}$
- And queue is quasi-reversible
Multi-class quasi-reversible M/M/1-FIFO queue

- Proof equilibrium distribution: reversed process
- M/M/1 with C customer classes, Poisson arrivals at rates $\lambda_i$ and reversed queue positions:
  - (i) Each job requires exponential($\mu$) amount of service.
  - (ii) Total service effort supplied at rate $\varphi(n)=1(n>0)$
  - (iii) Proportion $\gamma^r(k,n)=1(k=n)$ of this effort to job in position $k$,
  - (iv) job arriving moves into pos $k$ w.p. $\delta^r(k,n)=1(k=1)$
- Kelly’s lemma: $\pi(c)q(c,c') = \pi(c')q^r(c',c)$
- Row sums: obvious,

arrival transition $c' = (c,d)$: $q(c,c') = \lambda_d, q^r(c',c) = \mu$
departure transition $c = (d,c')$: $q(c,c') = \mu, q^r(c',c) = \lambda_d$

satisfied iff $\lambda_d = \rho \varphi_d \mu$
Multi-class quasi-reversible M/M/1-FIFO queue

• Proof quasi-reversibility:
• Arrivals to M/M/1 with C customer classes are Poisson at rates $\lambda_i$ and hence given $t$ independent of state at time $t$
• Reversed process also M/M/1 with Poisson arrivals (we just constructed this process). Identification of departures of process with arrivals in reversed process completes the proof.
Multi-class quasi-reversible M/M/1-FIFO queue

- Operation of the queue with C customer classes:
- Equilibrium distribution:
  \[ \pi(c_1, \ldots, c_n) = (1 - \rho) \rho^n \prod_{i=1}^{n} \varphi_{c_i} \]
- Number of customers \( N = (N_1, \ldots, N_C) \)

\[ \pi(N = n) = (1 - \rho) \left( \begin{array}{c} n \\ n_1, \ldots, n_C \end{array} \right) \prod_{i=1}^{C} \rho_{c_i}^{n_c} \]

- Total number of customers \( N \)

\[ \pi(N = n) = (1 - \rho) \rho^n \]
Multi-class Queues and Stochastic Networks

Today (lecture 3): queue length based on Kelly’s lemma

Nelson, sec 10.4—10.7, Kelly, chapter 3

- Kelly Whittle network
- Alternative proof
- Quasi reversibility
- Queue disciplines
- Network of quasi reversible queues
- BCMP networks
Symmetric queues

• Operation of the queue j:
  (i) Each job requires exponential(1) amount of service.
  (ii) Total service effort supplied at rate $\phi_j(n_j)$
  (iii) Proportion $\gamma_j(k,n_j)$ of this effort directed to job in position $k$,
  (iv) job arriving at queue j moves into position k with prob. $\delta_j(k,n_j + 1)$

\[ \gamma_j(k,n) = \delta_j(k,n) \]

• Examples: IS, LCFS, PS
• Symmetric queue QR (for general service requirement)
• Instantaneous attention
Network of quasi reversible queues

- Multiclass queueing network, type $i=1,\ldots,I$
- $J$ queues
- Customer type identifies route
- Poisson arrival rate per type $\nu(i)$, $i=1,\ldots,I$
- Route $r(i,1), r(i,2), \ldots, r(i,S(i))$
- Type $i$ at stage $s$ in queue $r(i,s)$
- $S(c,x)$ set of states in which queue contains one more class $c$ than in state $x$

- State $X(t) = (x_1(t), \ldots, x_J(t))$

- Fixed number of visits; cannot use Markov routing
- 1, 2, or 3 visits to queue: use 3 types
Network of quasi reversible queues

- Construct network by multiplying rates for individual queues
- Transition rates
  - Arrival of type \( i \) causes queue \( k = r(i,1) \) to change at
    \[
    q_k(x_k, x_k') \quad x_k' \in S_k(i,1,x_k)
    \]
  - Departure type \( i \) from queue \( j = r(i,S(i)) \)
    \[
    q_j(x_j, x_j') \quad x_j \in S_j(i,S(i),x_j')
    \]
- Routing
  \[
  q_j(x_j, x_j') \frac{q_k(x_k, x_k')}{\sum_{x' \in S_k(i,s+1,x_k)} q_k(x_k, x')} = q_j(x_j, x_j') \frac{q_k(x_k, x_k')}{\alpha_k(i,s+1)}
  \]
  \[
  x_k' \in S_k(i,s+1,x_k) \quad x_j \in S_j(i,s,x_j')
  \]
- Internal
  \[
  q_j(x_j, x_j')
  \]
Network of Quasi-reversible queues

- Rates

\[ q(x, x') = \begin{cases} 
q_k(x_k, x_k') & x_k' \in S_k(i,1,x_k) \\
q_j(x_j, x_j') & x_j \in S_j(i, S(i), x_j') \\
q_j(x_j, x_j') \frac{q_k(x_k, x_k')}{\alpha_k(i, s+1)} & x_k' \in S_k(i, s+1, x_k) \quad x_j \in S_j(i, s, x_j')
\end{cases} \]

- Theorem: For an open network of QR queues
  (i) the states of individual queues are independent at fixed time

\[ \pi(x) = \pi_1(x_1) \ldots \pi_J(x_J) \]

(ii) an arriving customer sees the equilibrium distribution
(iii) the equilibrium distribution for a queue is as it would be in isolation with arrivals forming a Poisson process.
(iv) time-reversal: another open network of QR queues
(iv) system is QR, so departures form Poisson process
Network of Quasi-reversible queues

- Proof of part (i): Kelly’s lemma
- Rates

\[
q(x, x') = \begin{cases} 
  q_k(x_k, x_k') & x_k' \in S_k(i, 1, x_k) \\
  q_j(x_j, x_j') & x_j \in S_j(i, S(i), x_j') \\
  q_j(x_j, x_j') \frac{q_k(x_k, x_k')}{\alpha_k(i, s+1)} & x_k' \in S_k(i, s+1, x_k) \quad x_j \in S_j(i, s, x_j')
\end{cases}
\]

- Transition rates reversed process (guess)

\[
q'(x, x') = \begin{cases} 
  q'_k(x_k, x_k') & x_k' \in S_k(i, S(i), x_k) \\
  q'_j(x_j, x_j') & x_j \in S_j(i, 1, x_j') \\
  q'_k(x_k, x_k') \frac{q'_j(x_j, x_j')}{\alpha_j(i, s)} & x_k \in S_k(i, s+1, x_k') \quad x_j \in S_j(i, s, x_j')
\end{cases}
\]
Network of Quasi-reversible queues

- Proof of part (i): Kelly’s lemma
- For

\[ x_k \in S_k(i, s+1, x_k'), x_j \in S_j(i, s, x_j') \]

- We have

\[ \pi_j(x_j)\pi_k(x_k)q_j(x_j, x_j') \frac{q_k(x_k, x_k')}{\alpha_k(i, s+1)} = \pi_j(x_j')\pi_k(x_k')q'_k(x_k', x_k) \frac{q'_j(x_j', x_j)}{\alpha_j(i, s)} \]

- Satisfied due to

\[ \alpha_k(i, s+1) = \alpha_j(i, s) = \nu(i) \]
Multi-class Queues and Stochastic Networks

Today (lecture 3): queue length based on Kelly’s lemma

Nelson, sec 10.4—10.7, Kelly, chapter 3

• Kelly Whittle network
• Alternative proof
• Quasi reversibility
• Queue disciplines
• Network of quasi reversible queues
• BCMP networks
Queue disciplines

• Operation of the queue j:
  (i) Each job requires exponential(1) amount of service.
  (ii) Total service effort supplied at rate $\Phi_j(n_j)$
  (iii) Proportion $\gamma_j(k,n_j)$ of this effort directed to job in position $k$, $k=1,\ldots, n_j$; when this job leaves, his service is completed, jobs in positions $k+1,\ldots, n_j$ move to positions $k,\ldots, n_j-1$.
  (iv) When a job arrives at queue j he moves into position $k$ with probability $\delta_j(k,n_j+1)$, $k=1,\ldots, n_j+1$; jobs previously in positions $k,\ldots, n_j$ move to positions $k+1,\ldots, n_j+1$.

$$\sum_{k=1}^n \gamma_j(k,n) = 1 \quad \sum_{k=1}^n \delta_j(k,n) = 1 \quad \phi_j(n) > 0 \text{ if } n > 0$$
Queue disciplines

- Operation of the queue j:
  (i) Each job requires exponential(1) amount of service.
  (ii) Total service effort supplied at rate $\phi_j(n_j)$
  (iii) Proportion $\gamma_j(k,n_j)$ of this effort directed to job in position $k$,
  (iv) job arriving at queue j moves into position $k$ with prob. $\delta_j(k,n_j + 1)$

- In the form of network of quasi reversible queues

- Examples: BCMP network
  FCFS
  LCFS
  PS
  infinite server queue,
Symmetric queues

- Operation of the queue $j$:
  (i) Each job requires exponential(1) amount of service.
  (ii) Total service effort supplied at rate $\phi_j(n_j)$
  (iii) Proportion $\gamma_j(k,n_j)$ of this effort directed to job in position $k$,
  (iv) job arriving at queue $j$ moves into position $k$ with prob. $\delta_j(k,n_j + 1)$

$$\gamma_j(k,n) = \delta_j(k,n)$$

- Examples: IS, LCFS, PS
- Symmetric queue QR for general service requirement
- Instantaneous attention
- Symmetric queue is insensitive
Quasi-reversibility and partial balance

- **QR**: fairly general queues, service disciplines, Markov routing, product form equilibrium distribution factorizes over queues.
- **PB**: fairly general relation between service rate at queues, state-dependent routing (blocking), product form equilibrium distribution factorizes over service and routing parts.
- Identical for single type queueing network with Markov routing

- **QR → partial balance**
- **NOT partial balance → QR** (exercise)
- **NOT QR → Reversibility** (see Nelson, ex 10.14)
- **NOT Reversibility → QR** (see Nelson, ex 10.12)