Advanced Queueing Theory

• Networks of queues
  (reversibility, output theorem, tandem networks, partial balance, product-form distribution, blocking, insensitivity, BCMP networks, mean-value analysis, Norton's theorem, sojourn times)

• Analytical-numerical techniques
  (matrix-analytical methods, compensation method, error bound method, approximate decomposition method)

• Polling systems
  (cycle times, queue lengths, waiting times, conservation laws, service policies, visit orders)

Richard J. Boucherie
department of Applied Mathematics
University of Twente
http://wwwhome.math.utwente.nl/~boucherierj/onderwijs/Advanced_Queueing_Theory/AQT.html
Advanced Queueing Theory
Today (lecture 9): Polling models / branching

• J.A.C. Resing. Polling systems and multitype branching processes, Queueing Systems, 13, p 409 – 426
• I. Adan: Queueing Systems, lecture notes

• M/G/1 vacation model
• Branching processes
• Polling model
• Multi-type branching process
• Polling model as branching process
• Work decomposition
M/G/1 queue with server vacations

- Consider M/G/1 queue
- Server takes a vacation (general distribution) at arbitrary times: vacation = server unavailable
- Service time independent of sequence of vacation periods preceding that service time.
- Order of service independent of service times.
- Service non-preemptive
- Rules that govern when server begins and ends vacations do not anticipate future jumps of the arrival process.
Generalized vacations

• Assume that the number of customers that arrive during a vacation is independent of the number of customers present in the system when vacation began.

• Theorem: Under this assumption

\[ \psi(z) = \zeta(z) \frac{1 - \alpha(z)}{(1 - z)\alpha'(1)} \pi(z) \]

• P.g.f. # customers present at start vacation
• P.g.f. # customers at arbitrary time in exhaustive corresp, M/G/1 vacation model during vacation
• P.g.f. # customers at arbitrary time in corresp. M/G/1
Generalized vacations: examples

- Exhaustive service: no customers at start vacation
  \[ \zeta(z) = 1 \]
- M/G/1 with gated vacations: when server returns from vacation, a gate closes, and server serves only those customers present when gate closes, service of other customers deferred to next visit
Generalized vacations: Gated service

- recall:
  \[ \overline{B}(\lambda - \lambda z) = P_A(z) \]

- Off-spring: replace customer by arrivals during his service

- Define
  \[
  R^{(1)}(z) = \overline{B}(\lambda - \lambda z) \\
  R^{(k)}(z) = R^{(1)}(R^{(k-1)}(z)), \quad k \geq 2
  \]

- is p.g.f. number of the k-th generation offspring, and
  \[
  \zeta(z) = \prod_{k=1}^{\infty} \alpha(R^{(k)}(z))
  \]
Advanced Queueing Theory
Today (lecture 9): Polling models / branching

- J.A.C. Resing. Polling systems and multitype branching processes, Queueing Systems, 13, p 409 – 426
- I. Adan: Queueing Systems, lecture notes

- M/G/1 vacation model
- Branching processes
- Polling model
- Multi-type branching process
- Polling model as branching process
- Work decomposition
Branching processes (Wolff, sec 3-9)

- Let $Y_r$ (i.i.d) be the first generation off-spring of individual $r$

  $$P(Y_r = j) = \gamma_j, \ j = 0,1,\ldots$$

- $X_n$ n-th generation off-spring of particular individual

  $$p(X_{n+1} = j \mid X_n = 1) = p_{1j} = \gamma_j$$

  $$p(X_{n+1} = j \mid X_n = i) = p_{ij} = \gamma_j^{*i}, i, j = 0,1,\ldots$$

- Pgf n-th generation off-spring individual

  $$A_n(z) = E\{z^{X_n} \mid X_0 = 1\}$$

  $$A_0(z) = 1$$

  $$A_1(z) = A(z) = \sum_{j=0}^{\infty} \gamma_j z^j$$
Branching processes (Wolff, sec 3-9)

• $X_{n+1}$ is sum of descendants of the $j$ individuals of the first generation:

\[
E\{z^{X_{n+1}} \mid X_1 = j, X_0 = 1\} = E\{z^{X_{n+1}} \mid X_1 = j\} = [A_n(z)]^j
\]

\[
A_{n+1}(z) = E\{[A_n(z)]^j \mid X_0 = 1\} = A(A_n(z))
\]
Advanced Queueing Theory
Today (lecture 8): Vacation models

• J.A.C. Resing. Polling systems and multitype branching processes, Queueing Systems, 13, p 409 – 426
• I. Adan: Queueing Systems, lecture notes

• M/G/1 vacation model
• Branching processes
  • Polling model
  • Multi-type branching process
  • Polling model as branching process
• Work decomposition
Polling models

- $N$ infinite buffer queues, $Q_1, \ldots, Q_N$
- Service time distribution at queue $j$: $B_j(\cdot)$, mean $\beta_j$, LST $\beta_j(\cdot)$
- Poisson arrivals to queue $j$ at rate $\lambda_j$
- Single server in cyclic order
- Switch over times: random variable $S_j$, mean $\sigma_j$, LST $\sigma_j(\cdot)$
Polling models: 2 queues with gated service

- $X_{ij} = \#$ customers at $Q_j$ when server polls $Q_i$
- $Y_{ij} = \#$ customers at $Q_j$ when server leaves $Q_i$

- Generating functions

$$F_i(z_1,z_2) = E\{z_1^{X_{i1}} z_2^{X_{i2}}\}, \quad i = 1,2$$
$$G_i(z_1,z_2) = E\{z_1^{Y_{i1}} z_2^{Y_{i2}}\}, \quad i = 1,2$$

- Clearly, with $\sigma_i(.)$ the LST of switchover time after visit to $Q_i$

$$F_2(z_1,z_2) = G_1(z_1,z_2)\sigma_1(\lambda_1 (1-z_1) + \lambda_2 (1-z_2)), \quad i = 1,2$$

in words: the number when server polls $Q_2$ are the sums of the numbers when server left $Q_1$ + the numbers of arrivals during the subsequent switch
Polling models: 2 queues

Lemma

\[ G_1(z_1, z_2) = F_1(B_1(\lambda_1(1-z_1) + \lambda_2(1-z_2)), z_2) \]

- With the additional relation we have four formulas
  \[ F_1(z_1, z_2) = G_2(z_1, z_2)\sigma_2(\lambda_1(1-z_1) + \lambda_2(1-z_2)) \]
  \[ G_2(z_1, z_2) = F_2(z_1, B_2(\lambda_1(1-z_1) + \lambda_2(1-z_2))) \]
  \[ F_2(z_1, z_2) = G_1(z_1, z_2)\sigma_1(\lambda_1(1-z_1) + \lambda_2(1-z_2)) \]
  \[ G_1(z_1, z_2) = F_1(B_1(\lambda_1(1-z_1) + \lambda_2(1-z_2)), z_2) \]

- And we may express, via iteration

\[ \Sigma = \lambda_1(1-z_1) + \lambda_2(1-z_2) \]

\[ F_1^{(k+1)}(z_1, z_2) = \sigma_2(\Sigma)\sigma_1(\lambda_1(1-z_1) + \lambda_2(1-\bar{B}_2(\Sigma))) \]
\[ \times F_1^{(k)}(\bar{B}_1(\lambda_1(1-z_1) + \lambda_2(1-\bar{B}_2(\Sigma))), \bar{B}_2(\Sigma)) \]
Polling: offspring

• While served at queue i, customer is replaced by random population with p.g.f.

• Exhaustive

\[ h_i(z_1, \ldots, z_N) = \theta_i(\sum_{j \neq i} \lambda_j (1 - z_j)) \]

• gated

\[ h_i(z_1, \ldots, z_N) = \beta_i(\sum_j \lambda_j (1 - z_j)) \]
Polling models: 2 queues

Starting with the initial relations

\[ F_1(z_1, z_2) = G_2(z_1, z_2) \sigma_2 (\lambda_1 (1 - z_1) + \lambda_2 (1 - z_2)) \]
\[ G_2(z_1, z_2) = F_2(z_1, h_2(z_1, z_2)) \]
\[ F_2(z_1, z_2) = G_1(z_1, z_2) \sigma_1 (\lambda_1 (1 - z_1) + \lambda_2 (1 - z_2)) \]
\[ G_1(z_1, z_2) = F_1(h_1(z_1, z_2)), z_2 \]

Iterating, we get

\[ \Sigma = \lambda_1 (1 - z_1) + \lambda_2 (1 - z_2) \]
\[ F_1^{(k+1)}(z_1, z_2) = \sigma_2 (\Sigma) \sigma_1 (\lambda_1 (1 - z_1) + \lambda_2 (1 - h_2(z_1, z_2))) \]
\[ F_1^{(k)}(h_1(z_1, h_2(z_1, z_2)), h_2(z_1, z_2)) \]
Polling: offspring

- While served at queue $i$, customer is replaced by random population with p.g.f.

- Exhaustive

$$h_i(z_1, ..., z_N) = \theta_i \left( \sum_{j \neq i} \lambda_j (1 - z_j) \right)$$

- gated

$$h_i(z_1, ..., z_N) = \beta_i \left( \sum_j \lambda_j (1 - z_j) \right)$$
Polling : offspring

• Assumption
  If the server arrives at $Q_i$ to find $k_i$ customers, then during the course of the servers visit, each of these $k_i$ customers is effectively replaced in an iid manner by a random population with p.g.f. $h_i(z_1, \ldots, z_N)$

• Examples: exhaustive, gated
• Not included: 1-limited, because all but first have pgf $s_i$

• Bernoulli-type. When server arrives all customers present are handled as follows: customer is served, and all offspring served with probability $p_i$. 
For the Bernoulli-type service discipline the p.g.f. \( h_i(s_1, \ldots, s_N) \) is given by

\[
h_i(s_1, \ldots, s_N) = \Phi_{p_i, i} \left( \sum_{j \neq i} \lambda_j (1 - s_j) \right) \\
\quad + \frac{(1 - p_i) \beta_i (\sum_j \lambda_j (1 - s_j))}{s_i - p_i \beta_i (\sum_j \lambda_j (1 - s_j))} \left( s_i - \Phi_{p_i, i} \left( \sum_{j \neq i} \lambda_j (1 - s_j) \right) \right),
\]

where \( \Phi_{p_i, i}(s) \) is the unique solution of

\[
\Phi_{p_i, i}(s) = p_i \beta_i (s + \lambda_i (1 - \Phi_{p_i, i}(s))).
\]
Advanced Queueing Theory
Today (lecture 8): Vacation models

• J.A.C. Resing. Polling systems and multitype branching processes, Queueing Systems, 13, p 409 – 426
• I. Adan: Queueing Systems, lecture notes

• M/G/1 vacation model
• Branching processes
• Polling model
  • Multi-type branching process
• Polling model as branching process
• Work decomposition
Multitype branching processes

finite number $N$ of particle types. To define the particle production we need $N$ generating functions, each in $N$ variables,

$$f^{(i)}(s_1, \ldots, s_N) = \sum_{j_1, \ldots, j_N \geq 0} p^{(i)}(j_1, \ldots, j_N)s_1^{j_1} \cdots s_N^{j_N}, \quad i = 1, \ldots, N, \quad (6)$$

where $p^{(i)}(j_1, \ldots, j_N)$ is the probability that a type $i$ particle produces $j_1$ particles of type 1, $j_2$ of type 2, \ldots, $j_N$ of type $N$, respectively. Let $m_{ij}$ be the expected number

- Polling model is MTBP with immigration
- Switchover times: immigration in each state
- No switchover times: immigration in state zero
MTBP with immigration in each state

Consider the multitype branching process with an independent immigration component in each state. So in addition to the generating functions \( f^{(i)}(s_1, \ldots, s_N), \ i = 1, \ldots, N, \) representing the offspring distributions, an additional generating function \( g(s_1, \ldots, s_N) \) is given, representing the immigration distribution, i.e.

\[
g(s_1, \ldots, s_N) = \sum_{j_1, \ldots, j_N \geq 0} q(j_1, \ldots, j_N) s_1^{j_1} \cdots s_N^{j_N},
\]

where \( q(j_1, \ldots, j_N) \) is the probability that a group of immigrants consists of \( j_1 \) particles of type 1, \( j_2 \) of type 2, \ldots, \( j_N \) of type \( N \).

Define the functions \( f_n(s_1, \ldots, s_N) \) inductively by

\[
\begin{aligned}
    f_0(s_1, \ldots, s_N) &= (s_1, \ldots, s_N), \\
    f_n(s_1, \ldots, s_N) &= (f^{(1)}(f_{n-1}(s_1, \ldots, s_N)), \ldots, f^{(N)}(f_{n-1}(s_1, \ldots, s_N))).
\end{aligned}
\]
THEOREM 1

Let $Z_n = (Z_n^{(1)}, \ldots, Z_n^{(N)})$ be a multitype branching process with immigration in each state with offspring generating functions $f^{(i)}(s_1, \ldots, s_N), i = 1, \ldots, N$, and immigration generating function $g(s_1, \ldots, s_N)$. Let the mean matrix $M$ corresponding to the branching process be primitive and its maximal eigenvalue $\nu_{\text{max}} < 1$. Assume the Markov chain $Z_n$ is irreducible and aperiodic. Then a necessary and sufficient condition for the existence of a stationary distribution $\pi(j_1, \ldots, j_N)$ for the process $Z_n$ is

$$\sum_{j_1+\ldots+j_N > 0} q(j_1, \ldots, j_N) \log(j_1 + \ldots + j_N) < \infty. \quad (9)$$

When this condition is satisfied, the generating function $P(s_1, \ldots, s_N)$ of the distribution $\pi(j_1, \ldots, j_N)$ satisfies

$$P(s_1, \ldots, s_N) = \prod_{n=0}^{\infty} g(f_n(s_1, \ldots, s_N)). \quad (10)$$

- Recall result for M/G/1 gated vacation!
MTBP with immigration in each state

Proof
See Quine [14]. The formula (10) is derived by iteration of

\[ P(s_1, \ldots, s_N) = g(s_1, \ldots, s_N)P(f_1(s_1, \ldots, s_N)). \]

- Each particle replaced by its off-spring, and independently by immigration
MTBP with immigration in state zero

• Immigration only in state zero, not in all states

THEOREM 2

Let \( Z_n = (Z_n^{(1)}, \ldots, Z_n^{(N)}) \) be a multitype branching process with immigration at state zero with offspring generating functions \( f^{(i)}(s_1, \ldots, s_N), \ i = 1, \ldots, N \), and immigration generating function \( g(s_1, \ldots, s_N) \). Let the mean matrix \( M \) corresponding to the branching process be primitive and its maximal eigenvalue \( \nu_{\text{max}} < 1 \). Assume the Markov chain \( Z_n \) is irreducible and aperiodic, and finally assume \( Z_0 = (0, \ldots, 0) \). Then a necessary and sufficient condition for the existence of a stationary distribution \( \pi(j_1, \ldots, j_N) \) for the process \( Z_n \) is

\[
\sum_{j_1, \ldots, j_N \geq 0 \atop j_1 + \cdots + j_N > 0} q(j_1, \ldots, j_N) \log(j_1 + \cdots + j_N) < \infty. \tag{12}
\]

When this condition is satisfied, the generating function \( P(s_1, \ldots, s_N) \) of the distribution \( \pi(j_1, \ldots, j_N) \) satisfies

\[
P(s_1, \ldots, s_N) = 1 - \pi(0, \ldots, 0) \sum_{n=0}^{\infty} (1 - g(f_n(s_1, \ldots, s_N))), \tag{13}
\]

where

\[
\pi(0, \ldots, 0) = \left[ 1 + \sum_{n=0}^{\infty} (1 - g(f_n(0, \ldots, 0))) \right]^{-1}. \tag{14}
\]
MTBP with immigration in state zero

It only remains to prove eq. (13). Define the transition probabilities

\[ p_{i_1, \ldots, i_N; j_1, \ldots, j_N} := \Pr(Z_{n+1} = (j_1, \ldots, j_N) | Z_n = (i_1, \ldots, i_N)) \]

and define

\[ P_{i_1, \ldots, i_N}(s_1, \ldots, s_N) := \sum_{j_1, \ldots, j_N \geq 0} p_{i_1, \ldots, i_N; j_1, \ldots, j_N} s_1^{j_1} \cdots s_N^{j_N}. \]

Then

\[ P_{i_1, \ldots, i_N}(s_1, \ldots, s_N) = g(s_1, \ldots, s_N)1[(i_1, \ldots, i_N) = (0, \ldots, 0)] \]

\[ + \left[ f^{(1)}(s_1, \ldots, s_N) \right]^{i_1} \cdots \left[ f^{(N)}(s_1, \ldots, s_N) \right]^{i_N} 1[(i_1, \ldots, i_N) \neq (0, \ldots, 0)]. \]

Now we use
MTBP with immigration in state zero

Now we use

$$\pi_{j_1, \ldots, j_N} = \sum_{i_1, \ldots, i_N} \pi_{i_1, \ldots, i_N} p_{i_1, \ldots, i_N; j_1, \ldots, j_N}$$

to conclude

$$P(s_1, \ldots, s_N) = \sum_{j_1, \ldots, j_N} \pi_{j_1, \ldots, j_N} s_1^{j_1} \cdots s_N^{j_N}$$

$$= \sum_{j_1, \ldots, j_N} \sum_{i_1, \ldots, i_N} \pi_{i_1, \ldots, i_N} p_{i_1, \ldots, i_N; j_1, \ldots, j_N} s_1^{j_1} \cdots s_N^{j_N}$$

$$= \sum_{i_1, \ldots, i_N} \pi_{i_1, \ldots, i_N} p_{i_1, \ldots, i_N} (s_1, \ldots, s_N)$$

$$= P(f_1(s_1, \ldots, s_N)) + \pi(0, \ldots, 0)[g(s_1, \ldots, s_N) - 1]. \quad (16)$$

Iteration of this equation, together with $f_n(s_1, \ldots, s_N) \to 1$ and $\sum (g(f_n(s_1, \ldots, s_N)) - 1) < \infty$, yields (13).
Advanced Queueing Theory
Today (lecture 8): Vacation models

- J.A.C. Resing. Polling systems and multitype branching processes, Queueing Systems, 13, p 409 – 426
- I. Adan: Queueing Systems, lecture notes

- M/G/1 vacation model
- Branching processes
- Polling model
- Multi-type branching process
- Polling model as branching process
- Work decomposition
THEOREM 3

Consider a polling system with switchover times $S_j$ with LST $\sigma_j(\cdot)$. Assume that the service discipline at $Q_j$ satisfies property 1 with p.g.f. $h_j(s_1, \ldots, s_N)$, $j = 1, \ldots, N$. Then the numbers of customers in the different queues at time points $t_n$ constitute a multitype branching process with immigration in each state, where the offspring generating functions $f^{(i)}(s_1, \ldots, s_N)$, $i = 1, \ldots, N$, are given by

$$f^{(i)}(s_1, \ldots, s_N) = h_i(s_1, \ldots, s_i, f^{(i+1)}(s_1, \ldots, s_N), \ldots, f^{(N)}(s_1, \ldots, s_N))$$

and the immigration generating function $g(s_1, \ldots, s_N)$ is given by

$$g(s_1, \ldots, s_N) = \prod_{i=1}^{N} \sigma_i \left( \sum_{k=1}^{i} \lambda_k (1 - s_k) + \sum_{k=i+1}^{N} \lambda_k (1 - f^{(k)}(s_1, \ldots, s_N)) \right).$$
Polling model with switchover times as MTBP

• Proof: ancestral line
  Consider two successive times server at Q1: t(n) and t(n+1). Let $C_A$ customer be typical customer present.

  Collection of cust present at t(n+1) consists of replacements of customers present at t(n) and the replacements of customers $C_B$ arriving during switching intervals. Poisson arrivals and all serv disciplines satisfy property 1 implies MTBP with immigration in each state.

• Compute p.g.f.’s $g$ and $f$
Polling model with switchover times as MTBP

Next we shall calculate the offspring generating functions. By definition we have \( f^{(N)}(s_1, \ldots, s_N) = h_N(s_1, \ldots, s_N) \). Assume we have calculated \( f^{(k)}(s_1, \ldots, s_N) \) for \( k = i + 1, \ldots, N \). Then we will calculate \( f^{(i)}(s_1, \ldots, s_N) \) by conditioning on the number of customers in the ancestral line present at the moment that the server leaves \( Q_i \). With the notation

\[
\begin{align*}
    p(i) &= \Pr \{ \text{collection of effective replacants of a } c_A \text{ customer from } Q_i \text{ consists of } i_k \text{ customers in } Q_k, k = 1, \ldots, N \}, \\
    q(j) &= \Pr \{ \text{collection of customers in the ancestral line at time that server leaves } Q_i \text{ consists of } j_k \text{ customers in } Q_k, k = 1, \ldots, N \}, \\
    p(i|j) &= \Pr \{ \text{collection of effective replacants of a } c_A \text{ customer from } Q_i \text{ consists of } i_k \text{ customers in } Q_k, k = 1, \ldots, N | \text{ collection of customers in the ancestral line at time that server leaves } Q_i \text{ consists of } j_k \text{ customers in } Q_k, k = 1, \ldots, N \},
\end{align*}
\]
Polling model with switchover times as MTBP

\[ f^{(i)}(s_1, \ldots, s_N) = \sum_{i_1, \ldots, i_N \geq 0} p(i) s_1^{i_1} \ldots s_N^{i_N} \]

\[ = \sum_{i_1, \ldots, i_N \geq 0} s_1^{i_1} \ldots s_N^{i_N} \sum_{0 \leq j_k \leq i_k, k=1,\ldots,i} q(j) p(i|j) \]

\[ = \sum_{j_1, \ldots, j_N \geq 0} q(j) s_1^{j_1} \ldots s_i^{j_i} \sum_{i_k \geq j_k, k=1,\ldots,i} p(i|j) s_1^{i_1-j_1} \ldots s_i^{i_i-j_i} s_{i+1}^{j_{i+1}} \ldots s_N^{j_N} \]

\[ = \sum_{j_1, \ldots, j_N \geq 0} q(j) s_1^{j_1} \ldots s_i^{j_i} [f^{(i+1)}(s_1, \ldots, s_N)]^{j_{i+1}} \ldots [f^{(N)}(s_1, \ldots, s_N)]^{j_N} \]

\[ = h_i(s_1, \ldots, s_i, f^{(i+1)}(s_1, \ldots, s_N), \ldots, f^{(N)}(s_1, \ldots, s_N)). \]
Polling model without switchover times as MTBP

THEOREM 4

Consider a polling system without switchover times. Assume that the service discipline at \( Q_j \) satisfies property 1 with p.g.f. \( h_j(s_1, \ldots, s_N), j = 1, \ldots, N \). Then the numbers of customers in the different queues at time points \( t_n \) constitute a multi-type branching process with immigration in state zero, where the offspring generating functions \( f^{(i)}(s_1, \ldots, s_N), i = 1, \ldots, N \), are given by

\[
f^{(i)}(s_1, \ldots, s_N) = h_i(s_1, \ldots, s_i, f^{(i+1)}(s_1, \ldots, s_N), \ldots, f^{(N)}(s_1, \ldots, s_N))
\]

and the immigration generating function \( g(s_1, \ldots, s_N) \) is given by

\[
g(s_1, \ldots, s_N) = \sum_{j=1}^{N} \frac{\lambda_j}{\lambda} f^{(j)}(s_1, \ldots, s_N), \tag{21}
\]

where \( \lambda := \sum_{i=1}^{N} \lambda_i \).
Advanced Queueing Theory
Today (lecture 8): Vacation models

• J.A.C. Resing. Polling systems and multitype branching processes, Queueing Systems, 13, p 409 – 426
• I. Adan: Queueing Systems, lecture notes

• M/G/1 vacation model
• Branching processes
• Polling model
• Multi-type branching process
• Polling model as branching process

• Work decomposition
Work conservation: multi class M/G/1

• Amount of work in system independent of service discipline

• $E[V]=E[V^{FCFS}]$ for any work conserving discipline

• Multi class queue, $i=1,...,N$
  arrival rates $\lambda_i$
  service times $B_i$, mean $\beta_i$, second moment $\beta_i^{(2)}$ finite
• Traffic intensity $\rho = \sum_i \lambda_i \beta_i$
• Stability $\rho < 1$
Work conservation: multi class M/G/1

• Observe system as single class M/G/1

• Arrival rate \( \lambda = \sum_i \lambda_i \)

• Service times \( B \) distributed as \( B_i \) w.p. \( \lambda_i / \lambda \)

• In particular
  \[
  \beta = \mathbb{E}[B] = \sum_i \beta_i \lambda_i / \lambda = \rho / \lambda ,
  \]

  \[
  \beta^{(2)} = \mathbb{E}[B^2] = \sum_i \beta_i^{(2)} \lambda_i / \lambda
  \]

• Pollaczek-Khintchine:

\[
\mathbb{E}[V_{FCFS}] = \sum_i \lambda_i \beta_i^{(2)}/[2(1-\rho)]
\]
Work conservation: conservation law

- We may decompose the total amount of work in the system:

  \[ E[V] = \sum_i E[V_i] \]

  with \( V_i \) representing amount of class i work in system

- Assume FCFS in each class:

  \[ E[V_i] = E[L_i] \beta_i + \rho_i E[R_i] \]

  with \( L_i \) number of waiting class i customers at arbitrary epoch in equilibrium, excluding possible class i customer \( R_i \) remaining service time of class i customer, if any
Work conservation: conservation law

- Little’s law $E[L_i] = \lambda_i E[W_i]$

  with $W_i$ the waiting time in equil of arbitrary class $i$ cust, excluding its own service time

- Further, via renewal argument: $E[R_i] = \beta_i^{(2)}/[2 \beta_i]$

$$E[V] = \sum_{i=1}^{N} \lambda_i E[W_i] \beta_i + \sum_{i=1}^{N} \rho_i \frac{\beta_i^{(2)}}{2\beta_i} = \sum_{i=1}^{N} \rho_i E[W_i] + \frac{1}{2} \sum_{i=1}^{N} \lambda_i \beta_i^{(2)}$$
Conservation law

• Inserting gives
\[
\sum_{i=1}^{N} \rho_i E[W_i] + \frac{1}{2} \sum_{i=1}^{N} \lambda_i \beta_i^{(2)} = \frac{\sum_{i=1}^{N} \lambda_i \beta_i^{(2)}}{2(1 - \rho)}
\]

• So that
\[
\sum_{i=1}^{N} \rho_i E[W_i] = \rho \frac{\sum_{i=1}^{N} \lambda_i \beta_i^{(2)}}{2(1 - \rho)}
\]

• Which is called conservation law
• If we want to decrease \( E[W_i] \) for some class, then we must increase \( E[W_j] \) for at least one other class
Polling model

• Polling as M/G/1: single server visits all the queues

• Polling as M/G/1 vacation queue includes switching times
Work decomposition: pseudo conservation law

- Expectations in (*)

\[
\sum_{i=1}^{n} \rho_i E W_i = \rho \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1 - \rho)} + EV_I.
\]

- \(EV_I = 0\): conservation law, weighted sum of waiting times does not depend on scheduling discipline

- Remains to compute \(EV_I\)

For strictly cyclic polling, Boxma & Groenendijk [42] show that \(EV_I\) may be determined as follows:

\[
EV_I = \rho \frac{s^{(2)}}{2s} + \frac{s}{2(1 - \rho)} \left[ \rho^2 - \sum_{i=1}^{n} \rho_i^2 \right] + \sum_{i=1}^{n} EZ_{ii},
\]

(2.14)

with \(Z_{ii}\) denoting the amount of work left behind by the server at \(Q_i\) at the completion of a visit.
Work decomposition (Borst sec 2.3)

pleasing circumstance however, EZ\textsubscript{ii} is determined by the service discipline at Q\textsubscript{i} only, i.e., not by the service discipline at Q\textsubscript{j}, j \neq i, e.g. [42]:

I. Exhaustive service:

\[ EZ\textsubscript{ii} = 0. \]

II. Gated service:

\[ EZ\textsubscript{ii} = \frac{\rho_i^2 s}{1 - \rho}. \]

III. 1-Limited service:

\[ EZ\textsubscript{ii} = \frac{\rho_i^2 s}{1 - \rho} + \rho_i \frac{\lambda_i s}{1 - \rho} EW_i. \]

IV. 1-Decrementing service:

\[ EZ\textsubscript{ii} = -\frac{\rho_i^2 \lambda_i^2 \rho_i^{(2)} s}{2(1 - \rho)} + \rho_i \frac{\lambda_i (1 - \rho_i) s}{1 - \rho} EW_i. \]

Substituting the above expressions into (2.15) yields the pseudo-conservation laws, which before were only known to hold in cases with the same service disciplines at each of the queues.
Exercise

• Consider the Bernoulli type service discipline of Resing, 1993, Lemma 1. Let the service discipline at $Q_i$ be Bernoulli type. Obtain an expression for the distribution of the amount of work left behind by the server at $Q_i$ at the completion of a visit.

• Consider the vacation queue with Bernoulli service type service discipline. Obtain an expression for the LST of the amount of work in the queue, $s(z)$, when the server leaves on vacation.