E.6 Approximate Analysis of Non-product-Form Queueing Networks

- Non exponentially distributed service times
- Priorities (time independent and time dependent)
- Different service times of different classes at Type-1 nodes
- Finite queues (Queueing networks with blocking)
- Parallel processing, synchronisation
- Fork-Join-Systems
- Queueing networks with heterogeneous M/M/m- oder G/G/m- nodes
- Simultaneous resource possession

1 Solution Methods

- Approximation of a non-product-form queueing network by a product-form queueing network
- Markovanalysis (expensive) $\rightarrow$ MOSEL
- Simulation (very expensive) $\rightarrow$ PEPSY
- Approximation methods $\rightarrow$ PEPSY
  - Extension of the product-form methods
  - Iterative use of product-form methods
2 Closed Non-Product-Form Queueing Networks

- At least one M/G/m-FCFS-node (-/G/m-FCFS-node)
  - Diffusionsapproximation
  - Maximum entropy method
  - Method of Marie
  - Summation method
  - Bottleneck Approximation
  - Robustness

Robustness:

➤ -/G/1-FCFS- and -/G/m-FCFS-nodes are replaced by M/M/1-FCFS- or M/M/m-FCFS-nodes

➤ Only little influence of the coefficient of variation of the non-product-form nodes (-/G/1 und -/G/m) in case of closed networks (not valid for open networks !!).

➤ This property is called "Robustness".

➤ The accuracy of the robustness is sufficient for many applications.
**Example**

Network 1

Network 2

Network 3

Coefficient of variation:

<table>
<thead>
<tr>
<th>Combinations</th>
<th>Node 1 $c_{B_1}^2$</th>
<th>Node 2 $c_{B_2}^2$</th>
<th>Node 3 $c_{B_3}^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>b</td>
<td>0.2</td>
<td>0.4</td>
<td>0.8</td>
</tr>
<tr>
<td>c</td>
<td>4.0</td>
<td>1.0</td>
<td>0.2</td>
</tr>
<tr>
<td>d</td>
<td>2.0</td>
<td>4.0</td>
<td>8.0</td>
</tr>
</tbody>
</table>

Service rates:

<table>
<thead>
<tr>
<th></th>
<th>$\mu_1$</th>
<th>$\mu_2$</th>
<th>$\mu_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Balanced network</td>
<td>0.5</td>
<td>0.333</td>
<td>0.666</td>
</tr>
<tr>
<td>Network with bottleneck</td>
<td>0.5</td>
<td>1.0</td>
<td>2.0</td>
</tr>
</tbody>
</table>
E.6 Approximate Analysis of Non-product-Form Queueing Networks

➤ Throughput of the "Balanced Network":

<table>
<thead>
<tr>
<th>Squared Coefficient of Variations</th>
<th>Network 1</th>
<th>Network 2</th>
<th>Network 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$K = 5$</td>
<td>$K = 10$</td>
<td>$K = 5$</td>
</tr>
<tr>
<td>a</td>
<td>0.43</td>
<td>0.47</td>
<td>0.41</td>
</tr>
<tr>
<td>b</td>
<td>0.47</td>
<td>0.50</td>
<td>0.45</td>
</tr>
<tr>
<td>c</td>
<td>0.39</td>
<td>0.44</td>
<td>0.48</td>
</tr>
<tr>
<td>d</td>
<td>0.37</td>
<td>0.42</td>
<td>0.33</td>
</tr>
</tbody>
</table>

➤ Throughput of the "Network with Bottleneck":

<table>
<thead>
<tr>
<th>Squared Coefficient of Variations</th>
<th>Network 1</th>
<th>Network 2</th>
<th>Network 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$K = 5$</td>
<td>$K = 10$</td>
<td>$K = 5$</td>
</tr>
<tr>
<td>a</td>
<td>0.50</td>
<td>0.50</td>
<td>0.51</td>
</tr>
<tr>
<td>b</td>
<td>0.50</td>
<td>0.50</td>
<td>0.52</td>
</tr>
<tr>
<td>c</td>
<td>0.49</td>
<td>0.50</td>
<td>0.52</td>
</tr>
<tr>
<td>d</td>
<td>0.47</td>
<td>0.50</td>
<td>0.47</td>
</tr>
</tbody>
</table>

➤ Throughput as a function of the number of jobs in the network:

<table>
<thead>
<tr>
<th>$K$</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>10</th>
<th>20</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>0.842</td>
<td>0.907</td>
<td>0.940</td>
<td>0.996</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>b</td>
<td>–</td>
<td>0.970</td>
<td>0.991</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>c</td>
<td>–</td>
<td>0.856</td>
<td>0.894</td>
<td>0.972</td>
<td>0.998</td>
<td>1.00</td>
</tr>
<tr>
<td>d</td>
<td>0.716</td>
<td>0.766</td>
<td>0.805</td>
<td>0.917</td>
<td>0.984</td>
<td>1.00</td>
</tr>
</tbody>
</table>
3 Open Non-Product-Form Queueing Networks

- **Methods:**
  - Diffusion approximation
  - Maximum entropy method
  - Decomposition methods
    - Pujolle
    - Whitt
    - Gelenbe
    - Chylla
    - Kühn

- **Decomposition methods**
  - Interarrival times and service times are **arbitrarily** distributed and given by the first and second moment (Coefficient of variation).
  - Queueing discipline is FCFS, queue length is not limited
  - Multiple job classes (no class switching)
  - Multiple Server nodes ( -/G/m ) are possible.
Algorithm (Example: method of Whitt):

- **Step 1**: Arrival rates and utilization of the nodes:

\[
\lambda_i = \lambda_0 + \sum_{j=1}^{N} \lambda_j \cdot p_{ji}
\]

\[
\rho_i = \frac{\lambda_i}{m_i \cdot \mu_i}
\]

- **Step 2**: Iterative computation of the coefficient of variation of inter arrival times of the individual nodes (see below).

**Step 3**: Mean queue length and other performance measures

- **M/M/m-FCFS**:

\[
\overline{Q}_{iM/M/m} = \frac{\rho_i}{1 - \rho_i} \cdot P_{mi}
\]

Probability of waiting \( P_{mi} \) (\( = \rho_i \) für M/M/1)

- **Allen-Cunneen formula for G/G/m-FCFS**:

\[
\overline{Q}_{iAC} \approx \overline{Q}_{iM/M/m} \cdot \frac{(c_{Ai}^2 + c_{Bi}^2)}{2}
\]
Krämer/Langenbach-Belz formula for G/G/m-FCFS:

\[ Q_{iKLB} \approx Q_{iAC} \cdot G_{KLB} \]

with:

\[
G_{KLB} = \begin{cases} 
    e \left( \frac{2}{3} \cdot \frac{(1 - \rho_i)}{P_{mi}} \cdot \frac{(1 - c_{Ai}^2)^2}{(c_{Ai}^2 + c_{Bi}^2)} \right), & c_{Ai}^2 \leq 1, \\
    e \left( (1 - \rho_i) \cdot \frac{(c_{Ai}^2 - 1)}{(c_{Ai}^2 + c_{Bi}^2)} \right), & c_{Ai}^2 > 1 
\end{cases}
\]

- **Step 2**: Iterative computation of the coefficient of variation of inter arrival times of the individual nodes (Whitt):
E.6 Approximate Analysis of Non-product-Form Queueing Networks

- **Initialization:**  \( c_{ij} = 1 \quad i, j = 1, 2, \ldots, N \)

- **Merging:**  \( i = 1, 2, \ldots, N \)

\[
c_{Ai}^2 = \frac{1}{\lambda_i} \cdot \left( \sum_{j=1}^{N} c_{ji}^2 \cdot \lambda_j \cdot p_{ji} + c_{0i}^2 \cdot \lambda_0 \cdot p_{0i} \right)
\]

- **Flow:**  \( i = 1, 2, \ldots, N \)

\[
c_{Di}^2 = 1 + \rho_i^2 \cdot \left( \frac{c_{Bi}^2 - 1}{\sqrt{n_i}} \right) + (1 - \rho_i^2) \cdot (c_{Ai}^2 - 1)
\]

- **Splitting:**  \( i, j = 1, 2, \ldots, N \)

\[
c_{ij}^2 = 1 + p_{ij} \cdot (c_{Di}^2 - 1)
\]

**Example:**

![Queueing Network Diagram](image)

\( p_{12} = 0.5, \quad p_{13} = 0.5, \quad p_{31} = 0.6, \quad p_{21} = p_{41} = 1 \)

\( \mu_1 = 12.5, \quad \mu_2 = 33.333 \quad \mu_3 = 16.666, \quad \mu_4 = 20 \)

\( c_{B1}^2 = 2.0, \quad c_{B2}^2 = 6.0, \quad c_{B3}^2 = 0.5, \quad c_{B4}^2 = 0.2, \quad c_{04}^2 = 4.0 \)
E.6 Approximate Analysis of Non-product-Form Queueing Networks

• **Step 1:** Arrival rates and utilizations;

\[
\begin{align*}
\lambda_1 &= 20, \quad \lambda_2 = 10, \quad \lambda_3 = 10, \quad \lambda_4 = 4 \\
\rho_1 &= 0.8, \quad \rho_2 = 0.3, \quad \rho_3 = 0.6, \quad \rho_4 = 0.2
\end{align*}
\]

• **Step 2:** Coefficient of variation of inter arrival times:

➤ **Initialization:**

\[
c^2_{12} = c^2_{13} = c^2_{21} = c^2_{31} = c^2_{41} = 1
\]

1. **Iteration:**

➤ **Merging:**

\[
c^2_{A_1} = \frac{1}{\lambda_1} \left( c^2_{21} \lambda_2 p_{21} + c^2_{31} \lambda_3 p_{31} + c^2_{41} \lambda_4 p_{41} \right)
\]

\[
= \frac{1}{20} \left( 1 \cdot 10 \cdot 1 + 1 \cdot 10 \cdot 0.6 + 1 \cdot 4 \cdot 1 \right) = 1.
\]

\[
c^2_{A_2} = 1, \quad c^2_{A_3} = 1, \quad c^2_{A_4} = 4
\]
Flow:

\[ c_{D_1}^2 = 1 + \frac{\rho_1^2(c_{B_1}^2 - 1)}{\sqrt{m_1}} + (1 - \rho_1^2)(c_{A_1}^2 - 1) \]
\[ = 1 + \frac{0.64(2 - 1)}{\sqrt{2}} + (1 - 0.64)(1 - 1) \]
\[ = 1.453 \]
\[ c_{D_2}^2 = 1.45, \quad c_{D_3}^2 = 0.82, \quad c_{D_4}^2 = 3.848. \]

Splitting:

\[ c_{12}^2 = 1 + p_{12} \cdot (c_{D_1}^2 - 1) = 1.226 \]
\[ c_{13}^2 = 1.226, \quad c_{21}^2 = 1.450, \quad c_{31}^2 = 0.892, \quad c_{41}^2 = 3.848. \]

2. Iteration:

Merging:

\[ c_{A_1}^2 = 1.762, \quad c_{A_2}^2 = 1.226, \quad c_{A_3}^2 = 1.226, \quad c_{A_4}^2 = 4.0 \]

Flow:

\[ c_{D_1}^2 = 1.727, \quad c_{D_2}^2 = 1.656, \quad c_{D_3}^2 = 0.965, \quad c_{D_4}^2 = 3.848 \]

Splitting:

\[ c_{12}^2 = 1.363, \quad c_{13}^2 = 1.363, \quad c_{21}^2 = 1.656 \]
\[ c_{31}^2 = 0.979, \quad c_{41}^2 = 3.848. \]
### E.6 Approximate Analysis of Non-product-Form Queueing Networks

#### Step 3: Mean number of jobs

<table>
<thead>
<tr>
<th>Iteration</th>
<th>$c_{A_1}^2$</th>
<th>$c_{A_2}^2$</th>
<th>$c_{A_3}^2$</th>
<th>$c_{A_4}^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Methods</td>
<td>$K_1$</td>
<td>$K_2$</td>
<td>$K_3$</td>
<td>$K_4$</td>
</tr>
<tr>
<td>AC</td>
<td>6.48</td>
<td>0.78</td>
<td>1.46</td>
<td>0.31</td>
</tr>
<tr>
<td>KLB</td>
<td>6.21</td>
<td>0.77</td>
<td>1.42</td>
<td>0.27</td>
</tr>
<tr>
<td>Sim</td>
<td>4.62</td>
<td>0.57</td>
<td>1.38</td>
<td>0.23</td>
</tr>
</tbody>
</table>

AC: Allen-Cunneen-Approximation  
KLB: Krämer/Langenbach-Belz-Approximation
4 Priority Queueing Networks

Description:

- Multiple job classes with priorities 1, 2, ..., R
- 1: highest priority; R: lowest priority
- Two additional node types:
  - M/M/1-FCFS-NONPRE (without preemption)
  - M/M/1-FCFS-PRE (with preemption)
- Only approximate solutions:
  - Extended MVA (PRIOMVA)
  - Shadow Method

PRIOMVA:

MVA for multiple job classes:

Mean response time (arrival theorem):

\[
\bar{T}_{ir}(k) = \begin{cases} 
\frac{1}{\mu_{ir}} \left[ 1 + \sum_{s=1}^{R} K_{is}(k - 1_r) \right] & \text{Type-1,2,4} \\
\frac{1}{\mu_{ir}} & \text{Type-3.}
\end{cases}
\]

\((k - 1_r) = (k_1, \ldots, k_r - 1, \ldots, k_R)\) is the population vector with one class-\(r\) job less in the system
E.6 Approximate Analysis of Non-product-Form Queueing Networks

Mean response time (M/M/1-PRE node):

\[
\bar{T}_{ir}(k) = \frac{1}{\mu_{ir}} + \sum_{s=1}^{r} \frac{K_{is}(k - 1_{r})}{\mu_{is}},
\]

\[
1 - \sum_{s=1}^{r-1} \rho'_{is},
\]

Mean response time (M/M/1-NONPRE node):

\[
\bar{T}_{ir}(k) = \frac{1}{\mu_{ir}} + \sum_{s=1}^{r} \frac{K_{is}(k - 1_{r})}{\mu_{is}} + \sum_{s=r+1}^{R} \frac{\rho_{is}(k - 1_{r})}{\mu_{is}},
\]

\[
1 - \sum_{s=1}^{r-1} \rho'_{is},
\]

Shadow Method for M/M/1-FCFS-PRE Nodes:

- M/M/1-FCFS-PRE-Knoten is replaced by a "shadow node" with \( R \) parallel M/M/1-FCFS nodes.

- Thus a product-form queueing network arises, which can be analyzed by MVA, SCAT or convolution.

- Since the jobs of the different job classes are processed in parallel, the service times \( s_{ir} = 1/\mu_{ir} \) have to be extended. This has to be done, that the mean response times of the shadow nodes are equal to those of the original M/M/1-FCFS-PRE nodes. This is done iteratively using an approximation formula. The initial value is \( s_{ir} \).
E.6 Approximate Analysis of Non-product-Form Queueing Networks

**Algorithm:**

1. **Step 1:** Transform the original model into the shadow model
2. **Step 2:** Set $\lambda_{i,r} = 0$
3. **Schritt 3:** Iteration
   - **Step 3.1:** Compute the utilization for each shadow node:
     \[
     \tilde{\rho}_{i,r} = \lambda_{i,r} \cdot \tilde{s}_{i,r}
     \]
   - **Step 3.2:** Compute the shadow service times:
     \[
     \tilde{s}_{i,r} = \frac{s_{i,r}}{1 - \sum_{s=1}^{r-1} \tilde{\rho}_{i,s}}
     \]

$s_{i,r}$: original service time of a class $r$ job at node $i$ job

$\tilde{s}_{i,r}$: approximated service time in the shadow node
Step 3.3: Compute the new values of the throughput $\lambda_{ir}$ of class $r$ at node $i$.

Step 4: If the $\lambda_{ir}$ differ less than $\epsilon$ in two successive iterations, then stop the iteration. Otherwise go back to Step 3.1.