Queuing Networks, MVA, Bottleneck Analysis

Advanced Systems Lab

October 31, 2019
Queuing Models in Your Project

**M/M/1:**
Use one M/M/1 queue to model your system. Build a model for each worker-thread configuration in the throughput experiment. Motivate your input parameters. Plot the calculated values and compare them with the experiment.

**M/M/m:**
Build an M/M/m model based on the throughput experiment. Treat each worker thread as one service. Motivate your input parameters. Plot the calculated values and compare them with the experiment.

**Network of Queues:**
Build a network of queues to model the baseline with middleware experiment (with one and two middlewares). Motivate the design of your model and its input parameters. Perform an analysis of each component and determine the bottleneck of your system. Compare the predictions of your model with the actual experiments.
Last Week:
We used $M/M/1$ and $M/M/m$ to model the whole system/middleware.

This week:
We build a network of queues consisting of multiple queues, each representing a component of the system.

The more detailed your network of queues is, the more accurate is the resulting model.
Last Week:
We used M/M/1 and M/M/m to model the whole system/middleware.

This week:
We build a network of queues consisting of multiple queues, each representing a component of the system. The more detailed your network of queues is, the more accurate is the resulting model.
Operational Laws

- Utilization Law
- Forced Flow Law
- Little’s Law
- Interactive Response Time Law

**Notation:**

- \( T \) Observation interval (time)
- \( C \) Number of completions
- \( A \) Number of arrivals
- \( B \) Busy time
- \( X \) Throughput
- \( S \) Service time

**Type of devices:**

- **Fixed-capacity service center**, for instance \( M/M/1 \)
- **Delay center**, no queuing, response time independent of load, infinite capacity, can also be modeled as \( M/M/\infty \)
- **Load-dependent service center**, for instance \( M/M/m \)
Utilization of a device $i$, $U_i$, can be calculated as follows:

$$U_i = \frac{B_i}{T} = \frac{C_i}{T} \ast \frac{B_i}{C_i}$$

or

$$U_i = X_i \ast S_i$$

Example:

Pizza chef bakes a pizza in:

6 min

Every hour 7 customers enter the shop and order 1 pizza each

Throughput:

7 pizzas/hour

Service time:

0.1 hour

Utilization:

$$U_i = X_i \ast S_i = 7 \ast 0.1 = 0.7$$
Utilization Law

Utilization of a device $i$, $U_i$, can be calculated as follows:

$$U_i = \frac{B_i}{T} = \frac{C_i}{T} * \frac{B_i}{C_i}$$

or

$$U_i = X_i * S_i$$

Example:
Pizza chef bakes a pizza in: 6 min
Every hour 7 customers enter the shop and order 1 pizza each
Throughput: 7 pizzas/hour
Service time: 0.1 hour
Utilization: $U_i = X_i * S_i = 7 * 0.1 = 0.7$
Forced Flow Law (I)

The Forced Flow law holds if the number of job arrivals $A_i$ at each device is the same as the number of job completions $C_i$. $\Rightarrow A_i = C_i$

$V_i$ is the visit ratio to the device $i$:

$$V_i = \frac{C_i}{C_0} \text{ or } C_i = C_0 \times V_i$$

$C_0$ is the number of jobs leaving the system
Forced Flow Law (I)

The Forced Flow law holds if the number of job arrivals $A_i$ at each device is the same as the number of job completions $C_i$. $\rightarrow A_i = C_i$

$V_i$ is the visit ratio to the device $i$:

$$V_i = \frac{C_i}{C_0} \text{ or } C_i = C_0 \times V_i$$

$C_0$ is the number of jobs leaving the system

**Example:**

$C_0$: 10

$C_{pizza} = C_0 \times V_{pizza} = 10 \times 0.7 = 7$

$C_{gelato} = C_0 \times V_{gelato} = 10 \times 0.3 = 3$
Forced Flow Law (II)

The throughput of device $i$ is:

$$X_i = \frac{C_0}{T} \cdot \frac{C_i}{C_0}$$

$$X_i = X \cdot V_i$$
Forced Flow Law (II)

The throughput of device $i$ is:

$$X_i = \frac{C_0}{T} \cdot \frac{C_i}{C_0}$$

$$X_i = X \cdot V_i$$

Combining Utilization & Forced Flow Law:

Utilization of device $i$:

$$U_i = X_i \cdot S_i = X \cdot V_i \cdot S_i$$
Forced Flow Law (II)

The throughput of device i is:

\[ X_i = \frac{C_0}{T} \times \frac{C_i}{C_0} \]

\[ X_i = X \times V_i \]

Combining Utilization & Forced Flow Law:
Utilization of device i:

\[ U_i = X_i \times S_i = X \times V_i \times S_i \]

Device with highest utilization \( U_i \) is the \textbf{bottleneck device}. 
Little’s Law

Can be applied to devices

\[ Q_i = \lambda_i \times R_i \]

if the job flow is balanced:

\[ Q_i = X_i \times R_i \]
Interactive Response Time Law

\( N \): number of users/clients  
\( Z \): think time  
Interactive law: \( R = \frac{N}{X} - Z \) or \( X = \frac{N}{R+Z} \)

Example:
There are 25 people in a bar. It takes 5min to order and receive a beer. It takes 15min to drink the beer.

\( N \): 25  
\( R \): 5min  
\( Z \): 15min

Throughput of bartender: \( X = \frac{25}{15+5} = \frac{25}{20} = 1.25 \) beers/min
Interactive Response Time Law

\[ R = \frac{N}{X} - Z \] or \[ X = \frac{N}{R+Z} \]

Example:
There are 25 people in a bar.
It takes 5 min to order and receive a beer.
It takes 15 min to drink the beer.

\[ N: 25 \]
\[ R: 5 \text{ min} \]
\[ Z: 15 \text{ min} \]

Throughput of bartender: \[ X = \frac{25}{15+5} = 1.25 \text{ beers/min} \]
Mean-Value-Analysis (MVA)

- Iteratively computes the performance for $n \in \{1, N\}$ users/clients
- Applies operational laws
- Allows you to plot graphs based on your model

Use a tool/library to compute the MVA, for instance GNU Octave:
http://www.gnu.org/software/octave/
http://www.moreno.marzolla.name/software/queueing/

The (above) algorithm only works with fix-capacity service centers and delay centers.
The example system is a **closed system**. The system implements a **Queue** data structure which supports *PUSH* and *POP* operations. To make it more scalable a middleware layer is introduced. The middleware has **for each type of request** a queue and corresponding worker(s). *PUSH* requests are forwarded to all backend servers, while *POP* requests are load-balanced among them.
1st Example: Network of Queues (I)

Modeling:
Network as delay center
Parsing thread as M/M/1 $\rightarrow V_{Parse} = 1$

PUSH worker as M/M/1 $\rightarrow V_{PUSH} = \text{pushratio}$

POP worker as M/M/1 $\rightarrow V_{POP} = 1 - V_{PUSH}$

Servers as M/M/1 $\rightarrow V_{server0} = V_{PUSH} + 1$
1st Example: Network of Queues (I)

Modeling:

- Network as delay center
- Parsing thread as M/M/1 → $V_{Parse} = 1$
- PUSH worker as M/M/1 → $V_{PUSH} = \text{pushratio}$
- POP worker as M/M/1 → $V_{POP} = 1 - V_{PUSH}$
- Servers as M/M/1 → $V_{server0} = V_{PUSH} + \frac{1}{3} V_{POP}$
1st Example: Network of Queues (II)

Parameter:
Push ratio: 0.3

Measured/Derived:
Throughput $X$ of the system: 100 req/s
Service time $S_i$ for each device

<table>
<thead>
<tr>
<th>Device</th>
<th>Throughput</th>
<th>Service Time [ms]</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>100 req/s</td>
<td>0.1</td>
<td>0.01</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>30 req/s</td>
<td>20</td>
<td>0.6</td>
</tr>
<tr>
<td>POP worker</td>
<td>70 req/s</td>
<td>13</td>
<td>0.91</td>
</tr>
<tr>
<td>Server 0</td>
<td>53 req/s</td>
<td>10</td>
<td>0.53</td>
</tr>
</tbody>
</table>
1st Example: Network of Queues (II)

Parameter:
Push ratio: 0.3

Measured/Derived:
Throughput $X$ of the system: 100 req/s
Service time $S_i$ for each device

**Calculate throughput for each device:** $X_i = X \times V_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Visit ratio</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>1</td>
<td>100 req/s</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>0.3</td>
<td>30 req/s</td>
</tr>
<tr>
<td>POP worker</td>
<td>0.7</td>
<td>70 req/s</td>
</tr>
<tr>
<td>Server 0</td>
<td>0.53</td>
<td>53 req/s</td>
</tr>
</tbody>
</table>
1st Example: Network of Queues (II)

Parameter:
Push ratio: 0.3

Measured/Derived:
Throughput $X$ of the system: 100 req/s
Service time $S_i$ for each device

Calculate throughput for each device: $X_i = X \times V_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Visit ratio</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>1</td>
<td>100 req/s</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>0.3</td>
<td>30 req/s</td>
</tr>
<tr>
<td>POP worker</td>
<td>0.7</td>
<td>70 req/s</td>
</tr>
<tr>
<td>Server 0</td>
<td>0.53</td>
<td>53 req/s</td>
</tr>
</tbody>
</table>

Calculate utilization for each device: $U_i = X_i \times S_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Throughput</th>
<th>Service Time[ms]</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>100 req/s</td>
<td>0.1</td>
<td>0.01</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>30 req/s</td>
<td>20</td>
<td>0.6</td>
</tr>
<tr>
<td>POP worker</td>
<td>70 req/s</td>
<td>13</td>
<td>0.91</td>
</tr>
<tr>
<td>Server 0</td>
<td>53 req/s</td>
<td>10</td>
<td>0.53</td>
</tr>
</tbody>
</table>

(Advanced Systems Lab) Queuing Networks, MVA, Bottleneck Analysis
Let’s change our example system and correspondingly the model by adding a second POP worker and POP queue. Now we are modeling a different system.
2nd Example: Network of Queues (I)

Modeling:
- Network as delay center
- Parsing thread as M/M/1 → \( V_{\text{Parse}} = 1 \)
- PUSH worker as M/M/1 → \( V_{\text{PUSH}} = \text{pushratio} \)
- POP workers as two M/M/1 → \( V_{\text{POP0}} = 1 - V_{\text{PUSH}} + 1 \)
- Server as M/M/1 → \( V_{\text{server0}} = V_{\text{POP1}} + 1 \)

(Advanced Systems Lab)
Modeling:

- Network as delay center
- Parsing thread as M/M/1 $\rightarrow V_{\text{Parse}} = 1$
- PUSH worker as M/M/1 $\rightarrow V_{\text{PUSH}} = \text{pushratio}$
- POP workers as two M/M/1 $\rightarrow V_{\text{POP0}} = \frac{1-V_{\text{PUSH}}}{2}$
- Servers as M/M/1 $\rightarrow V_{\text{server0}} = V_{\text{PUSH}} + \frac{1}{3} V_{\text{POP0}} + \frac{1}{3} V_{\text{POP1}}$
2nd Example: Network of Queues (II)

Parameter:
Push ratio: 0.3

Measured/Derived:
Throughput $X$ of the system: 100 req/s
Service time $S_i$ for each device

<table>
<thead>
<tr>
<th>Device</th>
<th>Throughput</th>
<th>Service Time [ms]</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>100 req/s</td>
<td>0.1</td>
<td>0.01</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>30 req/s</td>
<td>20</td>
<td>0.6</td>
</tr>
<tr>
<td>POP 0 worker</td>
<td>35 req/s</td>
<td>13</td>
<td>0.45</td>
</tr>
<tr>
<td>Server 0</td>
<td>53 req/s</td>
<td>10</td>
<td>0.53</td>
</tr>
</tbody>
</table>

The system is not saturated.
2nd Example: Network of Queues (II)

Parameter: Push ratio: 0.3

Measured/Derived: Throughput $X$ of the system: 100 req/s
Service time $S_i$ for each device

Calculate throughput for each device: $X_i = X \times V_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Visit ratio</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>1</td>
<td>100 req/s</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>0.3</td>
<td>30 req/s</td>
</tr>
<tr>
<td>POP 0 worker</td>
<td>0.35</td>
<td>35 req/s</td>
</tr>
<tr>
<td>Server 0</td>
<td>0.53</td>
<td>53 req/s</td>
</tr>
</tbody>
</table>

The system is not saturated.
2nd Example: Network of Queues (II)

Parameter: 
Push ratio: 0.3

Measured/Derived:
Throughput $X$ of the system: 100 req/s
Service time $S_i$ for each device

Calculate throughput for each device: $X_i = X \times V_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Visit ratio</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>1</td>
<td>100 req/s</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>0.3</td>
<td>30 req/s</td>
</tr>
<tr>
<td>POP 0 worker</td>
<td>0.35</td>
<td>35 req/s</td>
</tr>
<tr>
<td>Server 0</td>
<td>0.53</td>
<td>53 req/s</td>
</tr>
</tbody>
</table>

Calculate utilization for each device: $U_i = X_i \times S_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Throughput</th>
<th>Service Time[ms]</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>100 req/s</td>
<td>0.1</td>
<td>0.01</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>30 req/s</td>
<td>20</td>
<td>0.6</td>
</tr>
<tr>
<td>POP 0 worker</td>
<td>35 req/s</td>
<td>13</td>
<td>0.45</td>
</tr>
<tr>
<td>Server 0</td>
<td>53 req/s</td>
<td>10</td>
<td>0.53</td>
</tr>
</tbody>
</table>
2nd Example: Network of Queues (II)

Parameter: Push ratio: 0.3

Measured/Derived: Throughput $X$ of the system: 100 req/s

Service time $S_i$ for each device

Calculate throughput for each device: $X_i = X \times V_i$

### Device | Visit ratio | Throughput
--- | --- | ---
Parser | 1 | 100 req/s
PUSH worker | 0.3 | 30 req/s
POP 0 worker | 0.35 | 35 req/s
Server 0 | 0.53 | 53 req/s

Calculate utilization for each device: $U_i = X_i \times S_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Throughput</th>
<th>Service Time[ms]</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>100 req/s</td>
<td>0.1</td>
<td>0.01</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>30 req/s</td>
<td>20</td>
<td>0.6</td>
</tr>
<tr>
<td>POP 0 worker</td>
<td>35 req/s</td>
<td>13</td>
<td>0.45</td>
</tr>
<tr>
<td>Server 0</td>
<td>53 req/s</td>
<td>10</td>
<td>0.53</td>
</tr>
</tbody>
</table>

The system is not saturated.
Let’s adapt our example system again. Now the two POP workers are going to share the same POP queue. This is again a different system.
3rd Example: Network of Queues (I)

Modeling:
- Network as delay center
- Parsing thread as M/M/1 → \( V_{\text{Parse}} = 1 \)
- PUSH worker as M/M/1 → \( V_{\text{PUSH}} = \text{pushratio} \)
- POP workers as M/M/2 → \( V_{\text{POP}} = 1 \)
- Servers as M/M/1 → \( V_{\text{server}0} = \) pushratio + 1

Queuing Networks, MVA, Bottleneck Analysis
October 31, 2019
3rd Example: Network of Queues (I)

Modeling:
- Network as delay center
- Parsing thread as $M/M/1 \rightarrow V_{Parse} = 1$
- PUSH worker as $M/M/1 \rightarrow V_{PUSH} = \text{pushratio}$
- POP workers as $M/M/2 \rightarrow V_{POP} = 1 - V_{PUSH}$
- Servers as $M/M/1 \rightarrow V_{server0} = V_{PUSH} + \frac{1}{3} V_{POP}$
3rd Example: Network of Queues (II)

Parameter:
Push ratio: 0.3

Measured/Derived:
Throughput $X$ of the system: 100 req/s
Service time $S_i$ for each device

Calculate throughput for each device:

Device | Throughput | Visit ratio |
--- | --- | --- |
Parser | 100 req/s | 1 |
PUSH worker | 30 req/s | 0.3 |
POP workers | 70 req/s | 0.7 |
Server 0 | 53 req/s | 0.53 |

Calculate utilization for each device:

Device | Throughput | Service Time [ms] | Utilization |
--- | --- | --- | --- |
Parser | 100 req/s | 0.1 ms | 0.01 |
PUSH worker | 30 req/s | 20 ms | 0.6 |
POP workers | 70 req/s | 13 ms | 0.91 |
Server 0 | 53 req/s | 10 ms | 0.53 |

M/M/2 cannot be modelled as fixed-capacity device, use extended MVA instead.
3rd Example: Network of Queues (II)

Parameter:
Push ratio: 0.3

Measured/Derived:
Throughput $X$ of the system: 100 req/s
Service time $S_i$ for each device

Calculate throughput for each device: $X_i = X \times V_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Visit ratio</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>1</td>
<td>100 req/s</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>0.3</td>
<td>30 req/s</td>
</tr>
<tr>
<td>POP workers</td>
<td>0.7</td>
<td>70 req/s</td>
</tr>
<tr>
<td>Server 0</td>
<td>0.53</td>
<td>53 req/s</td>
</tr>
</tbody>
</table>

M/M/2 cannot be modelled as fixed-capacity device, use extended MVA instead.
3rd Example: Network of Queues (II)

Parameter: Push ratio: 0.3

Measured/Derived: Throughput $X$ of the system: 100 req/s
Service time $S_i$ for each device

Calculate throughput for each device: $X_i = X \times V_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Visit ratio</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>1</td>
<td>100 req/s</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>0.3</td>
<td>30 req/s</td>
</tr>
<tr>
<td>POP workers</td>
<td>0.7</td>
<td>70 req/s</td>
</tr>
<tr>
<td>Server 0</td>
<td>0.53</td>
<td>53 req/s</td>
</tr>
</tbody>
</table>

Calculate utilization for each device: $U_i = X_i \times S_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Throughput</th>
<th>Service Time[ms]</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>100 req/s</td>
<td>0.1</td>
<td>0.01</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>30 req/s</td>
<td>20</td>
<td>0.6</td>
</tr>
<tr>
<td>POP workers</td>
<td>70 req/s</td>
<td>13</td>
<td>0.91</td>
</tr>
<tr>
<td>Server 0</td>
<td>53 req/s</td>
<td>10</td>
<td>0.53</td>
</tr>
</tbody>
</table>
3rd Example: Network of Queues (II)

Parameter: Push ratio: 0.3

Measured/Derived: Throughput $X$ of the system: 100 req/s
Service time $S_i$ for each device

Calculate throughput for each device: $X_i = X \times V_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Visit ratio</th>
<th>Throughput</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>1</td>
<td>100 req/s</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>0.3</td>
<td>30 req/s</td>
</tr>
<tr>
<td>POP workers</td>
<td>0.7</td>
<td>70 req/s</td>
</tr>
<tr>
<td>Server 0</td>
<td>0.53</td>
<td>53 req/s</td>
</tr>
</tbody>
</table>

Calculate utilization for each device: $U_i = X_i \times S_i$

<table>
<thead>
<tr>
<th>Device</th>
<th>Throughput</th>
<th>Service time[ms]</th>
<th>Utilization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parser</td>
<td>100 req/s</td>
<td>0.1</td>
<td>0.01</td>
</tr>
<tr>
<td>PUSH worker</td>
<td>30 req/s</td>
<td>20</td>
<td>0.6</td>
</tr>
<tr>
<td>POP workers</td>
<td>70 req/s</td>
<td>13</td>
<td>0.91</td>
</tr>
<tr>
<td>Server 0</td>
<td>53 req/s</td>
<td>10</td>
<td>0.53</td>
</tr>
</tbody>
</table>

M/M/2 cannot be modelled as fixed-capacity device, use extended MVA instead.
Lesson learned

In the previous example we saw the limitations of the operation laws and the basic MVA algorithm. As we have seen last week: n M/M/1 are not equal to one M/M/n. Depending on your actual system one of them is more suitable.

M/M/m devices are load-dependent service centers which means their service rate depends on the number of jobs in the system. They cannot be modeled using the Operational Laws. If you want to model your network of queues with M/M/m devices use the extended MVA algorithm (see book page 610).
Office hours

Office hours are indented to provide you advice that will help you to complete the project and the report. To make an appointment, **contact your teaching assistant by email**.

- Make sure you come prepared with concrete and well formulated questions. If possible, include them in your email.
- We will not complete the assignment for you and neither recommend nor make design decisions on your behalf.
- We will not debug your code, provide technical support for your setup/scripts/data analysis, or give hints about whether what you have done so far is enough.
- We will not grade your project in advance, so please avoid questions that try to determine whether what you have done is correct or sufficient for a passing grade.

**Days & Times can be found on the website.**