Queuing Networks, MVA, Bottleneck Analysis

Advanced Systems Lab

November 16, 2017

Network of Queues

Last Week:
We used M/M/1 and M/M/m to model the whole system/middleware.

Network of Queues

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

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 Law

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

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

or

$$U_i = X_i * S_i$$

Utilization Law

Utilization of a device i, U_i, can be calculate 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: **6min**
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:

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

C_0 is the number of jobs leaving the system

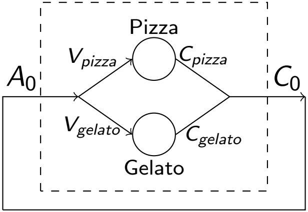
Forced Flow Law (I)

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:

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

C_0 is the number of jobs leaving the system

Example:
C_0: 10
$C_{pizza} = C_0 * V_{pizza} = 10 * 0.7 = 7$
$C_{gelato} = C_0 * V_{gelato} = 10 * 0.3 = 3$



Forced Flow Law (II)

The throughput of device i is:

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

$$X_i = X * V_i$$

Forced Flow Law (II)

The throughput of device i is:

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

$$X_i = X * V_i$$

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

$$U_i = X_i * S_i = X * V_i * S_i$$

Forced Flow Law (II)

The throughput of device i is:

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

$$X_i = X * V_i$$

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

$$U_i = X_i * S_i = X * V_i * S_i$$

Device with highest utilization U_i is the **bottleneck device**.

Little's Law

Can be applied to devices

$$Q_i = \lambda_i * R_i$$

if the job flow is balanced:

$$Q_i = X_i * R_i$$

Interactive Response Time Law

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

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: $X = \frac{25}{15+5} = 1.25$ beers/min

Mean-Value-Analysis (MVA)

MEAN-VALUE ANALYSIS 577

Box 34.2 MVA Algorithm

Inputs:
N = number of users
Z = think time
M = number of devices (not including terminals)
S_i = service time per visit to the ith device
V_i = number of visits to the ith device

Outputs:
X = system throughput
Q_i = average number of jobs at the ith device
R_i = response time of the ith device
R = system response time
U_i = utilization of the ith device

Initialization: FOR $i = 1$ TO M DO $Q_i = 0$

Iterations:
FOR $n = 1$ TO N DO
BEGIN

FOR $i = 1$ TO M DO $R_i = \begin{cases} S_i(1+Q_i) & \text{Fixed capacity} \\ S_i & \text{Delay centers} \end{cases}$

$$R = \sum_{i=1}^{M} R_i V_i$$

$$X = \frac{N}{Z+R}$$

FOR $i = 1$ TO M DO $Q_i = X V_i R_i$

END

Device throughputs: $X_i = XV_i$
Device utilizations: $U_i = XS_iV_i$

The (above) algorithm only works with fix-capacity service centers and delay centers.

- 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:

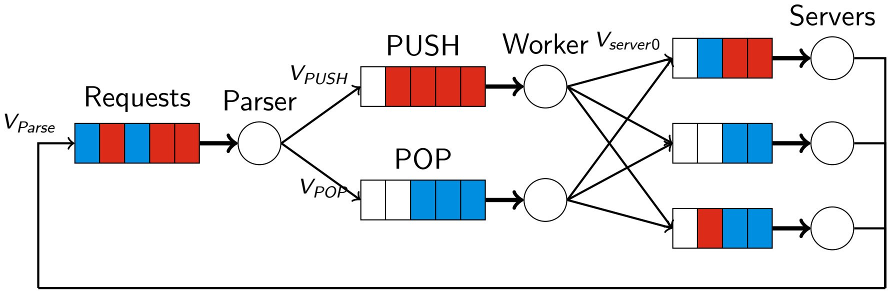
http://www.gnu.org/software/octave/

http://www.moreno.marzolla.name/software/queueing/

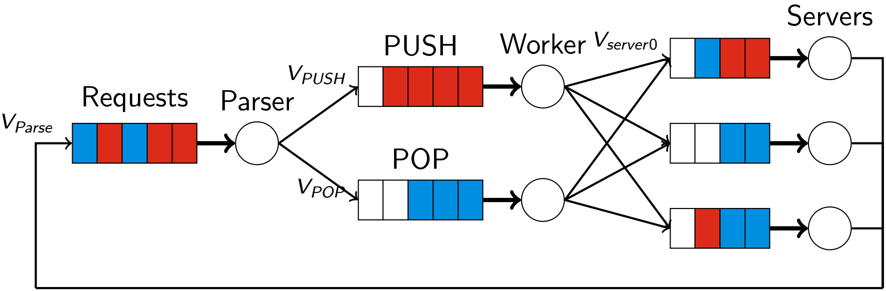
Network of Queues: Example System

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)



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 → $V_{PUSH} = pushratio$
- 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

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: $X_i = X * V_i$

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

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: $X_i = X * V_i$

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

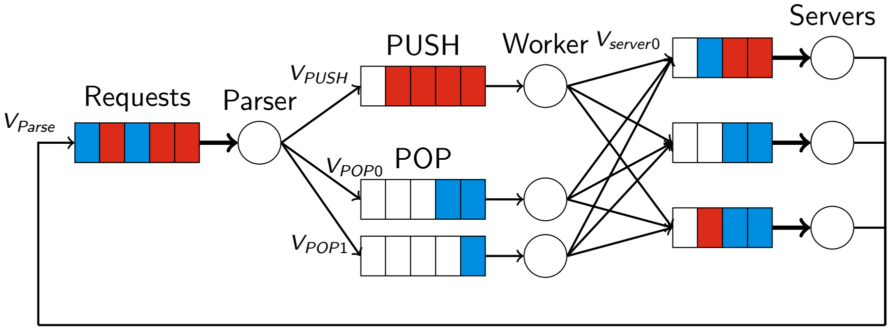
Calculate utilization for each device: $U_i = X_i * S_i$

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

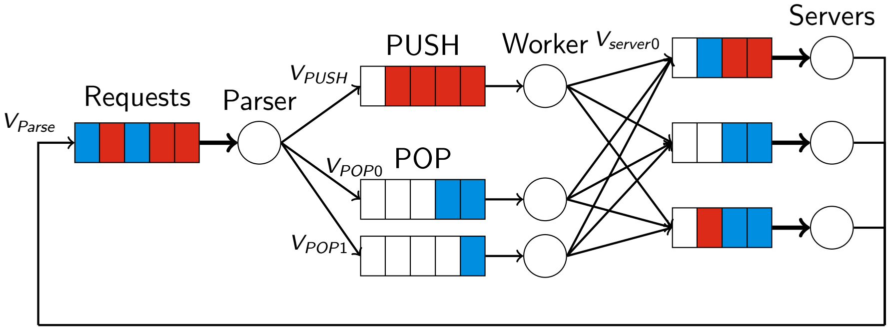
Network of Queues: 2nd Example System

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)



2nd 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 $\rightarrow V_{PUSH} = pushratio$
- POP workers as two M/M/1 $\rightarrow V_{POP0} = \frac{1-V_{PUSH}}{2}$
- Servers as M/M/1 $\rightarrow V_{server0} = V_{PUSH} + \frac{1}{3}V_{POP0} + \frac{1}{3}V_{POP1}$

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

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: $X_i = X * V_i$

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

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: $X_i = X * V_i$

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

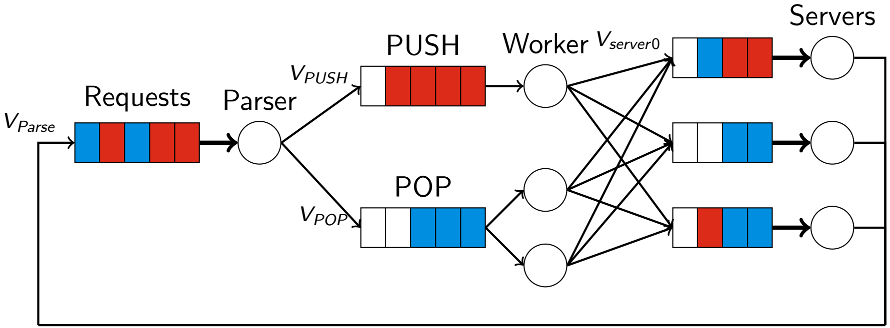
Calculate utilization for each device: $U_i = X_i * S_i$

Device	Throughput	Service Time[ms]	Utilization
Parser	100 req/s	0.1	0.01
PUSH worker	30 req/s	20	0.6
POP 0 worker	35 req/s	13	**0.45**
Server 0	53 req/s	10	**0.53**

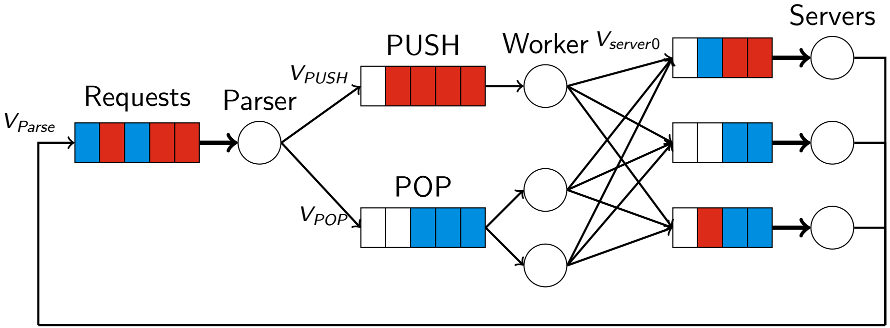
Network of Queues: 3rd Example System

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)



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 $\rightarrow V_{PUSH} = pushratio$
- 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

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: $X_i = X * V_i$

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

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: $X_i = X * V_i$

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

Calculate utilization for each device: $U_i = X_i * S_i$

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

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: $X_i = X * V_i$

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

Calculate utilization for each device: $U_i = X_i * S_i$

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

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 can be modeled with extended MVA.*

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.

Azure spending

The spending is now correctly reported on `https://www.microsoftazuresponsorships.com`, please check it there.