Value functions for M/G/1 & Task Assignment Problem

Esa Hyytiä

Joint work with

Samuli Aalto, Aleksi Penttinen, Jorma Virtamo

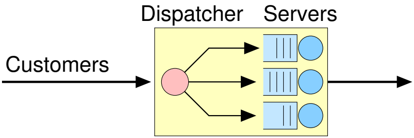
Department of Communications and Networking
Aalto University, School of Electrical Engineering, Finland

YEQT-VI, November 1-3, 2012

Aalto University
School of Electrical Engineering

Outline

Aalto University
School of Electrical Engineering



Latency $\mathrm{E}[\mathbf{T}]$:

- Sojourn time, Response Time, Delay, ...
- Objective:

$$\boxed{\min\ \mathrm{E}[T].}$$

Slowdown: "*long jobs can wait longer*"

- Slowdown of job i, $\gamma_i \triangleq \frac{\text{Latency } T_i}{\text{Service time } X_i}.$
- Objective:

$$\boxed{\min\ \mathrm{E}[\gamma].}$$

Aalto University
School of Electrical Engineering

Server Systems: Holding Cost Structure

Holding cost:

- Job i accrues costs at job-specific rate b_i

Latency: With $b_i = 1$,

- Total cost rate is the number of jobs in the system, N_t
- Cost a job incurs is equal to the latency,

$$b_i \cdot T_i = T_i.$$

Slowdown: With $b_i = 1/x_i$

- Cost a job incurs is equal to the slowdown,

$$b_i \cdot T_i = \frac{T_i}{x_i}.$$

Note: No costs associated with state transitions

Aalto University
School of Electrical Engineering

Value Function: Definition

- Let $C_{\mathbf{z}}(t)$ denote the cost rate at time t for an initial state $\mathbf{z}$
- Cumulative costs accrued during $(0, t)$ are

$$V_{\mathbf{z}}(t) \triangleq \int_0^t C_{\mathbf{z}}(s)\, ds.$$

- Relative value is the expected difference in the infinite horizon cumulative costs between
  a) a system initially in state $\mathbf{z}$, and
  b) a system initially in equilibrium,

$$\boxed{v_{\mathbf{z}} \triangleq \lim_{t\to\infty} \mathrm{E}[V_{\mathbf{z}}(t) - r\, t].}$$

Aalto University
School of Electrical Engineering

Value function: Latency in Server Systems

- For latency, the cost rate $C_{\mathbf{z}}(t)$ is simply

$$N_{\mathbf{z}}(t) \triangleq \text{"the number of jobs in the system"},$$

- Value function reads

$$v_{\mathbf{z}} = \lim_{t\to\infty} \left(\mathrm{E}\Big[\int_0^t N_{\mathbf{z}}(s)\, ds \Big] - \mathrm{E}[N]\, t \right).$$

- Similarly for the slowdown and general holding costs

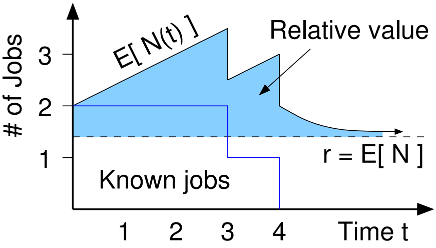
Aalto University
School of Electrical Engineering

Initial state $\mathbf{z} = (3, 1)$:

- First job with remaining size 3 currently receiving service
- Second job with size 1 is waiting
- Also later arriving jobs have to wait (FCFS)

Relative value of state **z** is the expected difference in infinite horizon costs:

$v_{\mathbf{z}}$ = blue shaded area.



Value function: Comparison of States

Given two states $\mathbf{z}_1$ and $\mathbf{z}_2$, the expected difference in the infinite horizon costs is

$$d(\mathbf{z}_1, \mathbf{z}_2) = \lim_{t \to \infty} \mathrm{E}[V_{\mathbf{z}_2}(t) - V_{\mathbf{z}_1}(t)],$$

which gives

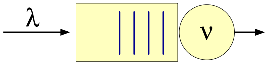
$$d(\mathbf{z}_1, \mathbf{z}_2) = v_{\mathbf{z}_2} - v_{\mathbf{z}_1}.$$

Example: Server system

- Suppose state $\mathbf{z}_2$ is state $\mathbf{z}_1$ plus one new job
- Value function gives the marginal cost for accepting a new job!

Aalto University
School of Electrical Engineering

Value Function for M/G/1 Queues



A. Elementary scheduling disciplines:
- M/G/1-FCFS
- M/G/1-LCFS

B. Size-aware scheduling disciplines:
- M/G/1-SPT (shortest-processing-time)
- M/G/1-SRPT (shortert-remaining-processing-time)
- M/G/1-SPTP (shortest-processing-time-product)

C. Processor sharing (PS)
- M/D/1-PS (fixed job sizes)
- M/M/1-PS

Aalto University
School of Electrical Engineering

Basic case:

- Poisson arrival rate λ
- Service times X_i i.i.d., $X_i \sim X$
- Offered load $\rho = \lambda \, \mathrm{E}[X]$
- Size-aware state $\mathbf{z} = (\Delta_1; ..; \Delta_n)$ with n jobs:
  - Δ_i is the remaining service time of job i
  - Job n is served first (FCFS,LCFS)
- Backlog $u_{\mathbf{z}} = \sum_i \Delta_i$

With arbitrary holding costs:

- State $\mathbf{z} = ((\Delta_1, b_1); ..; (\Delta_n, b_n))$
  b_i is the holding cost of job i
- $\mathrm{E}[B]$ is the mean holding cost (arbitrary job)

Aalto University
School of Electrical
Engineering

Proposition: The size-aware relative value of state **z** with respect to delay in an M/G/1-FCFS queue is[1,2]

$$v_{\mathbf{z}} - v_0 = \sum_{i=1}^{n} i\,\Delta_i + \frac{\lambda\, u_{\mathbf{z}}^2}{2(1-\rho)}. \tag{1}$$

With respect to arbitrary job specific holding costs b_i,

$$v_{\mathbf{z}} - v_0 = \sum_{i=1}^{n} \left(\Delta_i \sum_{j=1}^{i} b_j \right) + \frac{\lambda\, u_{\mathbf{z}}^2}{2\,(1-\rho)} \mathrm{E}[B]. \tag{2}$$

Note: Insensitive to service time distribution.

[1] Hyytiä et al., Eur. J. Oper. Research (2012)

[2] Hyytiä et al., J. Applied Probability (2012).

Aalto University
School of Electrical Engineering

M/G/1-LCFS (preemptive)

Proposition: The size-aware relative value of state $\mathbf{z}$ with respect to delay in an M/G/1-LCFS queue is[3,4]

$$v_{\mathbf{z}} - v_0 = \frac{1}{1-\rho} \sum_{i=1}^{n} i \cdot \Delta_i. \tag{3}$$

With respect to arbitrary job specific holding costs b_i,

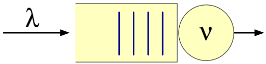
$$v_{\mathbf{z}} - v_0 = \frac{1}{1-\rho} \sum_{i=1}^{n} \left(\Delta_i \sum_{j=1}^{i} b_j \right). \tag{4}$$

Note:
- Later arrivals immune to state $\mathbf{z}$.
- Insensitivity: $v_{\mathbf{z}} - v_0$ depends only on ρ.

[3] Hyytiä et al., Eur. J. Oper. Research (2012)
[4] Hyytiä et al., J. Applied Probability (2012).

Aalto University
School of Electrical Engineering

Value Function for M/G/1 Queues



A. Elementary scheduling disciplines:
- M/G/1-FCFS
- M/G/1-LCFS

B. Size-aware scheduling disciplines:
- M/G/1-SPT (shortest-processing-time)
- M/G/1-SRPT (shortert-remaining-processing-time)
- M/G/1-SPTP (shortest-processing-time-product)

C. Processor sharing (PS)
- M/D/1-PS (fixed job sizes)
- M/M/1-PS

Aalto University
School of Electrical Engineering

Notation: (Δ_i, Δ_i^*) = remaining and initial service time of job i.

Index policy α serves first the job with the lowest index.

Scheduling	Index	Optimality
SPT	Δ_i^*	optimal non-preemptive / delay & slowdown
SRPT	Δ_i	optimal preemptive / delay
SPTP	$\Delta_i \cdot \Delta_i^*$	optimal preemptive / slowdown[5]

[5] Hyytiä, Aalto, Penttinen, SIGMETRICS'12.

Aalto University
School of Electrical Engineering

Size-aware M/G/1: Scheduling

	non-preemptive		preemptive	
	class-aware	size-aware	non-anticipating	anticipating size-aware
delay	SEPT ($c\mu$-rule)	SPT	FB, FIFO,... (depends on $f(x)$)	SRPT
slowdown	-”-	-”-	FB, FIFO, ... (depends on $f(x)$)	SPTP (M/G/1)

Aalto University
School of Electrical Engineering

Size-aware M/G/1: Additional notation

Notation:

- Jobs are numbered so that (without new arrivals) job 1 is served first and job n last.
- $f(x)$ denotes the service time pdf.
- $\rho(x)$ denotes the load due to jobs shorter than x,

$$\rho(x) = \lambda \int_0^x x\, f(x)\, dx.$$

- Define

$$h(x) \triangleq \frac{f(x)\, b(x)}{(1-\rho(x))^2},$$

where $b(x)$ is the mean holding cost of a job with size x,

$$b(x) = \mathrm{E}[B\,|\,X = x]$$

Aalto University
School of Electrical Engineering

Proposition: The size-aware relative value of state **z** with respect to arbitrary holding costs in an M/G/1-SPT queue is[6]

$$v_{\mathbf{z}} - v_0 = \sum_{i=1}^{n} b_i \left(\Delta_i + \frac{1}{1-\rho(\Delta_i)} \left(\sum_{j=1}^{i-1} \Delta_j \right) \right) + \frac{\lambda}{2} \sum_{i=1}^{n} \left[\left(\sum_{j=i+1}^{n} \Delta_j^2 + \left(\sum_{j=1}^{i} \Delta_j \right)^2 \right) \int_{\tilde{\Delta}_i}^{\tilde{\Delta}_{i+1}} h(x)\, dx \right] \quad (5)$$

where

- job 1 receives service and $\Delta_2 < \ldots < \Delta_n$
- $\tilde{\Delta}_i = \begin{cases} 0, & i = 1, \\ \Delta_i, & i = 2, \ldots, n \\ \infty & i = n+1. \end{cases}$

[6] Hyytiä et al., Eur. J. Oper. Research (2012)

Aalto University
School of Electrical Engineering

Proposition: The size-aware relative value of state **z** with respect to arbitrary holding costs in an M/G/1-SRPT queue is[7]

$$v_{\mathbf{z}} - v_0 = \sum_{i=1}^{n} b_i \left(\frac{1}{1-\rho(\Delta_i)} \left(\sum_{j=1}^{i-1} \Delta_j \right) + \int_0^{\Delta_i} \frac{1}{1-\rho(x)}\, dx \right) + \frac{\lambda}{2} \sum_{i=0}^{n} \left[\left(\sum_{j=1}^{i} \Delta_j \right)^2 \int_{\Delta_i}^{\Delta_{i+1}} h(x)\, dx + (n-i) \int_{\Delta_i}^{\Delta_{i+1}} x^2\, h(x)\, dx \right] \tag{6}$$

where

- job 1 receives currently service and $\Delta_1 < \ldots < \Delta_n$,
- $\Delta_0 = 0$ and $\Delta_{n+1} = \infty$

[7] Hyytiä et al., Eur. J. Oper. Research (2012)

Aalto University
School of Electrical Engineering

Proposition: The size-aware relative value of state **z** with respect to arbitrary holding costs in an M/G/1-SPTP queue is[8]

$$v_{\mathbf{z}} - v_0 = \sum_{i=1}^{n} b_i \left(\frac{1}{1-\rho(\tilde{\Delta}_i)} \left(\sum_{j=1}^{i-1} \Delta_j \right) + \frac{2}{\Delta_i^*} \int_0^{\tilde{\Delta}_i} \frac{x\,dx}{1-\rho(x)} \right)$$
$$+ \frac{\lambda}{2} \sum_{i=0}^{n} \left[\left(\sum_{j=1}^{i} \Delta_j \right)^2 \int_{\tilde{\Delta}_i}^{\tilde{\Delta}_{i+1}} h(x)\,dx + \left(\sum_{j=i+1}^{n} (\Delta_j^*)^{-2} \right) \int_{\tilde{\Delta}_i}^{\tilde{\Delta}_{i+1}} x^4\, h(x)\,dx \right]$$

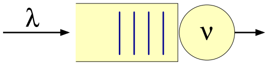
where

- Job 1 receives service and $\sqrt{\Delta_1 \Delta_1^*} < \ldots < \sqrt{\Delta_n \Delta_n^*}$ (SPTP)
- $\tilde{\Delta}_i = \begin{cases} 0, & i = 0 \\ \sqrt{\Delta_i \Delta_i^*}, & i = 1, \ldots, n \\ \infty, & i = n+1. \end{cases}$

[8] Hyytiä, Aalto, Penttinen, SIGMETRICS'12.

Aalto University
School of Electrical Engineering

Value Function for M/G/1 Queues



A. Elementary scheduling disciplines:
- M/G/1-FCFS
- M/G/1-LCFS

B. Size-aware scheduling disciplines:
- M/G/1-SPT (shortest-processing-time)
- M/G/1-SRPT (shortert-remaining-processing-time)
- M/G/1-SPTP (shortest-processing-time-product)

C. Processor sharing (PS)
- M/D/1-PS (fixed job sizes)
- M/M/1-PS

Aalto University
School of Electrical Engineering

M/G/1-PS: (Processor sharing)

Basics:

- PS serves the existing n jobs at equal rates $1/n$.
- Mean delay in M/G/1-PS is insensitive to job size distribution,

$$\mathrm{E}[T] = \frac{\mathrm{E}[X]}{1-\rho}.$$

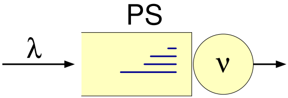
- Unfortunately, the size-aware value function is not!
- $(\Delta_1; ..; \Delta_n)$ denotes the remaining service times, $\Delta_1 \geq \ldots \geq \Delta_n$.

Without new arrivals:

- Job n leaves the system first and job 1 last
- Cumulative delay (myopic cost) is given by

$$\begin{aligned} V_{\mathbf{z}} &= \Delta_n n^2 + (\Delta_{n-1} - \Delta_n)(n-1)^2 + \ldots + (\Delta_1 - \Delta_2) \\ &= \sum_{i=1}^{n} (2i-1)\Delta_i. \end{aligned} \tag{7}$$

Aalto University
School of Electrical Engineering



Proposition: The size-aware relative value of state **z** with respect to the delay in an M/D/1-PS queue is given by[9]

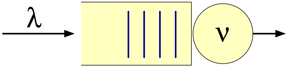
$$v_{(\Delta_1;..;\Delta_n)} - v_0 = \frac{\lambda}{1-\rho} u_{\mathbf{z}}^2 - u_{\mathbf{z}} + 2 \sum_{i=1}^{n} i\, \Delta_i. \tag{8}$$

Note:

- Compact form as a new job will always depart last.
- Converges to (7) when $\lambda \to 0$

[9] Hyytiä et al., ITC'11.

Aalto University
School of Electrical Engineering



Consider:

- M/M/1 queue
- any work conserving scheduling (FCFS, LCFS, PS, . . .)
- number-aware system: number of jobs m is known

Lemma: The value function for a work conserving and number-aware M/M/1 queue is[10]

$$\boxed{v_m = \frac{1}{2} \cdot \frac{m(m+1)}{\mu - \lambda} - \frac{\lambda\mu}{(\mu-\lambda)^3}.} \qquad (9)$$

[10] Aalto and Virtamo (1996), Virtamo (Lecture slides, 2004). The constant term follows from the identity $\sum_i \pi_i v_i = 0$.

Proposition: The relative value of state $(m; \Delta_1, \ldots, \Delta_n)$ in a size-aware M/M/1-PS queue is given by[11]

$$v_{(m;\Delta_1,\ldots,\Delta_n)} = v_m + \frac{1}{(1-\rho)^2}\sum_{k=1}^{n}(2k-1)\Delta_k + \frac{2-\rho}{\mu(1-\rho)^2}\sum_{k=1}^{n}\left(m-\frac{k\rho}{1-\rho}\right)\left(\sum_{i=1}^{k} e^{-\mu(1-\rho)(\Delta_i-\Delta_k)}\right)\left(1-e^{-\mu(1-\rho)(\Delta_k-\Delta_{k+1})}\right)$$

where

- Δ_i are n known remaining service times, $\Delta_1 > \ldots > \Delta_n$,
- m tasks have unknown Exp(μ) distributed service time,
- and $\Delta_{n+1} \triangleq 0$.

Note: Converges to (7) when $m = 0$ and $\lambda \to 0$.

[11] Hyytiä et al., Performance 2011.

Aalto University
School of Electrical Engineering

Related results for value functions

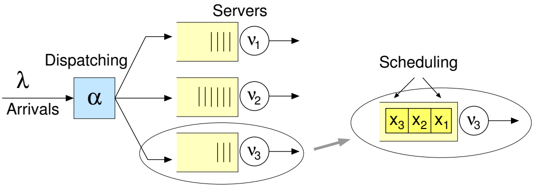
Queueing systems:

M/M/s:	Krishnan, CDC'87
M/G/1-FCFS (approx.)	Sassen et al. Neerlandica (1997)
M/M/1 & M/M/1/N (deviation matrix)	Koole, CDC'98
M/M/1 (FCFS/LCFS/PS)	Aalto&Virtamo, NTS-13 (1996); and Virtamo, Lecture notes on MDP (2004)
M/Cox(r)/1	Bhulai, J. Applied Prob. (2006)

Blocking systems:

M/M/s/s	Krishnan, CDC'86
M/M/s/k	Leeuwaarden et al. (2001)

Aalto University
School of Electrical Engineering



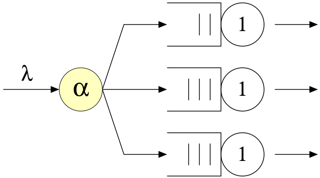
Task assignment (dispatching):

Route job to one of the m servers upon arrival.

Examples:

1. Manufacturing sites
2. Job assignment in supercomputing,
3. Data traffic routing
4. Web-server farms and Data centers
5. Other distributed computing systems . . .

Aalto University
School of Electrical Engineering



Model:

- Poisson arrival process, rate λ
- *m* parallel heterogeneous servers
- General job size distribution
- Service requirements become known upon arrival (possibly server specific)
- Queue states are known (job sizes and their service order)
- Scheduling discipline known: FCFS, LCFS, SRPT ...

Aalto University
School of Electrical Engineering

State-independent Policies

Definition:

State-independent policy chooses the server independently of the queue states.

1. **Bernoulli splitting (RND)**:
   Choose queue in random using probabilities p_i

2. **Size-Interval-Task-Assignment (SITA)**:
   "short jobs to one queue and rest to another"
   - Proposed in Crovella et. al (Sigmetrics'98) and Harchol-Balter et. al (J. of PDC, vol. 59, 1999).
   - SITA-E uses such intervals that balance the load.
   - Optimal size-aware state-free for FCFS (Feng et. al, 2005).

Aalto University
School of Electrical Engineering

1. **Join-the-Shortest-Queue (JSQ)**:
   Optimal when Poisson arrivals, Exp-distributed job sizes, identical servers, and only the queue occupancy is known (Winston, 1977).
2. **Round-robin (RR)**:
   Optimal with identical servers that were initially in a same state (Ephremides et. al, 1980).
3. **Least-Work-Left (LWL)**:
   Pick the queue with the shortest backlog (Sharifnia, 1997).

Aalto University
School of Electrical Engineering

Approach: First Policy Iteration (FPI)

- Size- and state-aware setting; future arrivals not known
- Idea: start with a reasonable basic dispatching policy, and carry out the first policy iteration (FPI) step
- Policy iteration finds the optimal policy; the first step typically yields the highest improvement.
- Requires the relative values of states $v_{\mathbf{z}}$

Aalto University
School of Electrical Engineering

First Policy Iteration (FPI)

- Assume: Relative values $v_{\mathbf{z}}$ available (for basic policy)
- Improved decision according to FPI at state $\mathbf{z}$:

$$\alpha(\mathbf{z}, x) \triangleq \underset{i}{\operatorname{argmin}} \left(v_{\mathbf{z}'(i)} - v_{\mathbf{z}}\right),$$

where $\mathbf{z}'(i)$ is the new state if job x is added to queue i.

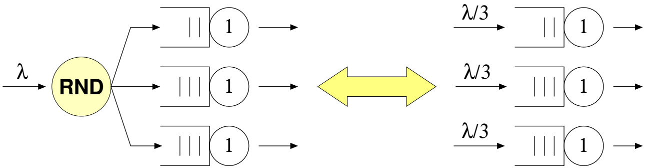
"Choose the action with the smallest expected future cost"

- Recall: in addition to $\mathbf{z}$, relative value $v_{\mathbf{z}}$ depends also on
  1. Basic dispatching policy
  2. Scheduling discipline
  3. Arrival rate λ, and
  4. Job size distribution.

Aalto University
School of Electrical Engineering

Decomposition to Independent M/G/1 Queues

- Deriving value function is generally difficult.
- However, any state-independent policy feeds each server jobs according to a Poisson process (cf. Bernoulli split)



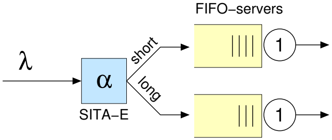
- System's value function is the sum of the queue specific value function:

$$v_{\mathbf{z}} = \sum_i v_{\mathbf{z}_i}.$$

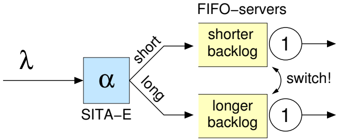
- Sufficient to analyze single M/G/1 queues instead!

Aalto University
School of Electrical Engineering

SITA-E



SITA-Es (switch)



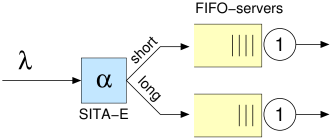
- Jobs shorter than *y* to Queue 1
- The rest to Queue 2
- Adjust *y* to balance the load
- Poisson arrivals
- Behaves as two independent M/G/1-FCFS queues

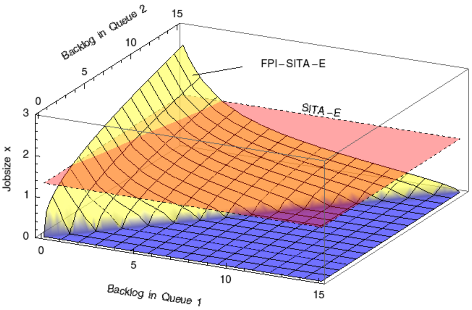
- Identical servers
- Roles can be swapped
- New initial state, same system otherwise
- Value function $v_{\mathbf{z}_1} + v_{\mathbf{z}_2}$ $\Rightarrow$ optimal permutation of roles

SITA-Es: “Short jobs to short queue, and long to long.”

Aalto University
School of Electrical Engineering

FPI-SITA-E, “Dynamic SITA-E”





- SITA-E uses a fixed threshold for separating the short jobs from the long jobs.
- FPI gives a new policy, FPI-SITA-E
- With FPI-SITA-E, the threshold is dynamically adjusted based on the current backlog in the queues.

Aalto University
School of Electrical Engineering

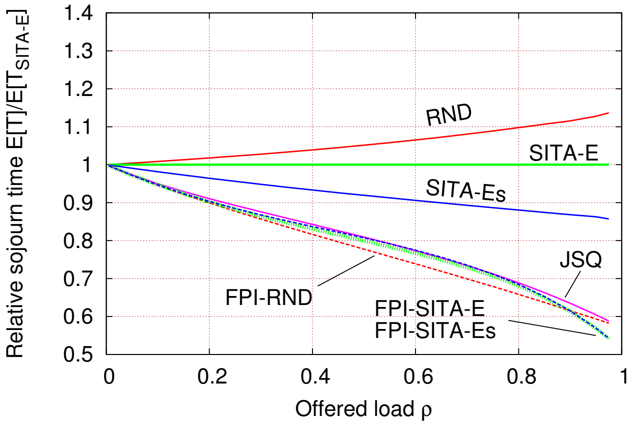
For Delay:

1. Two identical FCFS servers
2. Two identical SRPT servers
3. Heterogeneous PS servers

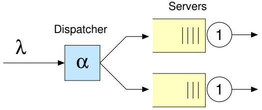
For Slowdown:

1. Three heterogeneous servers

FCFS with Uniformly distributed jobs

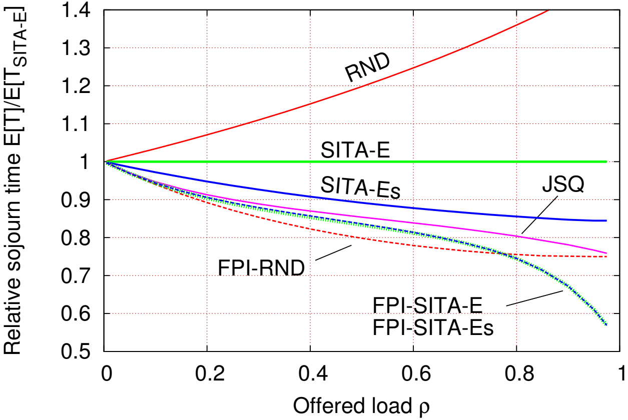


- Two identical FCFS servers
- $X \sim \mathrm{U}(0, 2)$

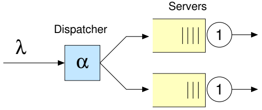


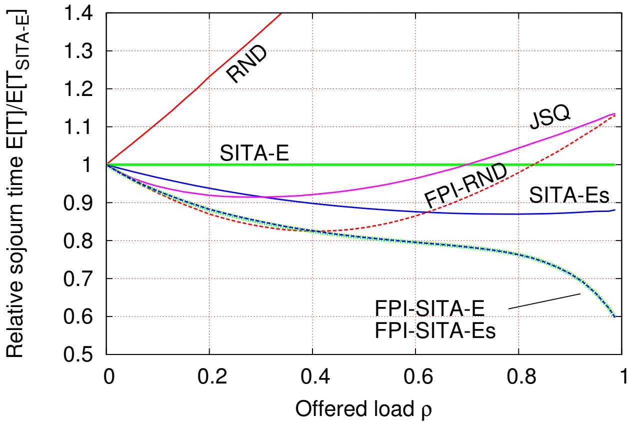
Aalto University
School of Electrical Engineering

FCFS with Exponentially distributed jobs

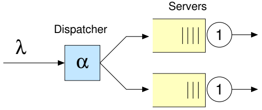


- Two identical FCFS servers
- $X \sim \text{Exp}(1)$



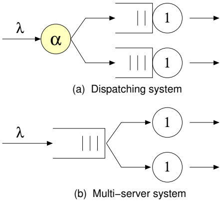


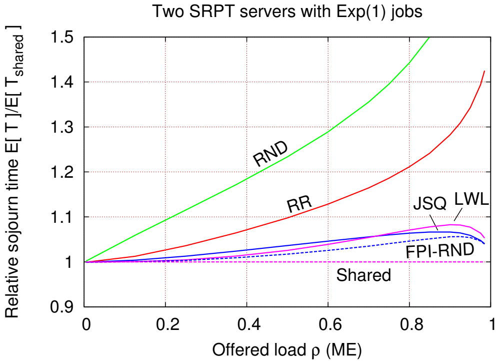
- Two identical FCFS servers
- $X \sim \text{Pareto}(1)$



Aalto University
School of Electrical Engineering

SRPT and Exponentially distributed jobs



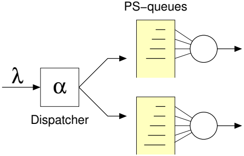


- Dispatching system vs. a shared queue with SRPT (M/M/2-SRPT).
- Disadvantage due to the dispatching can be insignificant (here order of 5% with FPI-RND).

Aalto University
School of Electrical Engineering

System:

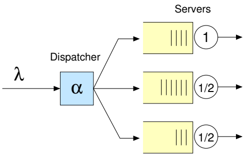
- Poisson arrival process
- Fixed server-specific service time $d_i = d/\nu_i$



Dispatching policies:

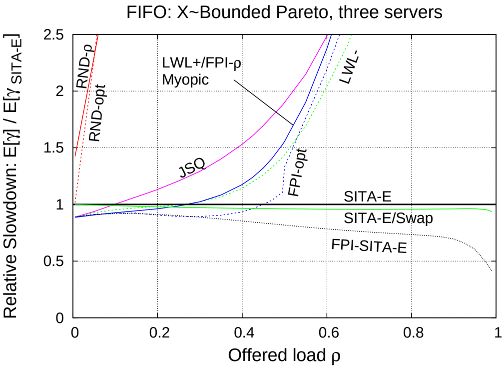
Random split balancing the load	RND-ρ	
Least-work-left (pre-assignment)	LWL$^-$:	$\arg\min_i u_i$
Least-work-left (post-assignment)	LWL$^+$:	$\arg\min_i u_i + d_i$
FPI for RND-ρ	FPI:	$\arg\min_i u_i + (1/2)d_i$

Aalto University
School of Electrical Engineering



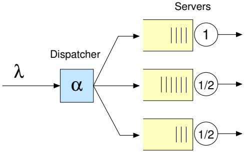
Slowdown:

$$\gamma = \frac{T}{X}.$$



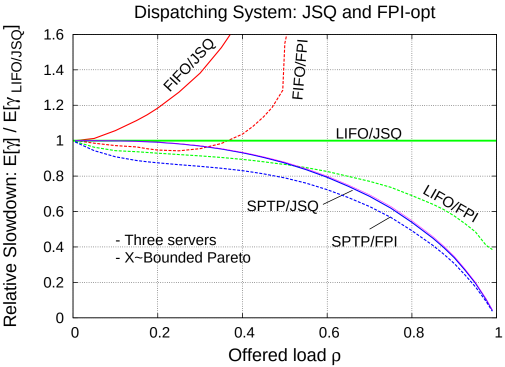
- Three servers with service rates 1, 1/2 and 1/2
- FCFS scheduling discipline
- Bounded Pareto distributed service times

Aalto University
School of Electrical Engineering



Slowdown:

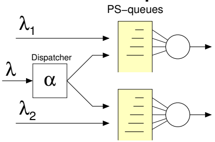
$$\gamma = \frac{T}{X}.$$



- Three servers with service rates 1, 1/2 and 1/2
- Scheduling discipline: FCFS, LCFS and SPTP
- Bounded Pareto distributed service times

Aalto University
School of Electrical Engineering

1. Each server can have dedicated input



2. Basic policy can be class-specific
   - Low and high priority customers with own queues
   - When to route a low priority job to a high priority queue?

3. Service times can be server-specific
   - General purpose vs. specialized servers

Aalto University
School of Electrical Engineering

Conclusions

- Size- and state-aware dispatching problem can be approached in the MDP framework
- Value functions $v_{\mathbf{z}}$ are required for the FPI step.
- For state-independent basic policies, sufficient to analyze an M/G/1 queue in isolation:
  - FCFS and LCFS: $v_{\mathbf{z}}$ is insensitive to job size distribution.
  - SPT, SRPT and SPTP: $v_{\mathbf{z}}$ is an integral expression.
  - PS: harder to analyze (M/D/1-PS and M/M/1-PS)
- Efficient dispatching policies that take into account
  - cost structure
  - existing and later arriving tasks

Thanks!

Aalto University
School of Electrical Engineering

References:

1. Hyytiä, Penttinen and Aalto, *Size- and State-Aware Dispatching Problem with Queue-Specific Job Sizes*, EJOR 2012.
2. Hyytiä, Virtamo, Aalto and Penttinen, *M/M/1-PS Queue and Size-Aware Task Assignment*, Performance 2011.
3. Hyytiä, Penttinen, Aalto and Virtamo, *Dispatching problem with fixed size jobs and processor sharing discipline*, ITC'23, 2011.
4. Hyytiä, Aalto and Penttinen, *Minimizing Slowdown in Heterogeneous Size-Aware Dispatching Systems*, SIGMETRICS 2012.
5. Hyytiä, Aalto, Penttinen and Virtamo, *On the value function of the M/G/1 FCFS and LCFS queues*, Journal of Applied Probability, 2012, *to appear*.

Aalto University
School of Electrical Engineering

Proof for M/G/1-FCFS

Consider two systems under the same arrivals:

- S1 initially in state $\mathbf{z} = (\Delta_1; ..; \Delta_n)$ and
- S2 initially empty.

Both systems behave identically once S1 becomes empty. The difference in the relative values is equal to the additional time jobs spend in S1,

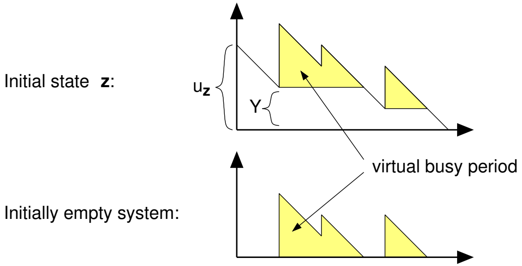
$$v_{\mathbf{z}} - v_0 = V_1 + V_2,$$

where V_1 denotes the (remaining) delay of present jobs, and V_2 the additional mean delay the later arrivals experience in S1.

The total delay of the *n* present jobs in S1 is already fixed,

$$V_1 = \sum_{i=1}^{n} i\,\Delta_i.$$

- A later arriving task starts a busy period in S2, which corresponds to a mini busy period in S1.



- During busy periods, arriving jobs increase the cumulative delay by an amount equal to the post arrival workload.
- These jobs experience an additional delay Y in S1.
- Otherwise the delay contributions are equal!

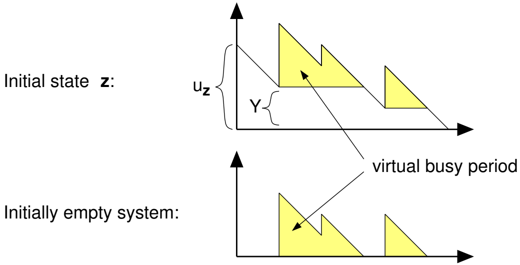
Aalto University
School of Electrical Engineering

Summing up:

- Mean number of busy periods before S1 empty: $\lambda\, u_{\mathbf{z}}$.
- Mean number of jobs arriving during a busy period: $1/(1-\rho)$.
- Mean offset $\mathrm{E}[Y] = u_{\mathbf{z}}/2$.

Therefore,

$$V_2 = \lambda\, u_{\mathbf{z}} \cdot \frac{1}{1-\rho} \cdot \frac{u_{\mathbf{z}}}{2} = \frac{\lambda\, u_{\mathbf{z}}^2}{2(1-\rho)},$$



and $V_1 + V_2 = v_{\mathbf{z}} - v_0$, which completes the proof.

Aalto University
School of Electrical Engineering

Proof for M/G/1-LCFS

- Consider two systems under same arrivals:
  1. S1 initially in state $\mathbf{z} = (\Delta_1, .., \Delta_n)$,
  2. S2 initially empty.
- Let D_i denote the (remaining) delay of job i in S1.
- With LCFS, the current state has no effect on the future arrivals' sojourn times.
- The difference between the relative value of S1 and S2 is equal to the mean remaining delay of the n present jobs,

$$v_{(\Delta_1;..;\Delta_n)} - v_0 = \sum_{i=1}^{n} \mathrm{E}[D_i].$$

Aalto University
School of Electrical
Engineering

- Remaining delay D_n of job n is given by a random sum,

$$D_n = \Delta_n + \left(B_1 + \ldots + B_{A(\Delta_n)}\right)$$

where $A(\Delta_n)$ denotes the number of (mini) busy periods during time Δ_n, and B_i the corresponding durations,

$$\mathrm{E}[B_i] = \mathrm{E}[X]/(1-\rho).$$

- Taking the expectation on both sides gives

$$\mathrm{E}[D_n] = \Delta_n + \mathrm{E}[A(\Delta_n)] \cdot \mathrm{E}[B] = \frac{\Delta_n}{1-\rho}.$$

- Similarly, $\mathrm{E}[D_i] = \dfrac{1}{1-\rho} \sum_{j=i}^{n} \Delta_j.$

$$\Rightarrow \quad v_{\mathbf{z}} - v_0 = \sum_{i=1}^{n} \mathrm{E}[D_i] = \boxed{\frac{1}{1-\rho} \sum_{i=1}^{n} i \cdot \Delta_i.}$$

Aalto University
School of Electrical Engineering

Proposition: The size-aware relative value of state **z** with respect to arbitrary holding costs in an M/G/1-SRPT queue is[12]

$$\boxed{\begin{aligned} v_{\mathbf{z}} - v_0 = {} & \sum_{i=1}^{n} b_i \left(\frac{u_{\mathbf{z}}(\Delta_i)}{1-\rho(\Delta_i)} + \int_0^{\Delta_i} \frac{1}{1-\rho(t)}\, dt \right) \\ & + \frac{\lambda}{2} \int_0^{\infty} h(x) \left(u_{\mathbf{z}}(x)^2 + n_{\mathbf{z}}(x)\, x^2 \right)\, dx, \end{aligned}} \qquad (10)$$

where

- job n receives currently service and $\Delta_1 > \ldots > \Delta_n$,
- $u_{\mathbf{z}}(x)$ = backlog due to jobs shorter than x in state **z**,
- $n_{\mathbf{z}}(x)$ = number of jobs longer than x in state **z**.

Aalto University
School of Electrical Engineering

[12] Hyytiä et al., Eur. J. Oper. Research (2012)