Product (Re)forms
Part 1

Kristy Gardner
Computer Science Department
Amherst College

Rhonda Righter
IEOR Department
UC Berkeley

INFORMS APS, July 4, 2019

Based on work by…

Adan, Ahn, Akgun, Anton, Atar, Ayesta, Bodas, Bonald,

Busic, Caldentey, Comte, Doroudi, Dorsman, Gardner,

Gel, Gupta, Harchol-Balter, Hellemans, Hopp, Hurkens,

Hyytiä, Jonckheere, Kaplan, Kesslassy, Kleiner, Krzesinski,

Mairesse, Mathieu, Mendelson, Moyal, Perry, Righter,

Scheller-Wolf, van Houdt, Van Oyen, Velednitsky,

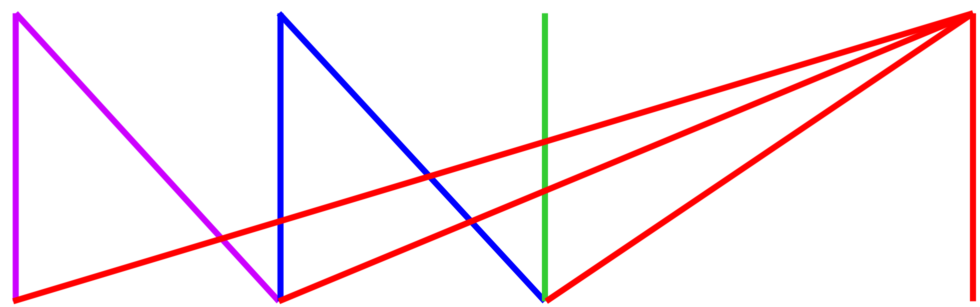
Verloop, Visschers, Weiss, Wolff, Zbarsky…

System Structure

Poisson arrivals: $\lambda_1 + \lambda_2 + \lambda_3 + \cdots + \lambda_J = \lambda$

Job classes: $1 \quad 2 \quad 3 \quad \ldots \quad J$

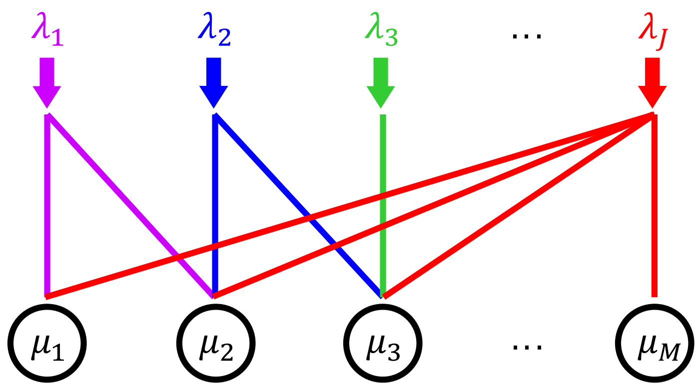
Bipartite matching structure:



Servers: $1 \quad 2 \quad 3 \quad \ldots \quad M$

Exponential service times: $\mu_1 + \mu_2 + \mu_3 + \cdots + \mu_M = \mu$

System Structure



For stability, assume:

$$\lambda(A) < \mu(A)$$

for all job class subsets A

Notation

$S_i = \{k \mid \text{server } k \text{ can serve class } i\}$

for subset of job classes A: $S(A) = \bigcup_{i \in A} S_i$

$$\lambda(A) = \sum_{i \in A} \lambda_i \qquad \mu(A) = \sum_{k \in S(A)} \mu_k$$

$$\lambda = \lambda(\{1, \dots, J\}) \qquad \mu = \mu(\{1, \dots, M\})$$

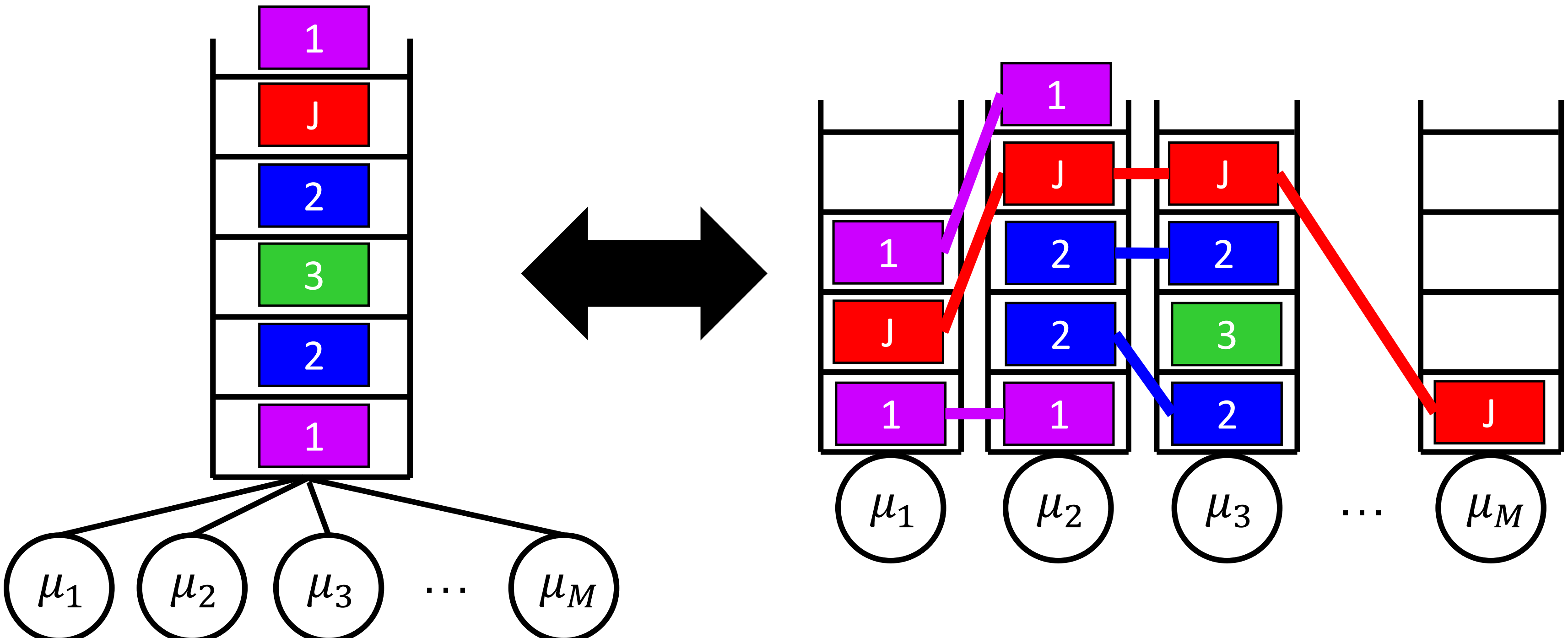
$C_k = \{i \mid \text{server } k \text{ can serve class } i\}$

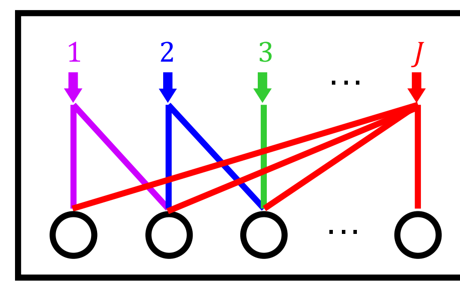
for subset of servers B: $C(B) = \bigcup_{k \in B} C_k$

$$\lambda(B) = \sum_{i \in C(B)} \lambda_i \qquad \mu(B) = \sum_{k \in B} \mu_k$$

Two Equivalent Views

Central Queue | Job Redundancy

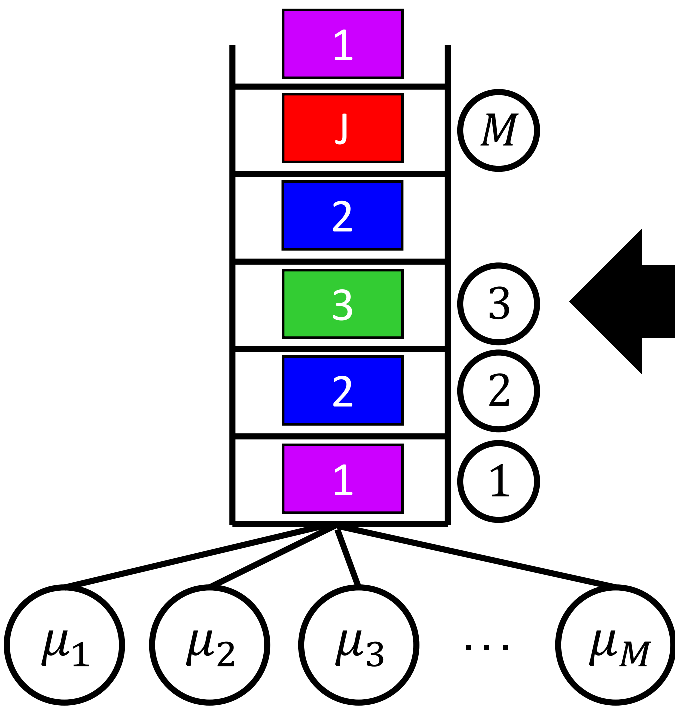




The Noncollaborative Model

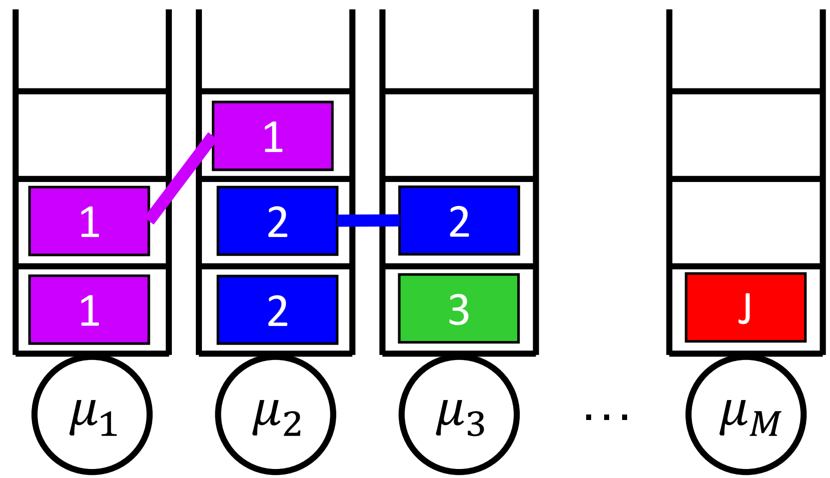
A job can be served by only one server

Central Queue

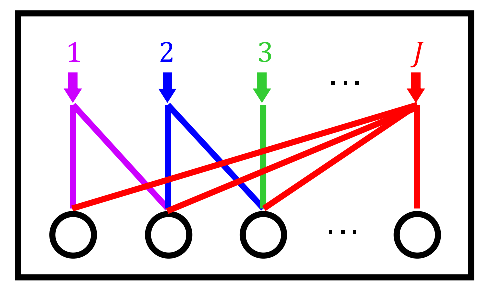


Job Redundancy

"Cancel-on-start" ⟷ Join the smallest work



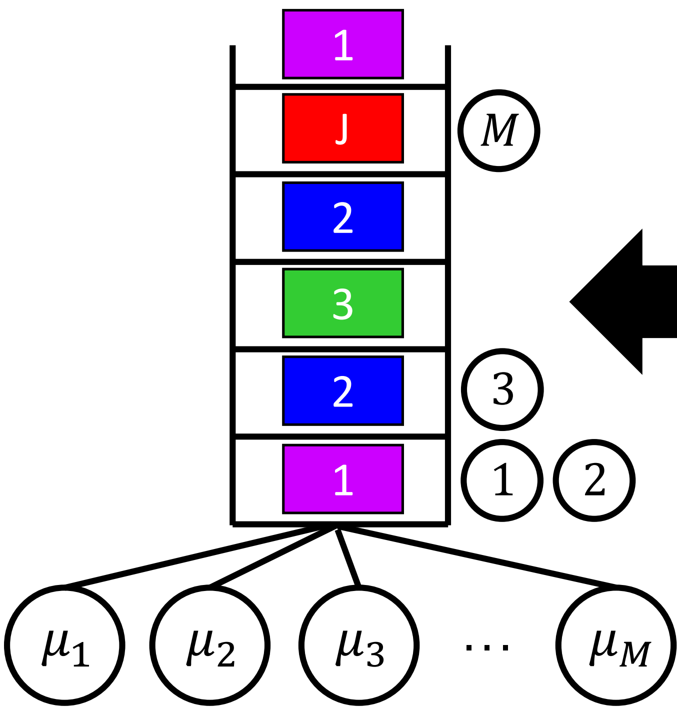
If multiple compatible idle servers upon arrival:
Assign Longest Idle Server (ALIS) [Adan, Weiss]

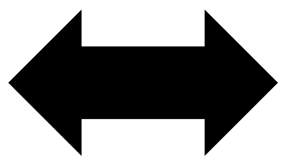


The Collaborative Model

Multiple servers can work on the same job at once, with additive service rate

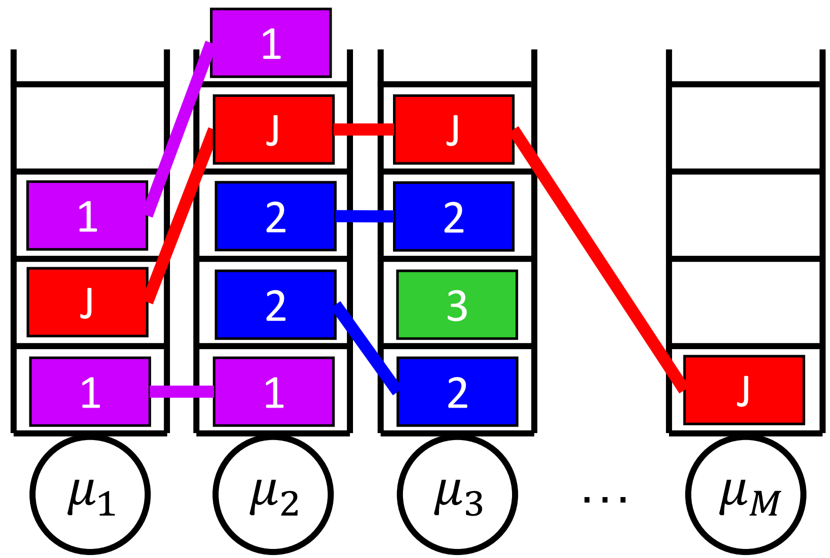
Central Queue



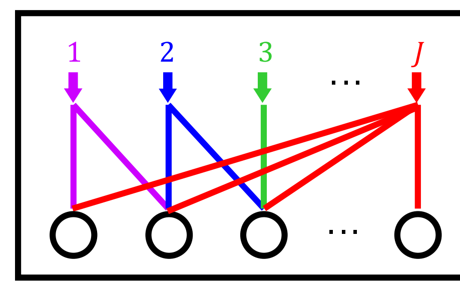


Job Redundancy

"Cancel-on-completion"



If multiple compatible idle servers upon arrival: job enters service on all of them



Outline

- System description
- **Review: reversibility**
- Detailed states

  Collaborative model — Noncollaborative model

  Relating the models
- Partial aggregation

  Collaborative model — Noncollaborative model

  - Special case: fully flexible class
  - Special case: nested systems
- Related product-form systems

Reversibility Review

Def: A stochastic process $X(t)$ is reversible if $(X(t_0), \dots, X(t_n))$ has the same distribution as $(X(t_n), \dots, X(t_0))$ for all $t_0, \dots, t_n$

Reversible ⟺ Detailed balance eqns

$$\pi_i v_{ij} = \pi_j v_{ji}$$

for all states i, j

v_{ij}: rate of going from state i to state j

Def: A queue is quasi-reversible if its state at time t is independent of:

- Arrival times after time t
- Departure times before time t

Quasi-reversible ⟺ Partial balance (for our model)

rate into state i due to arrival = rate out of state i due to departure

rate out of state i due to class-c arrival = rate into state i due to class-c departure

Reversibility Review

Def: A stochastic process $X(t)$ is reversible if $(X(t_0), \dots, X(t_n))$ has the same distribution as $(X(t_n), \dots, X(t_0))$ for all $t_0, \dots, t_n$

Reversible ⟺ Detailed balance eqns

$$\pi_i v_{ij} = \pi_j v_{ji}$$

for all states i, j

v_{ij}: rate of going from state i to state j

Def: A queue is quasi-reversible if its state at time t is independent of:

- Arrival times after time t
- Departure times before time t

Quasi-reversible ⟺ Burke's Theorem: Departure process is a Poisson process ⟺ Networks of queues have product-form stationary distribution

Outline

- System description
- Review: reversibility
- **Detailed states**

  Collaborative model Noncollaborative model

  Relating the models

- Partial aggregation

  Collaborative model Noncollaborative model

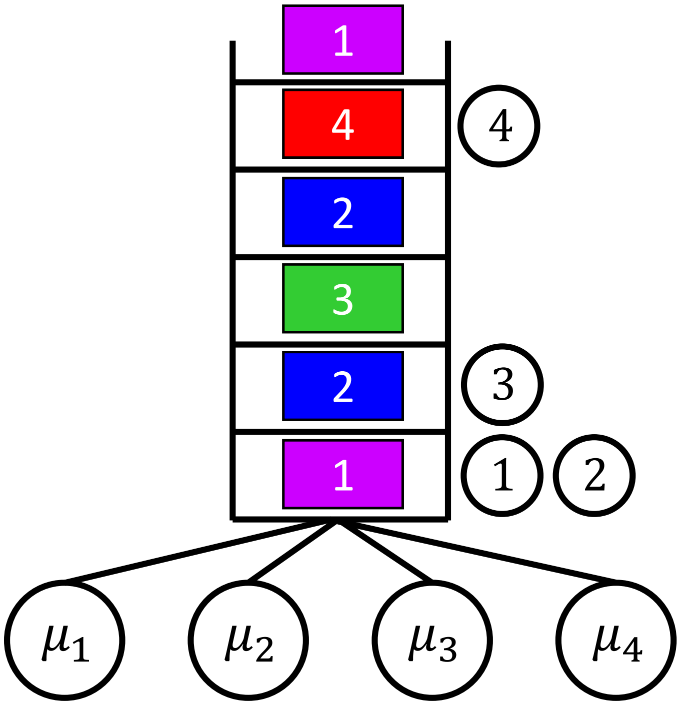
  - Special case: fully flexible class
  - Special case: nested systems

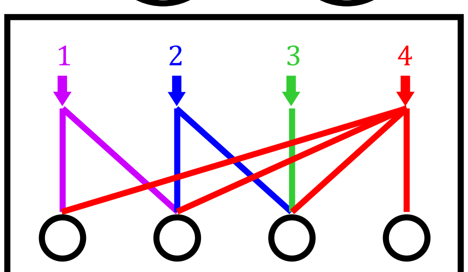
- Related product-form systems

Detailed States: Collaborative Model

$$\text{state } \vec{c}_n = (c_1, c_2, \dots, c_i, \dots, c_n)$$

class of job in position i in the system (including jobs in service)





Total rate of service given to jobs $1, \dots, i$:

$$\mu(\vec{c}_i) = \sum_{k \in S(\vec{c}_i)} \mu_k = \mu(\{c_1, \dots, c_i\})$$

Rate of service given to job i:

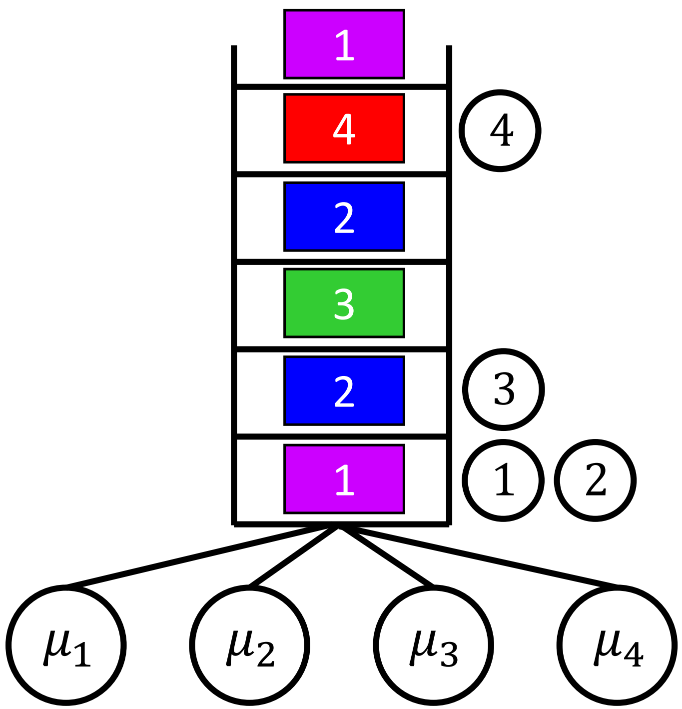
$$\Delta_i(\vec{c}_n) = \sum_{k \in S(\vec{c}_i)} \mu_k - \sum_{k \in S(\vec{c}_{i-1})} \mu_k$$

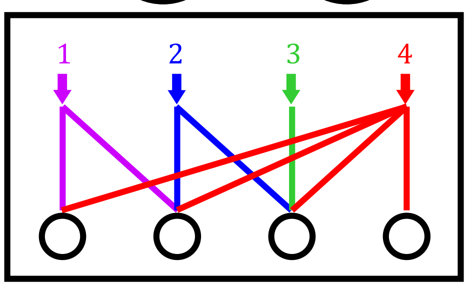
$$= \mu(\vec{c}_i) - \mu(\vec{c}_{i-1})$$

Detailed States: Collaborative Model

$$\text{state } \vec{c}_n = (c_1, c_2, \dots, c_i, \dots, c_n)$$

class of job in position i in the system (including jobs in service)





$$\mu(\vec{c}_i) = \sum_{k \in S(\vec{c}_i)} \mu_k = \mu(\{c_1, \dots, c_i\})$$

$$\Delta_i(\vec{c}_n) = \mu(\vec{c}_i) - \mu(\vec{c}_{i-1})$$

ex. state $\vec{c}_5 = (1,2,3,2,4,1)$

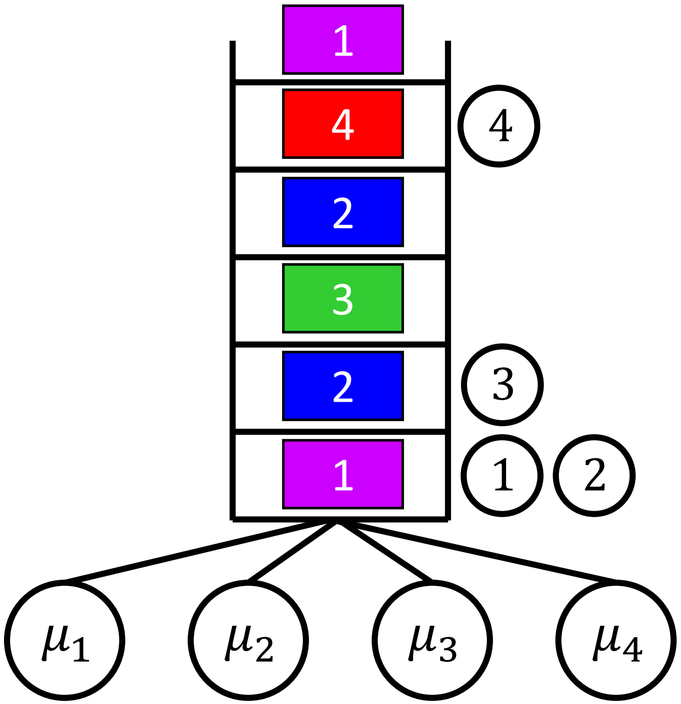
$$\mu(\vec{c}_4) = \mu_1 + \mu_2 + \mu_3$$

$$\Delta_2(\vec{c}_6) = \mu(\vec{c}_2) - \mu(\vec{c}_1)$$

$$= (\mu_1 + \mu_2 + \mu_3) - (\mu_1 + \mu_2) = \mu_3$$

Detailed States: Collaborative Model

state $\vec{c}_n = (c_1, c_2, \ldots, c_i, \ldots, c_n)$

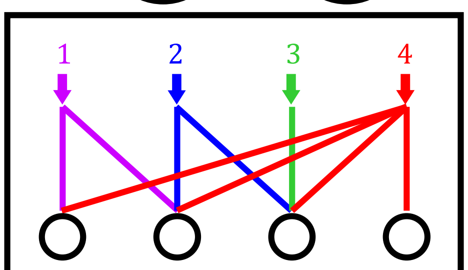


Properties

(i) $\Delta_i(\vec{c}_n) = \Delta_i(\vec{c}_i)$ for all $i \leq n$

(ii) $\mu(\vec{c}_n)$ is the same for any permutation of $c_1, \ldots, c_n$

(iii) $\mu(c) > 0$ for all job classes c



Queues satisfying (i)-(iii) are **order independent** (OI)

OI properties are sufficient to ensure product form stationary distribution

[Krzesinski]

Order Independence

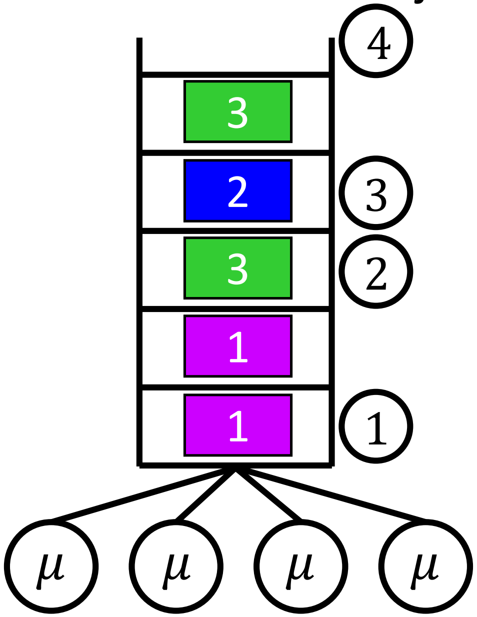
Queues satisfying (i)-(iii) are **order independent** (OI)

OI properties are sufficient to ensure product form stationary distribution

Much more general than collaborative model

ex. Multi-Server Station with Concurrent Classes of Customers (**MSCCC**) queue

- All job classes are compatible with all servers
- At most one job per class can be in service at a time



The MSCCC queue satisfies OI properties ➡ Product-form stationary distribution

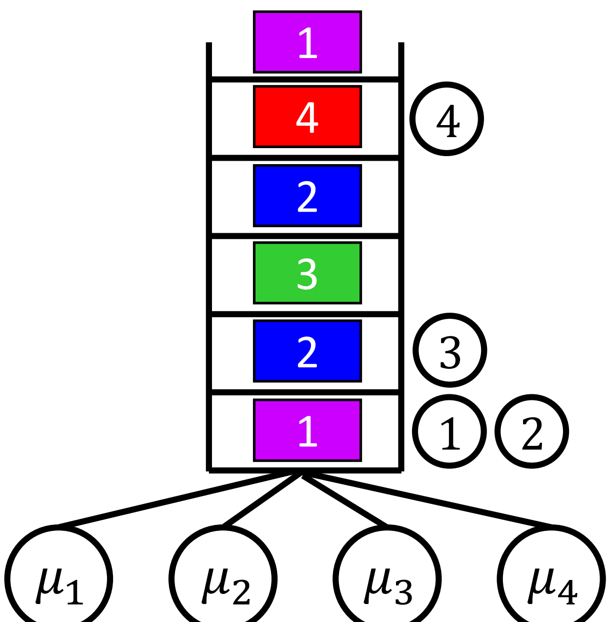
But does not fit within the collaborative model

Detailed States: Collaborative Model

Thm: Given OI properties, the system is quasi-reversible and the stationary distribution is:

$$\pi(\vec{c}_n) = \pi(0) \prod_{i=1}^{n} \frac{\lambda_{c_i}}{\mu(\vec{c}_i)} = \frac{\lambda_{c_n}}{\mu(\vec{c}_n)} \pi(\vec{c}_{n-1})$$

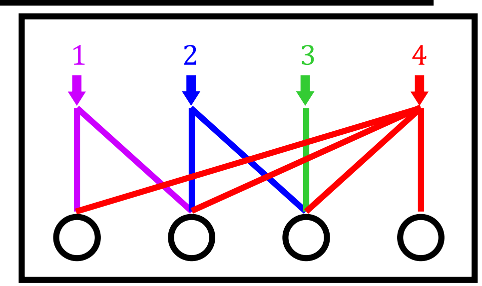
provided $G = \sum_{\vec{c}_n} \prod_{i=1}^{n} \frac{\lambda_{c_i}}{\mu(\vec{c}_i)} < \infty$. Then $\pi(0) = \frac{1}{G}$ is the probability the system is empty.



ex. state $\vec{c}_6 = (1,2,3,2,4,1)$

$$\pi(\vec{c}_6) = \pi(0) \left(\frac{\lambda_1}{\mu_{1,2}}\right)\left(\frac{\lambda_2}{\mu_{1,2,3}}\right)\left(\frac{\lambda_3}{\mu_{1,2,3}}\right)\left(\frac{\lambda_2}{\mu_{1,2,3}}\right)\left(\frac{\lambda_4}{\mu}\right)\left(\frac{\lambda_1}{\mu}\right)$$

[Gardner, Zbarsky, Doroudi, Harchol-Balter, Hyytiä, Scheller-Wolf, Krzesinski, Bonald, Comte]



Detailed States: Collaborative Model

Thm: $$\pi(\vec{c}_n) = \pi(0)\prod_{i=1}^{n}\frac{\lambda_{c_i}}{\mu(\vec{c}_i)} = \frac{\lambda_{c_n}}{\mu(\vec{c}_n)}\pi(\vec{c}_{n-1})$$ ★

[Krzesinski]

Proof: Show ★ satisfies partial balance

rate out of $\vec{c}_n$ due to departure = rate into $\vec{c}_n$ due to arrival

$$\pi(\vec{c}_n)\mu(\vec{c}_n) = \pi(\vec{c}_{n-1})\lambda_{c_n}$$ ✓

Detailed States: Collaborative Model

Thm: $$\pi(\vec{c}_n) = \pi(0)\prod_{i=1}^{n}\frac{\lambda_{c_i}}{\mu(\vec{c}_i)} = \frac{\lambda_{c_n}}{\mu(\vec{c}_n)}\pi(\vec{c}_{n-1}) \quad ★$$

[Krzesinski]

Proof: Show ★ satisfies partial balance

rate out of $\vec{c}_n$ due to class c arrival = rate into $\vec{c}_n$ due to class c departure

Approach: by induction

$$\pi(\vec{c}_n)\lambda_c \overset{?}{=} \sum_{j=1}^{n}\pi(c_1,\ldots,c_j,c,c_{j+1},\ldots,c_n)\Delta_{j+1}(c_1,\ldots,c_j,c,c_{j+1},\ldots,c_n)$$

$$= \sum_{j=1}^{n}\pi(c_1,\ldots,c_j,c,c_{j+1},\ldots,c_n)\Delta_{j+1}(\vec{c}_j,c)$$ (OI condition (i))

Base case ($n = 0$): $\pi(0)\lambda_c = \pi(c)\Delta_1(c) = \pi(c)\mu_c$ ✓

Detailed States: Collaborative Model

Thm:

$$\pi(\vec{c}_n) = \pi(0)\prod_{i=1}^{n}\frac{\lambda_{c_i}}{\mu(\vec{c}_i)} = \frac{\lambda_{c_n}}{\mu(\vec{c}_n)}\pi(\vec{c}_{n-1}) \quad \star$$

[Krzesinski]

Proof: Show ★ satisfies partial balance

Inductive hypothesis: $\pi(\vec{c}_{n-1})\lambda_c = \sum_{j=1}^{n-1}\pi(c_1,\ldots,c_j,c,c_{j+1},\ldots,c_{n-1})\Delta_{j+1}(\vec{c}_j,c)$

$$\pi(\vec{c}_n)\lambda_c \overset{?}{=} \sum_{j=1}^{n}\pi(c_1,\ldots,c_j,c,c_{j+1},\ldots,c_n)\Delta_{j+1}(\vec{c}_j,c)$$

$$= \sum_{j=1}^{n-1}\pi(c_1,\ldots,c_j,c,c_{j+1},\ldots,c_n)\Delta_{j+1}(\vec{c}_j,c) + \pi(\vec{c}_n,c)\Delta_{j+1}(\vec{c}_n,c)$$

Detailed States: Collaborative Model

Thm: $$\pi(\vec{c}_n) = \pi(0) \prod_{i=1}^{n} \frac{\lambda_{c_i}}{\mu(\vec{c}_i)} = \frac{\lambda_{c_n}}{\mu(\vec{c}_n)} \pi(\vec{c}_{n-1})$$ ★

[Krzesinski]

Proof: Show ★ satisfies partial balance

Inductive hypothesis: $$\pi(\vec{c}_{n-1})\lambda_c = \sum_{j=1}^{n-1} \pi\big(c_1, \dots, c_j, c, c_{j+1}, \dots, c_{n-1}\big)\Delta_{j+1}\big(\vec{c}_j, c\big)$$

$$\pi(\vec{c}_n)\lambda_c \stackrel{?}{=} \sum_{j=1}^{n-1} \pi\big(c_1, \dots, c_j, c, c_{j+1}, \dots, c_n\big)\Delta_{j+1}\big(\vec{c}_j, c\big) + \pi(\vec{c}_n, c)\Delta_{j+1}(\vec{c}_n, c)$$

$$= \frac{\lambda_{c_n}}{\mu(\vec{c}_n, c)} \sum_{j=1}^{n-1} \pi\big(c_1, \dots, c_j, c, c_{j+1}, \dots, c_{n-1}\big)\Delta_{j+1}\big(\vec{c}_j, c\big) + \frac{\lambda_c}{\mu(\vec{c}_n, c)} \pi(\vec{c}_n)\Delta_{j+1}(\vec{c}_n, c)$$ (★)

$$= \frac{\lambda_{c_n}}{\mu(\vec{c}_n, c)} \pi(\vec{c}_{n-1})\lambda_c + \frac{\lambda_c}{\mu(\vec{c}_n, c)} \pi(\vec{c}_n)\big(\mu(\vec{c}_n, c) - \mu(\vec{c}_n)\big)$$ (Inductive hypothesis)
(def. of $\Delta_{j+1}(\vec{c}_n, c)$)

Detailed States: Collaborative Model

Thm: $$\pi(\vec{c}_n) = \pi(0)\prod_{i=1}^{n}\frac{\lambda_{c_i}}{\mu(\vec{c}_i)} = \frac{\lambda_{c_n}}{\mu(\vec{c}_n)}\pi(\vec{c}_{n-1}) \quad ★$$

[Krzesinski]

Proof: Show ★ satisfies partial balance

Inductive hypothesis: $$\pi(\vec{c}_{n-1})\lambda_c = \sum_{j=1}^{n-1}\pi\left(c_1,\dots,c_j,c,c_{j+1},\dots,c_{n-1}\right)\Delta_{j+1}\left(\vec{c}_j,c\right)$$

$$\pi(\vec{c}_n)\lambda_c \overset{?}{=} \sum_{j=1}^{n-1}\pi\left(c_1,\dots,c_j,c,c_{j+1},\dots,c_n\right)\Delta_{j+1}\left(\vec{c}_j,c\right) + \pi(\vec{c}_n,c)\Delta_{j+1}(\vec{c}_n,c)$$

$$= \frac{\lambda_{c_n}}{\mu(\vec{c}_n,c)}\sum_{j=1}^{n-1}\pi\left(c_1,\dots,c_j,c,c_{j+1},\dots,c_{n-1}\right)\Delta_{j+1}\left(\vec{c}_j,c\right) + \frac{\lambda_c}{\mu(\vec{c}_n,c)}\pi(\vec{c}_n)\Delta_{j+1}(\vec{c}_n,c) \quad (★)$$

$$= \frac{\lambda_{c_n}}{\mu(\vec{c}_n,c)}\pi(\vec{c}_{n-1})\lambda_c + \frac{\lambda_c}{\mu(\vec{c}_n,c)}\pi(\vec{c}_n)\left(\mu(\vec{c}_n,c) - \mu(\vec{c}_n)\right)$$

(Inductive hypothesis)
(def. of $\Delta_{j+1}(\vec{c}_n,c)$)

Detailed States: Collaborative Model

Thm: $$\pi(\vec{c}_n) = \pi(0)\prod_{i=1}^{n}\frac{\lambda_{c_i}}{\mu(\vec{c}_i)} = \frac{\lambda_{c_n}}{\mu(\vec{c}_n)}\pi(\vec{c}_{n-1}) \quad ★$$

[Krzesinski]

Proof: Show ★ satisfies partial balance

Inductive hypothesis: $$\pi(\vec{c}_{n-1})\lambda_c = \sum_{j=1}^{n-1}\pi(c_1,\ldots,c_j,c,c_{j+1},\ldots,c_{n-1})\Delta_{j+1}(\vec{c}_j,c)$$

$$\pi(\vec{c}_n)\lambda_c \overset{?}{=} \sum_{j=1}^{n-1}\pi(c_1,\ldots,c_j,c,c_{j+1},\ldots,c_n)\Delta_{j+1}(\vec{c}_j,c) + \pi(\vec{c}_n,c)\Delta_{j+1}(\vec{c}_n,c)$$

$$= \frac{\lambda_{c_n}}{\mu(\vec{c}_n,c)}\sum_{j=1}^{n-1}\pi(c_1,\ldots,c_j,c,c_{j+1},\ldots,c_{n-1})\Delta_{j+1}(\vec{c}_j,c) + \frac{\lambda_c}{\mu(\vec{c}_n,c)}\pi(\vec{c}_n)\Delta_{j+1}(\vec{c}_n,c) \quad (★)$$

$$= \lambda_c\pi(\vec{c}_n)$$ ✓

(just a little algebra)

Computations for Collaborative Model

Thm: $\pi(0)$ can be computed recursively:

$$\pi(0) = \frac{\mu - \lambda}{\sum_{s \in M} \frac{\mu_s}{\pi_{-s}(0)}}$$

where -s denotes a system in which server s and all job classes compatible with server s are removed

Thm: Let $E\left[N^{(i)}\right]$ and $E[N]$ denote the number of class i jobs and the total number of jobs in the system.
$E\left[N^{(i)}\right]$ and $E[N]$ can be computed recursively:

$$E\left[N^{(i)}\right] = \frac{\lambda_i + \sum_{s \in M \setminus C_i} \mu_s \pi^{(s)}(0) E\left[N_{-s}^{(i)}\right]}{\mu - \lambda}$$

$$E[N] = \frac{\lambda + \sum_{s \in M} \mu_s \pi^{(s)}(0) N_{-s}}{\mu - \lambda}$$

[Bonald, Comte, Mathieu]

Outline

- System description
- Review: reversibility
- **Detailed states**

  Collaborative model **Noncollaborative model**

  Relating the models
- Partial aggregation

  Collaborative model Noncollaborative model
  - Special case: fully flexible class
  - Special case: nested systems
- Related product-form systems

Detailed States: Noncollaborative Model

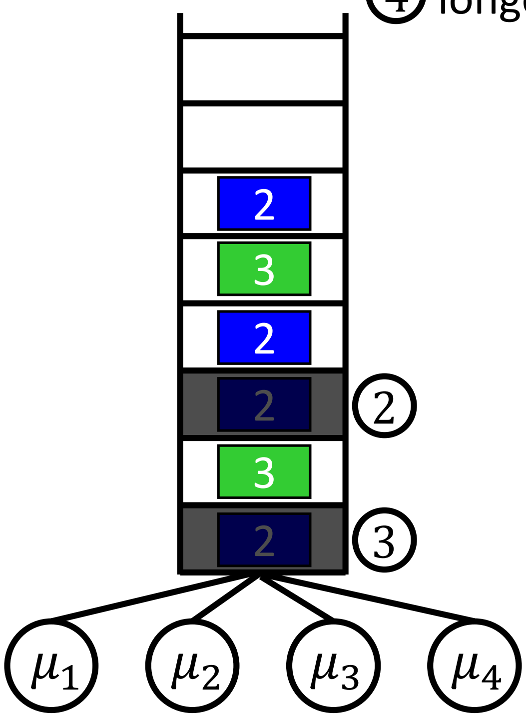
$$\text{state } (\vec{c}_n, \vec{s}_k) = (c_1, \dots, c_i, \dots, c_n; s_1, \dots, s_l, \dots, s_k)$$

class of job in position i in the **queue** (not including jobs in service)

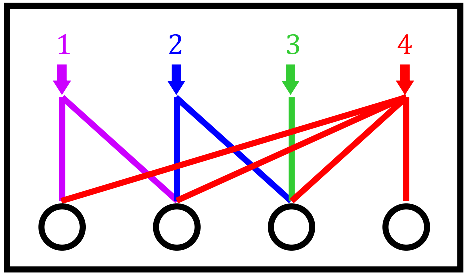
server that has been idle lth longest

① most recently idle

④ longest idle



Recall: Assign Longest Idle Server (ALIS)
if arrival finds multiple compatible idle servers

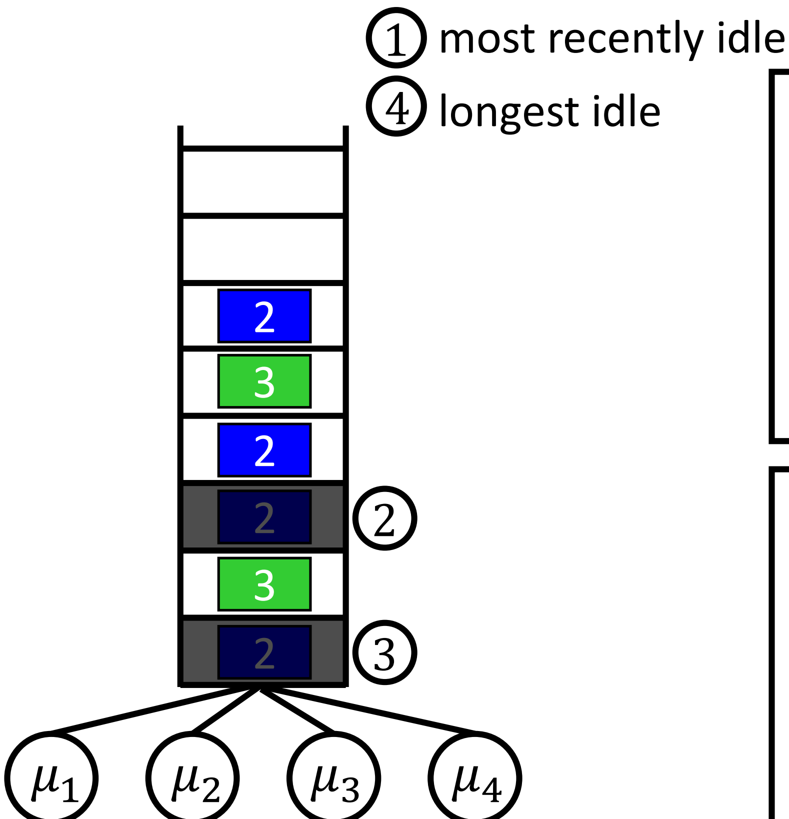


Detailed States: Noncollaborative Model

state $(\vec{c}_n, \vec{s}_k) = (c_1, \ldots, c_i, \ldots, c_n; s_1, \ldots, s_l, \ldots, s_k)$

class of job in position i in the **queue** (not including jobs in service)

server that has been idle lth longest



$$\mu(\vec{c}_i) = \sum_{k \in S(\vec{c}_i)} \mu_k = \mu(\{c_1, \ldots, c_i\})$$

$$\Delta_i(\vec{c}_n) = \mu(\vec{c}_i) - \mu(\vec{c}_{i-1}) =$$ rate at which job i leaves the **queue** (enters service)

ex. state $(\vec{c}_4; \vec{s}_2) = (3,2,3,2; 4,1)$

$$\mu(\vec{c}_4) = \mu_2 + \mu_3$$

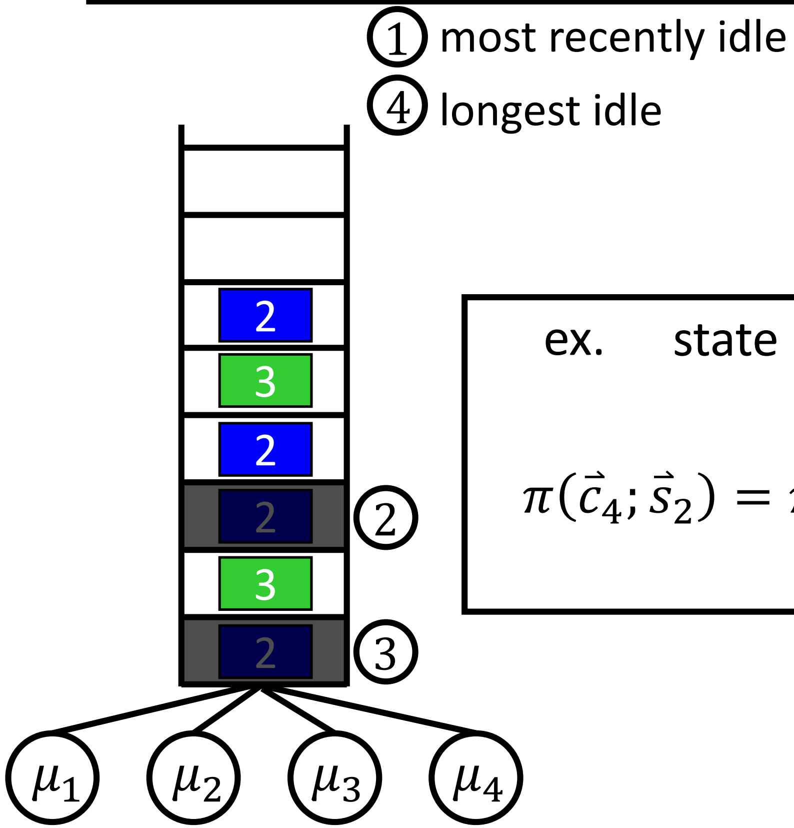
$$\Delta_2(\vec{c}_4) = \mu(\vec{c}_2) - \mu(\vec{c}_1) = (\mu_2 + \mu_3) - \mu_3 = \mu_2$$

Detailed States: Noncollaborative Model

Thm:

$$\pi(\vec{c}_n; \vec{s}_k) = \pi(0;0) \prod_{i=1}^{n} \frac{\lambda_{c_i}}{\mu(\vec{c}_i)} \prod_{l=1}^{k} \frac{\mu_{s_l}}{\lambda(\vec{s}_l)}$$

where $\pi(0;0)$ is the probability that the queue is empty and there are no idle servers



$\pi(0;0) \neq \Pr\{\text{system is empty}\}$

$\pi(0;0) \neq \pi(0)$ in collab system

ex. state $(\vec{c}_4; \vec{s}_2) = (3,2,3,2; 4,1)$

$$\pi(\vec{c}_4; \vec{s}_2) = \pi(0;0) \left(\frac{\lambda_3}{\mu_3}\right) \left(\frac{\lambda_2}{\mu_{2,3}}\right) \left(\frac{\lambda_3}{\mu_{2,3}}\right) \left(\frac{\lambda_2}{\mu_{2,3}}\right) \left(\frac{\mu_4}{\lambda_4}\right) \left(\frac{\mu_{1,4}}{\lambda_{1,4}}\right)$$

Detailed States: Noncollaborative Model

Thm:

$$\pi(\vec{c}_n; \vec{s}_k) = \pi(0;0) \prod_{i=1}^{n} \frac{\lambda_{c_i}}{\mu(\vec{c}_i)} \prod_{l=1}^{k} \frac{\mu_{s_l}}{\lambda(\vec{s}_l)}$$

where $\pi(0;0)$ is the probability that the queue is empty and there are no idle servers

Proof: Partial balance for job c_i: Similar to collaborative case

Partial balance for server s_l:

Let $\Delta_l^\lambda(\vec{s}_k) = \lambda(\vec{s}_l) - \lambda(\vec{s}_{l-1})$ denote the rate at which the lth idle server will become busy

Observation: idle servers satisfy OI properties:

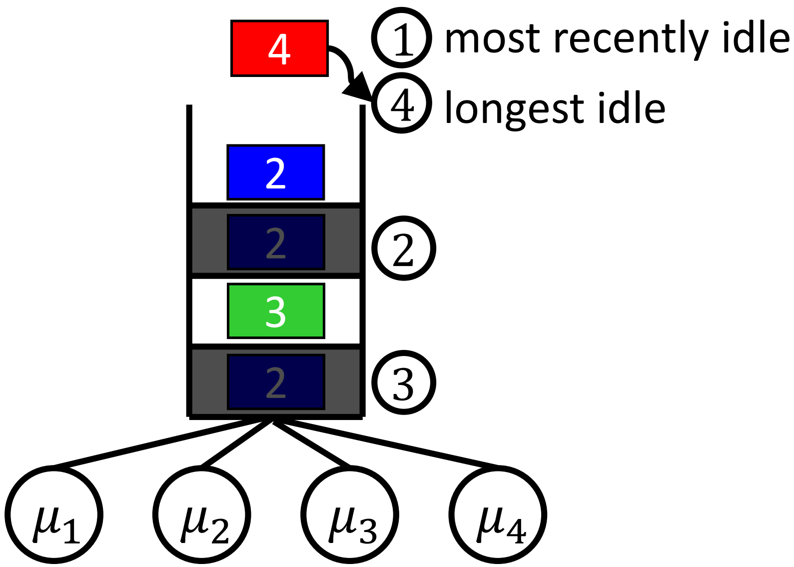
(i) $\Delta_l^\lambda(\vec{s}_k) = \Delta_l^\lambda(\vec{s}_l)$ for all $l \leq k$

(ii) $\lambda(\vec{s}_l)$ is the same for any permutation of $s_1, \dots, s_l$

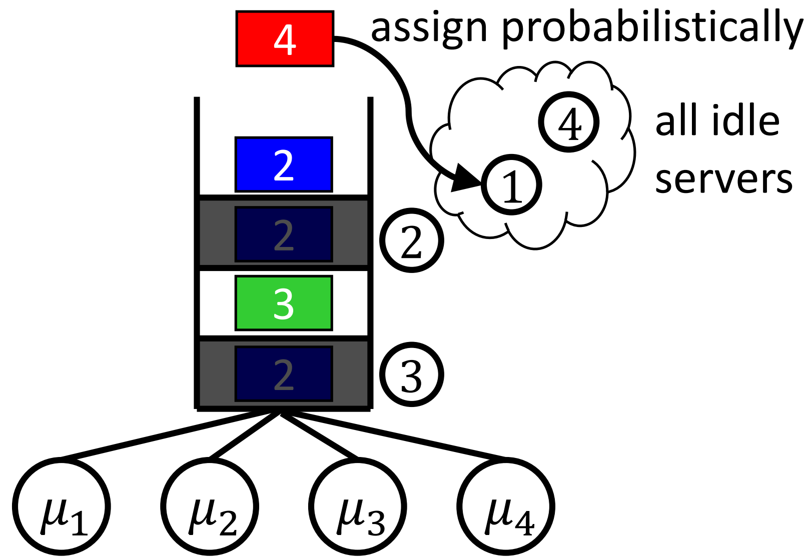
(iii) $\lambda(s) > 0$ for all servers s

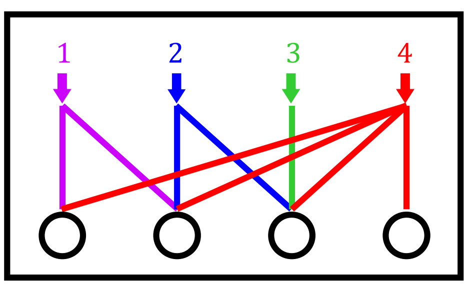
Noncollaborative with Random Assignment

Assign Longest Idle Server



Random Assignment





$\lambda^{A}_{b_l}\left(\vec{b}_{l-1}\right)$: activation rate of server b_l, given busy servers $\vec{b}_{l-1}$

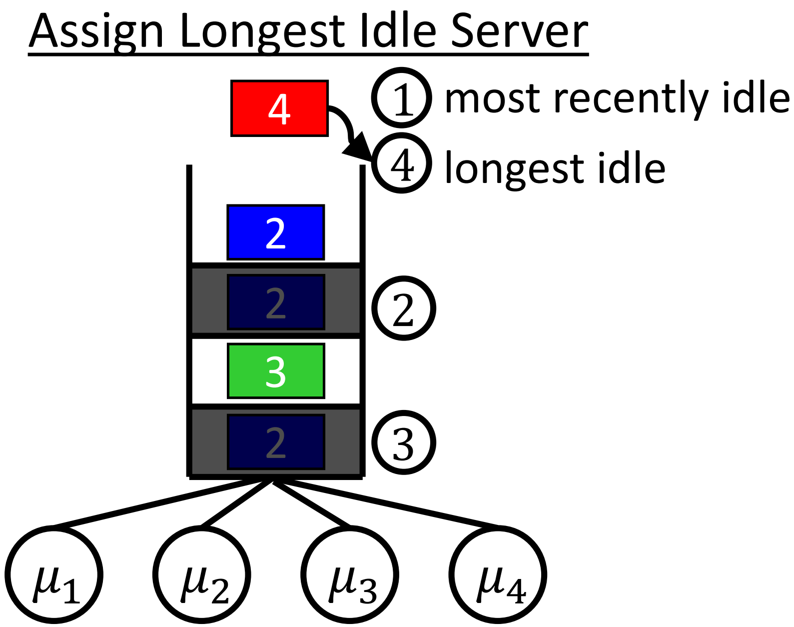
Activation rate condition: For all $1 \le m \le M$,

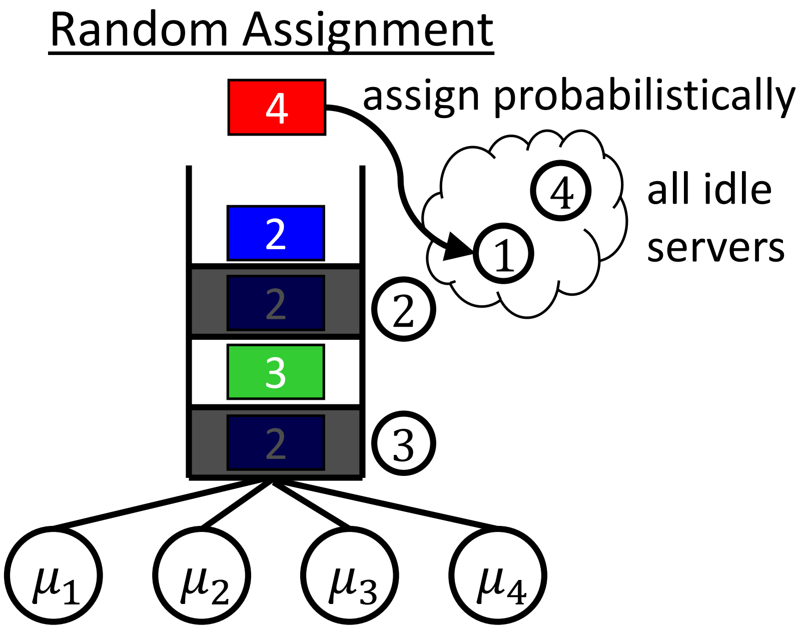
$$\prod_{l=1}^{m} \lambda^{A}_{b_l}\left(\vec{b}_{l-1}\right) \text{ is the same for all permutations of } \vec{b}_m$$

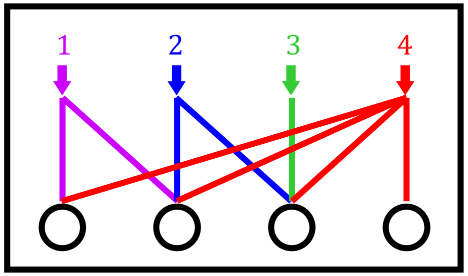
(similar to order independence for busy servers)

[Adan, Visschers, Weiss]

Noncollaborative with Random Assignment







Thm: Given the activation rate condition:

$$\pi^{RAND}\left(\vec{c}_n; \vec{b}_m\right) = \pi^{RAND}(0;0) \prod_{i=1}^{n} \frac{\lambda_{c_i}}{\mu(\vec{c}_i)} \prod_{l=1}^{m} \frac{\lambda_{b_l}^{A}\left(\vec{b}_{l-1}\right)}{\mu\left(\vec{b}_l\right)}$$

[Adan, Visschers, Weiss]

Outline

- System description
- Review: reversibility
- **Detailed states**

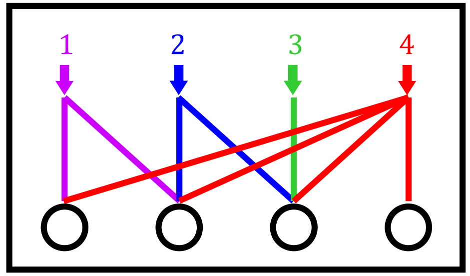
  Collaborative model — Noncollaborative model

  Relating the models
- Partial aggregation

  Collaborative model — Noncollaborative model
  - Special case: fully flexible class
  - Special case: nested systems
- Related product-form systems

Collaborative vs. Noncollaborative

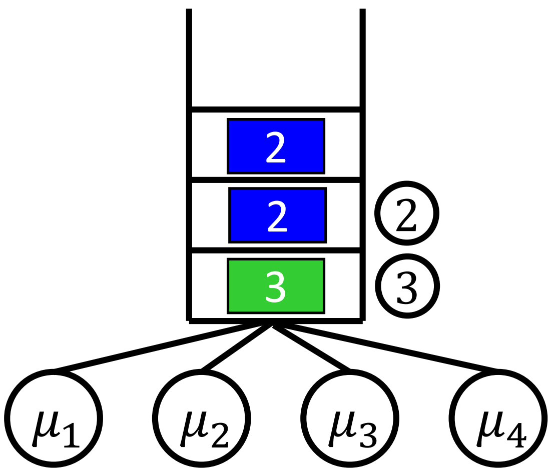
Collaborative (C)

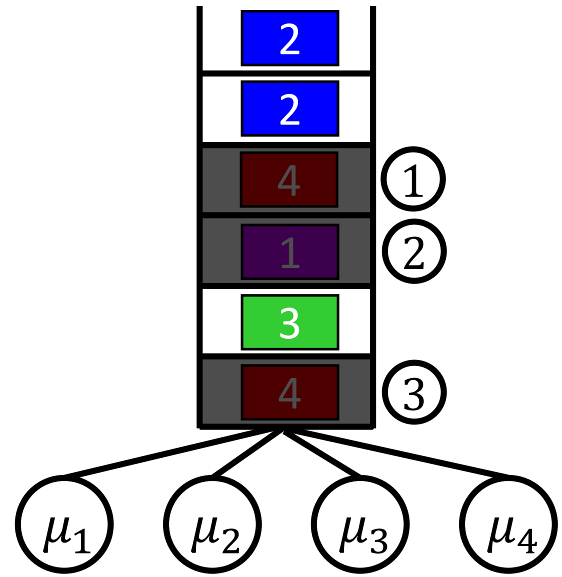


$\vec{c}_n = (3,2,2)$

Noncollaborative (NC) ALIS

$\vec{s}_k = (4)$

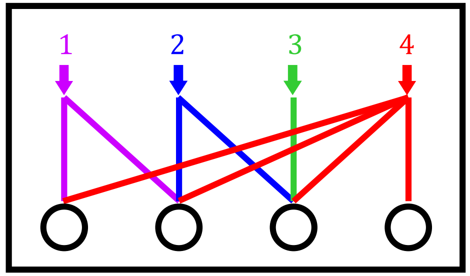




$$\pi(3,2,2) = \pi(0)\left(\frac{\lambda_3}{\mu_3}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)$$

$$\pi(3,2,2;4) = \pi(0;0)\left(\frac{\lambda_3}{\mu_3}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\mu_4}{\lambda_4}\right)$$

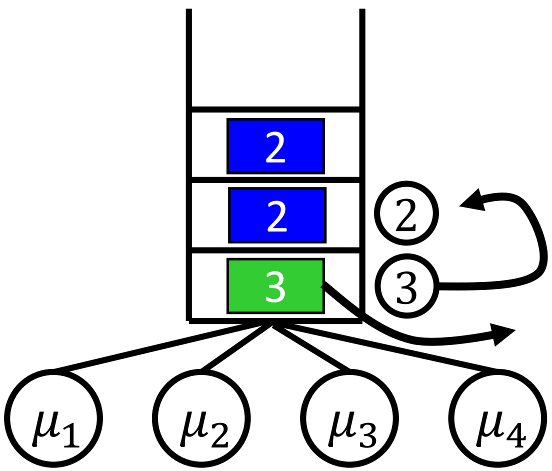
Collaborative vs. Noncollaborative



$$\vec{c}_n = (3,2,2)$$

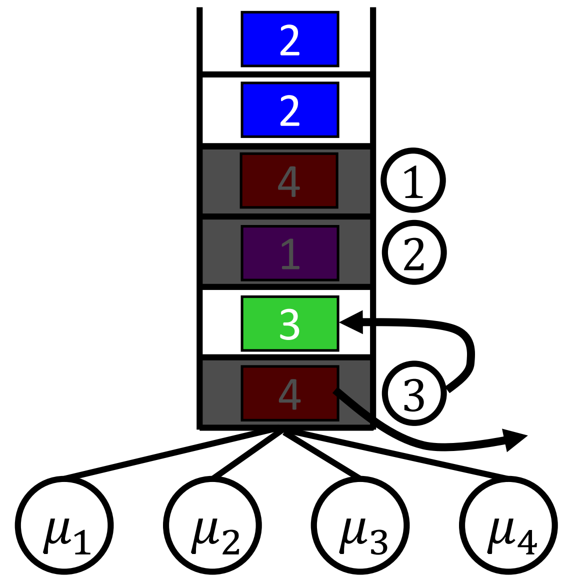
$$\vec{s}_k = (4)$$

Collaborative (C)



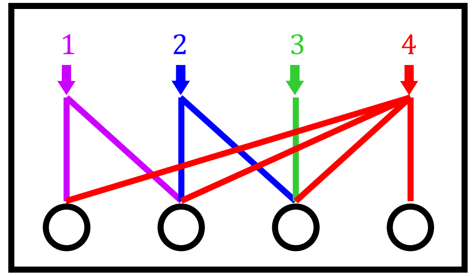
$$\pi(3,2,2) = \pi(0)\left(\frac{\lambda_3}{\mu_3}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)$$

Noncollaborative (NC) ALIS



$$\pi(3,2,2;4) = \pi(0;0)\left(\frac{\lambda_3}{\mu_3}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\mu_4}{\lambda_4}\right)$$

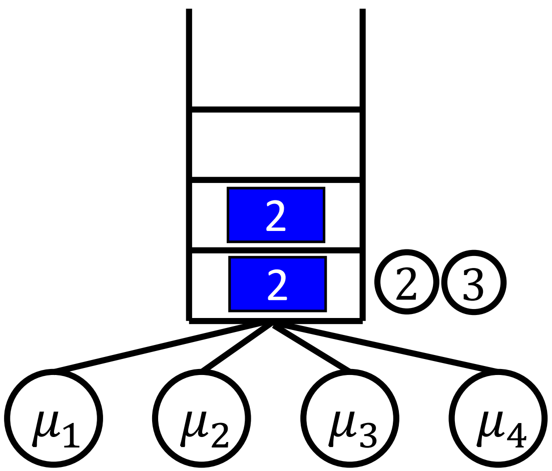
Collaborative vs. Noncollaborative



$$\vec{c}_n = (2,2)$$

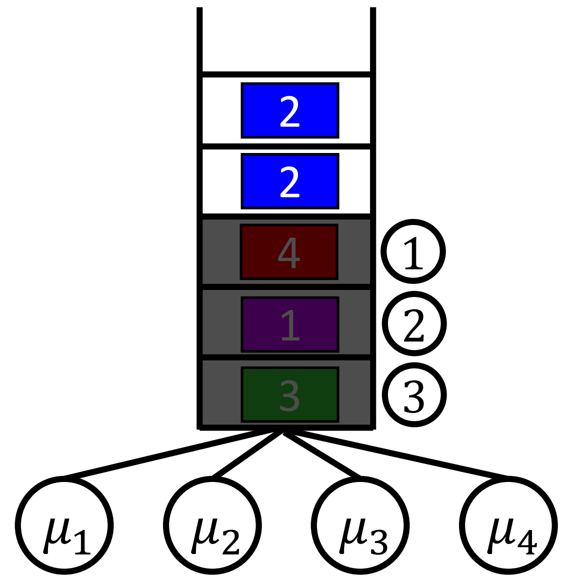
$$\vec{s}_k = (4)$$

Collaborative (C)



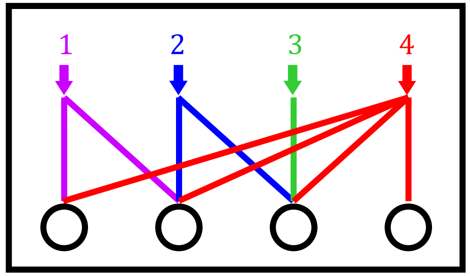
$$\pi(2,2) = \pi(0)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)$$

Noncollaborative (NC) ALIS



$$\pi(2,2;4) = \pi(0;0)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\mu_4}{\lambda_4}\right)$$

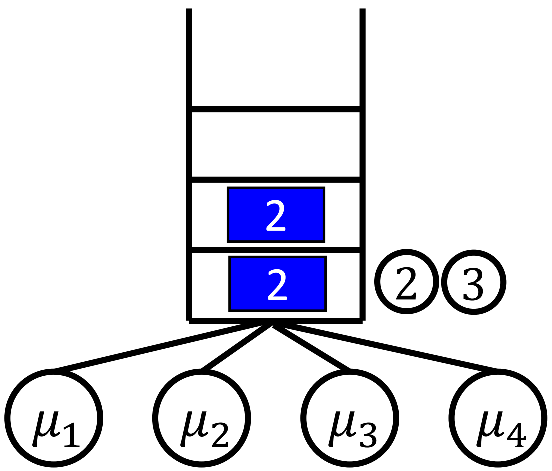
Collaborative vs. Noncollaborative



$\vec{c}_n = (2,2)$

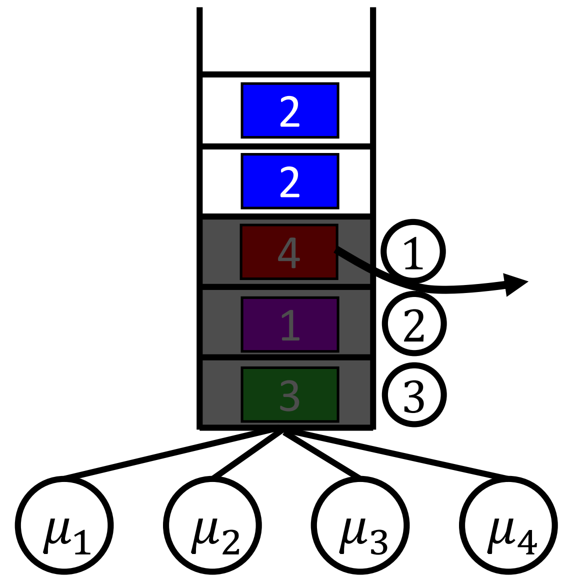
$\vec{s}_k = (4)$

Collaborative (C)



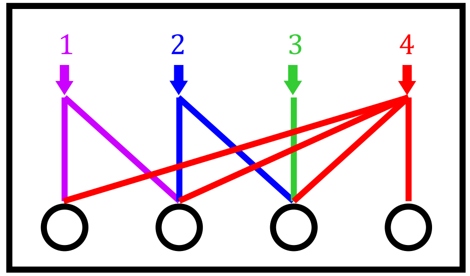
$$\pi(2,2) = \pi(0)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)$$

Noncollaborative (NC) ALIS



$$\pi(2,2;4) = \pi(0;0)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\mu_4}{\lambda_4}\right)$$

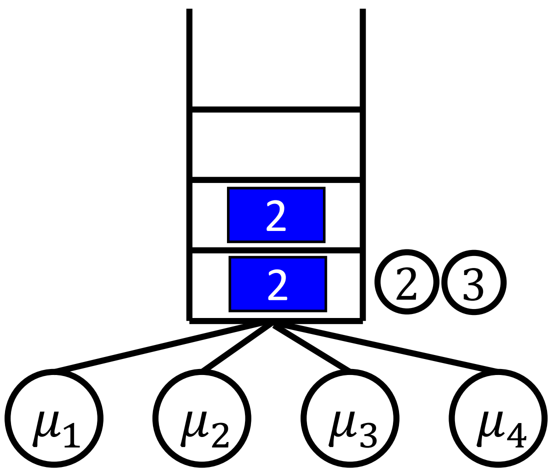
Collaborative vs. Noncollaborative



$\vec{c}_n = (2,2)$

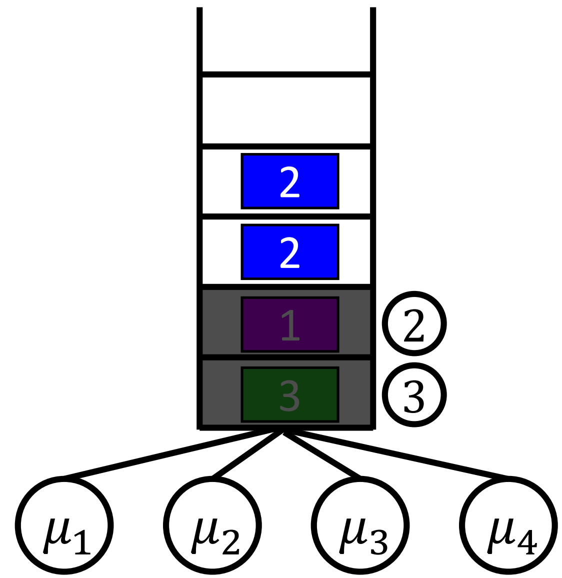
$\vec{s}_k = (4,1)$

Collaborative (C)



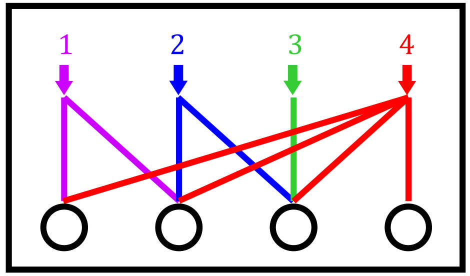
$$\pi(2,2) = \pi(0)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)$$

Noncollaborative (NC) ALIS



$$\pi(2,2;4,1) = \pi(0;0)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\mu_4}{\lambda_4}\right)\left(\frac{\mu_{1,4}}{\lambda_{1,4}}\right)$$

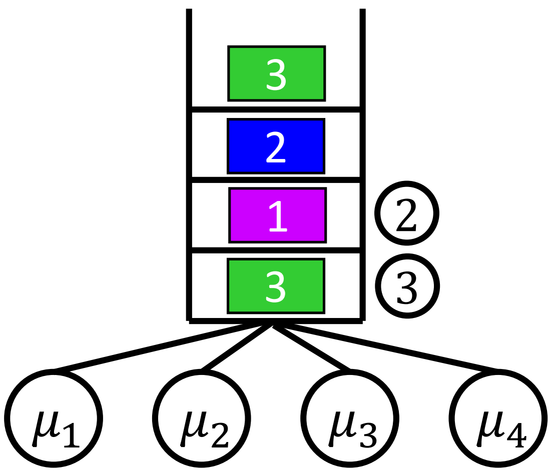
Collaborative vs. Noncollaborative



$\vec{c}_n = (3,1,2,3)$

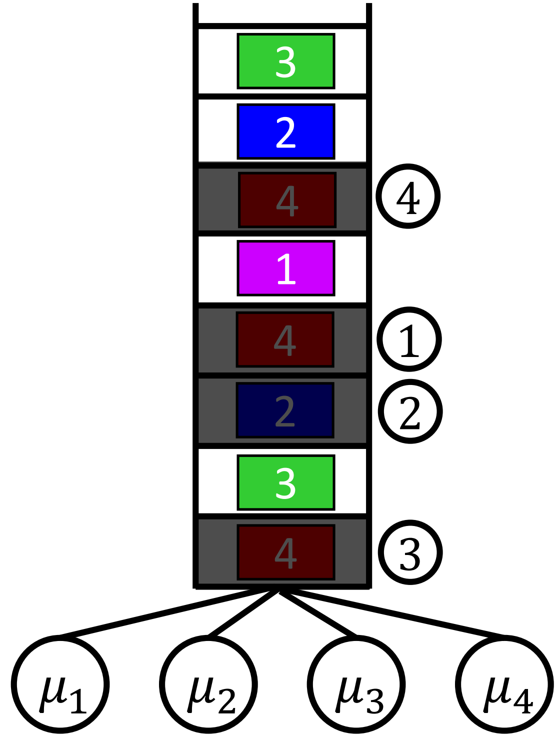
All servers are busy in noncollaborative system

Collaborative (C)



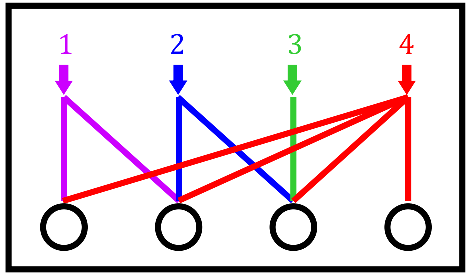
$$\pi(3,1,2,3) = \pi(0)\left(\frac{\lambda_3}{\mu_3}\right)\left(\frac{\lambda_1}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_3}{\mu_{2,3}}\right)$$

Noncollaborative (NC) ALIS



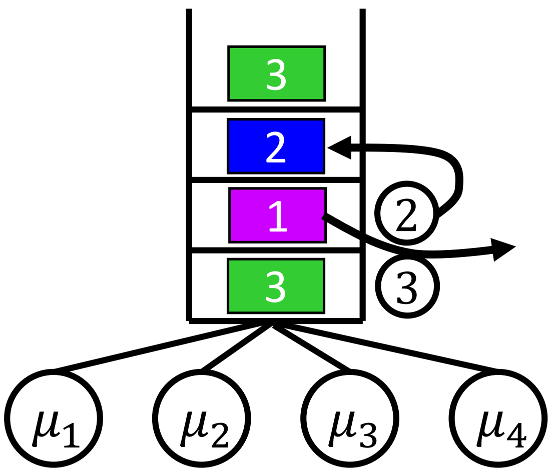
$$\pi(3,1,2,3;0) = \pi(0;0)\left(\frac{\lambda_3}{\mu_3}\right)\left(\frac{\lambda_1}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_3}{\mu_{2,3}}\right)$$

Collaborative vs. Noncollaborative



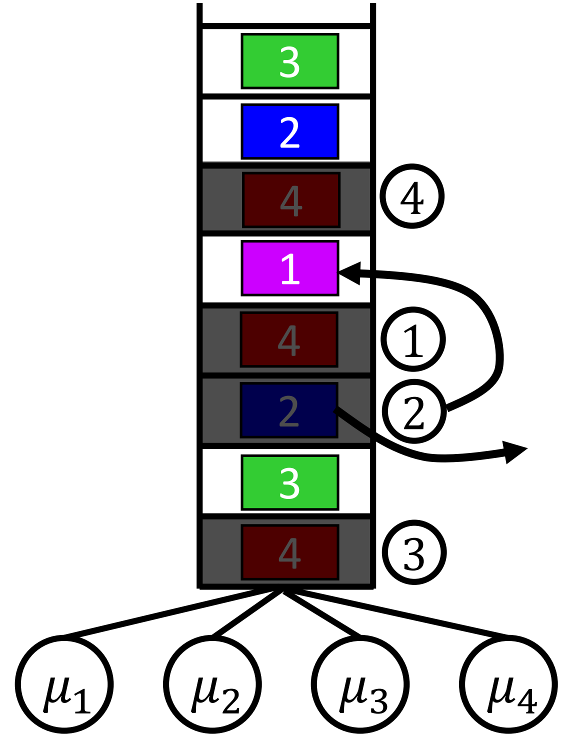
$\vec{c}_n = (3,1,2,3)$

Collaborative (C)



All servers are busy in noncollaborative system

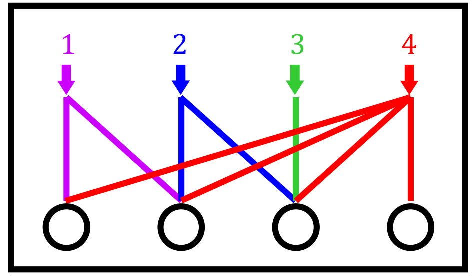
Noncollaborative (NC) ALIS



$$\pi(3,1,2,3) = \pi(0)\left(\frac{\lambda_3}{\mu_3}\right)\left(\frac{\lambda_1}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_3}{\mu_{2,3}}\right)$$

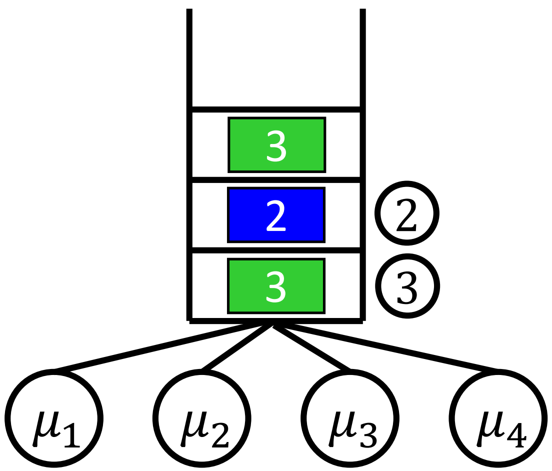
$$\pi(3,1,2,3;0) = \pi(0;0)\left(\frac{\lambda_3}{\mu_3}\right)\left(\frac{\lambda_1}{\mu_{2,3}}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_3}{\mu_{2,3}}\right)$$

Collaborative vs. Noncollaborative



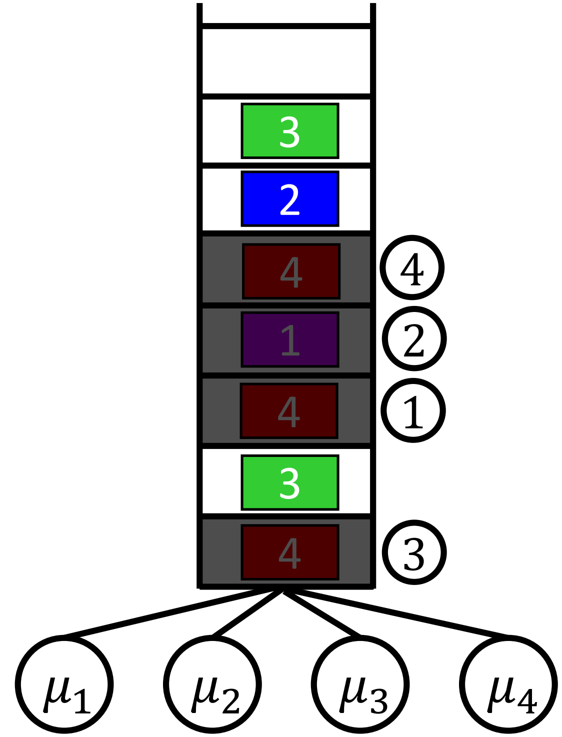
$\vec{c}_n = (3,2,3)$

Collaborative (C)



All servers are busy in noncollaborative system

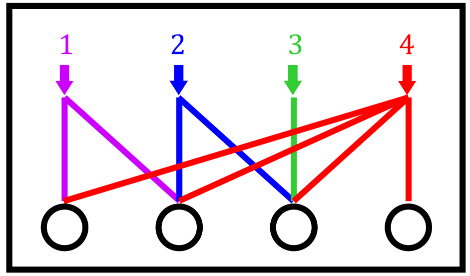
Noncollaborative (NC) ALIS



$$\pi(3,2,3) = \pi(0)\left(\frac{\lambda_3}{\mu_3}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_3}{\mu_{2,3}}\right)$$

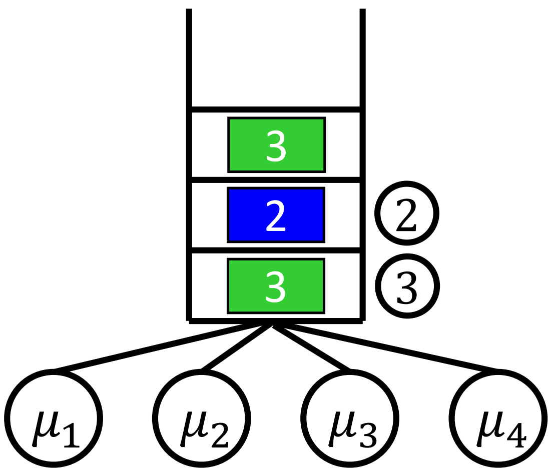
$$\pi(3,2,3;0) = \pi(0;0)\left(\frac{\lambda_3}{\mu_3}\right)\left(\frac{\lambda_2}{\mu_{2,3}}\right)\left(\frac{\lambda_3}{\mu_{2,3}}\right)$$

Collaborative vs. Noncollaborative

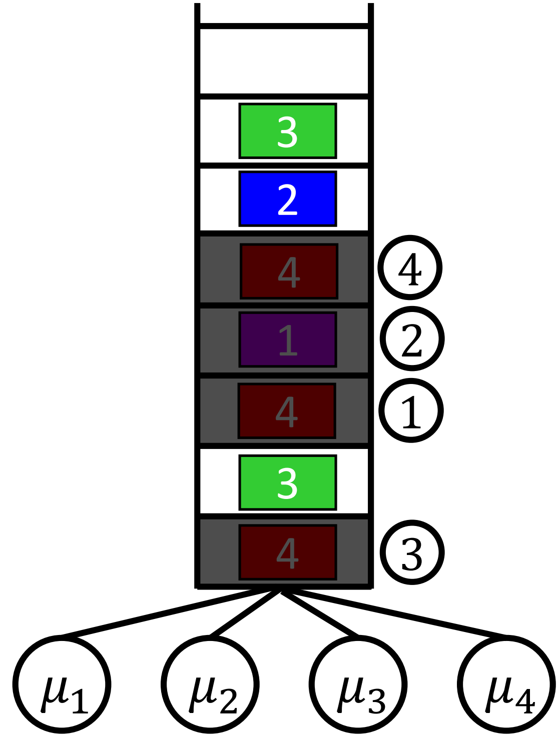


$\vec{c}_n = (3,2,3)$

Collaborative (C)



Noncollaborative (NC) ALIS



Conditioned on all servers being busy in NC, the noncollaborative **queue** has the same stationary distribution as the collaborative **system**

Outline

- System description
- Review: reversibility
- Detailed states

  Collaborative model　　　　Noncollaborative model

  Relating the models

- **Partial aggregation**

  Collaborative model　　　　Noncollaborative model

  - Special case: fully flexible class
  - Special case: nested systems

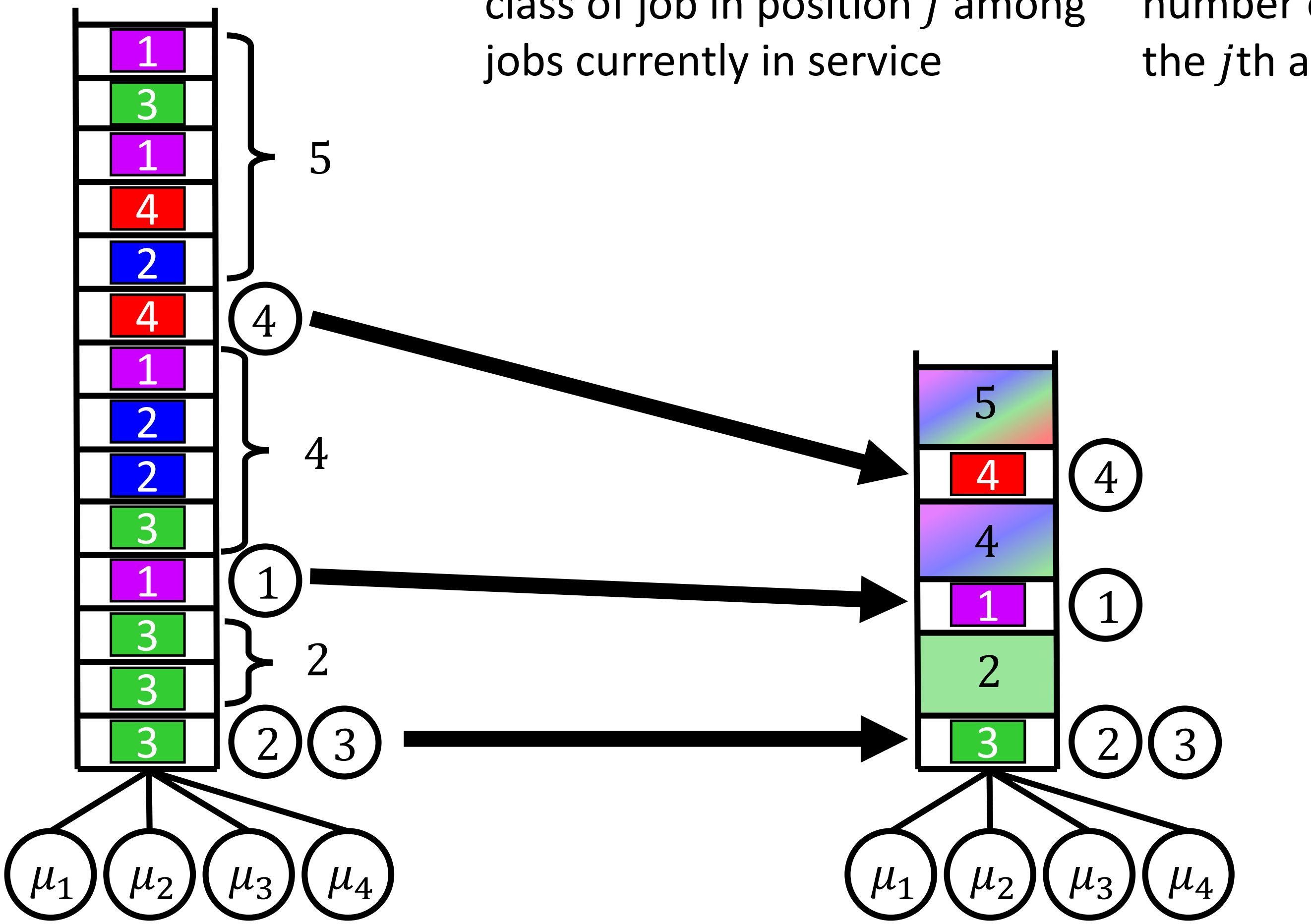
- Related product-form systems

Partial Aggregation: Collaborative Model

$$\text{state } \left(\vec{d}_l;\vec{n}_l\right) = \left(d_1,\dots,d_j,\dots,d_l;n_1,\dots,n_j,\dots,n_l\right)$$

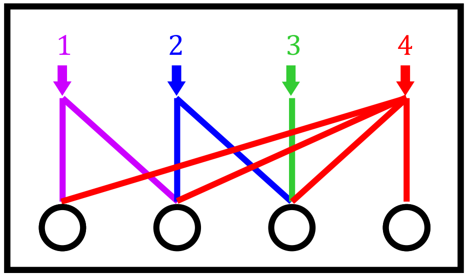
class of job in position j among jobs currently in service

number of jobs in the queue between the jth and $(j+1)$st jobs in service



$\vec{c}_n = (3,3,3,1,3,2,2,1,4,2,4,1,3,1)$

$\left(\vec{d}_l, n_l\right) = (3,1,4;2,4,5)$



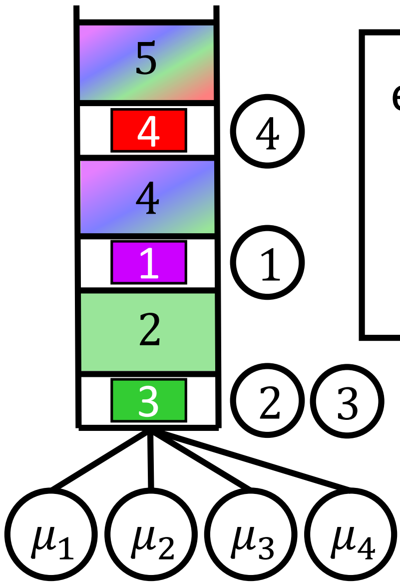
Partial Aggregation: Collaborative Model

Thm:

$$\pi\left(\vec{d}_l, \vec{n}_l\right) = \pi(0) \prod_{j=1}^{l} \frac{\lambda_{d_j}}{\mu\left(\vec{d}_j\right)} \left(\frac{\lambda\left(R\left(\vec{d}_j\right)\right)}{\mu\left(\vec{d}_j\right)} \right)^{n_j}$$

where $\pi(0)$ is the probability that the system is empty

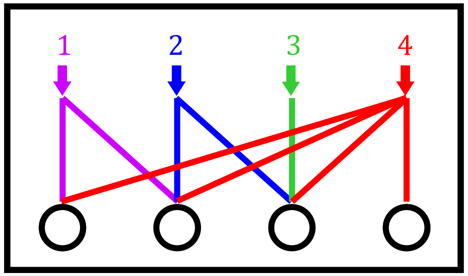
$$R\left(\vec{d}_j\right) = \left\{ i \,\middle|\, \text{class } i \textbf{ R}\text{equires a server in } S\left(\vec{d}_j\right) \right\}$$



$\left(\vec{d}_l, n_l\right) = (3,1,4;2,4,5)$

ex. state $\left(\vec{d}_3, n_3\right) = (3,1,4;2,4,5)$

$$\pi\left(\vec{d}_3, n_3\right) = \pi(0) \left(\frac{\lambda_3}{\mu_{2,3}}\right) \left(\frac{\lambda_3}{\mu_{2,3}}\right)^2 \left(\frac{\lambda_1}{\mu_{1,2,3}}\right) \left(\frac{\lambda_{1,2,3}}{\mu_{1,2,3}}\right)^4 \left(\frac{\lambda_4}{\mu}\right) \left(\frac{\lambda}{\mu}\right)^5$$



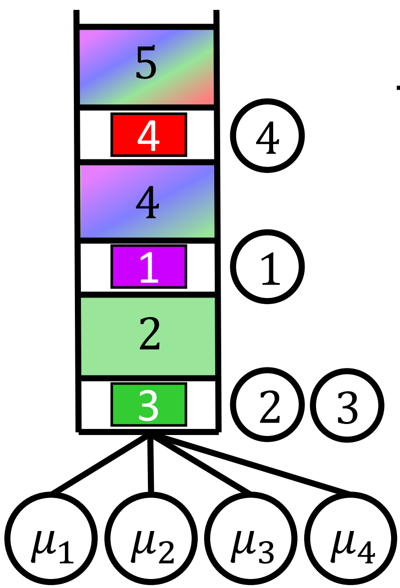
Partial Aggregation: Collaborative Model

Thm:

$$\pi\left(\vec{d}_l, \vec{n}_l\right) = \pi(0) \prod_{j=1}^{l} \frac{\lambda_{d_j}}{\mu\left(\vec{d}_j\right)} \left(\frac{\lambda\left(R\left(\vec{d}_j\right)\right)}{\mu\left(\vec{d}_j\right)} \right)^{n_j}$$

where $\pi(0)$ is the probability that the system is empty

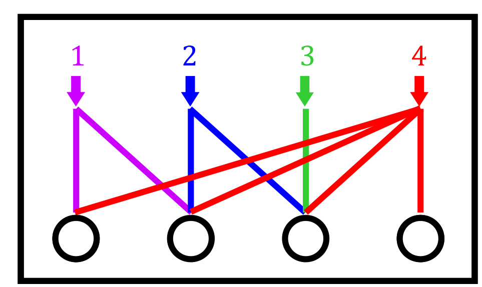
$$R\left(\vec{d}_j\right) = \left\{ i \middle| \text{class } i \textbf{ R}\text{equires a server in } S\left(\vec{d}_j\right) \right\}$$



Proof: Approach 1: Partially aggregated states satisfy partial balance

Approach 2: Sum up detailed states:

$$\pi\left(\vec{d}_l, \vec{n}_l\right) = \sum_{\substack{\vec{c}_n \text{ consistent} \\ \text{with } (\vec{d}_l, \vec{n}_l)}} \pi(\vec{c}_n)$$



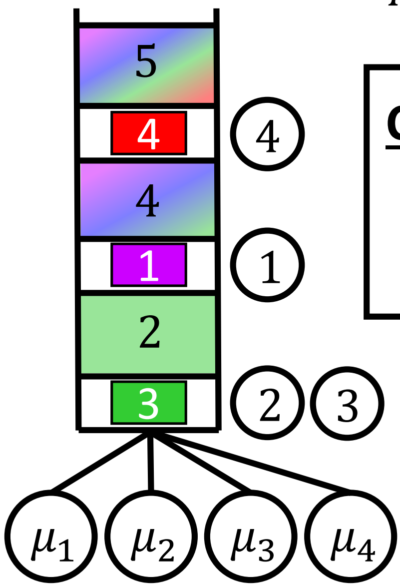
Partial Aggregation: Collaborative Model

Thm:

$$\pi\left(\vec{d}_l, \vec{n}_l\right) = \pi(0) \prod_{j=1}^{l} \frac{\lambda_{d_j}}{\mu\left(\vec{d}_j\right)} \left(\frac{\lambda\left(R\left(\vec{d}_j\right)\right)}{\mu\left(\vec{d}_j\right)} \right)^{n_j}$$

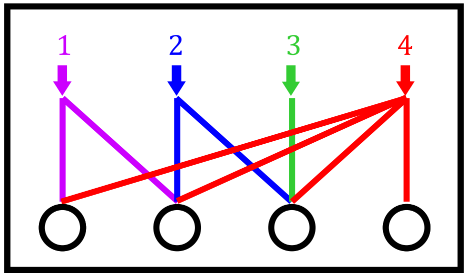
where $\pi(0)$ is the probability that the system is empty

N_j = # jobs between jth and $(j+1)$st jobs in service



Corollary:

$$N_j \left| \vec{d}_l =_{st} N_j \right| \vec{d}_j \sim \text{Geom}\left(1 - \frac{\lambda\left(R\left(\vec{d}_j\right)\right)}{\mu\left(\vec{d}_j\right)}\right)$$

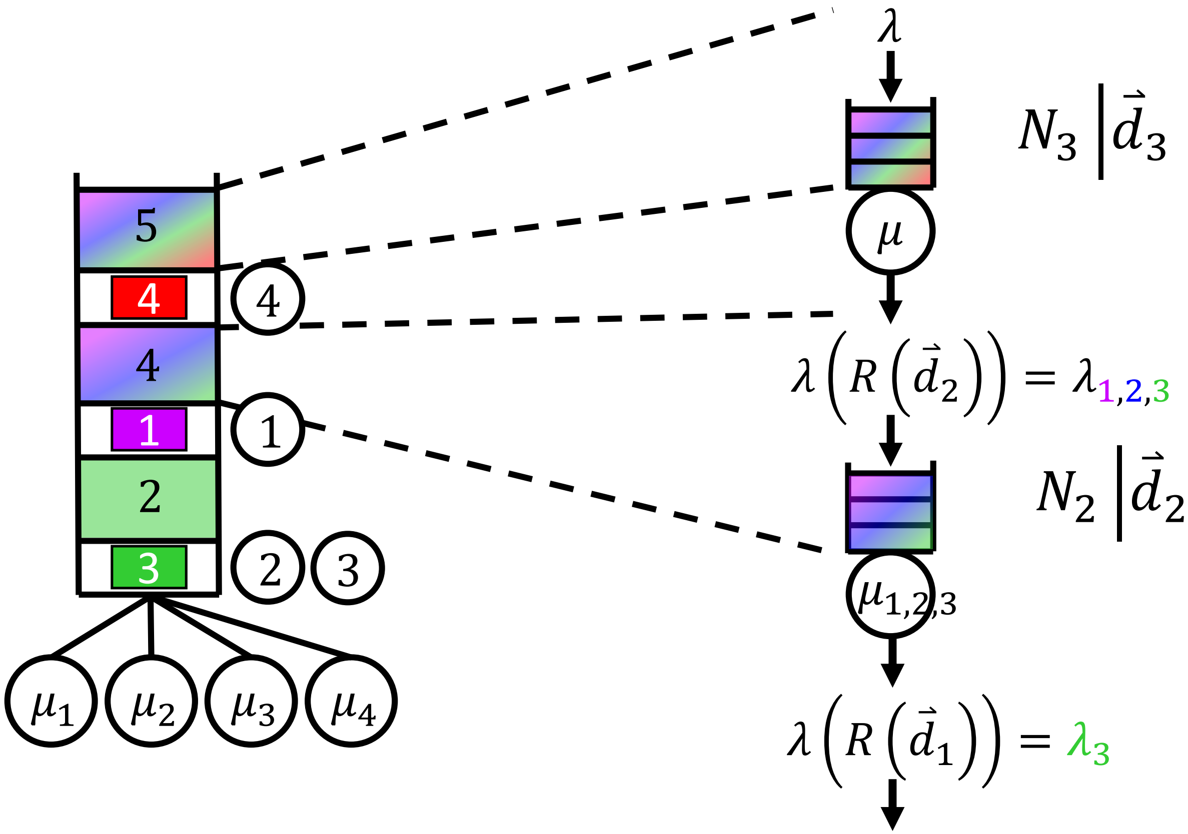


Partial Aggregation: Collaborative Model

Thm:

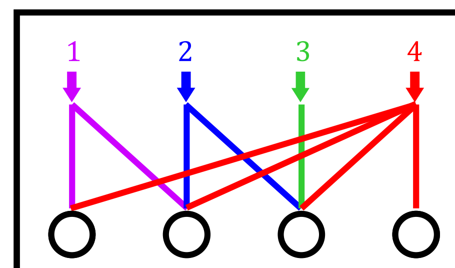
$$\pi\left(\vec{d}_l, \vec{n}_l\right) = \pi(0) \prod_{j=1}^{l} \frac{\lambda_{d_j}}{\mu\left(\vec{d}_j\right)} \left(\frac{\lambda\left(R\left(\vec{d}_j\right)\right)}{\mu\left(\vec{d}_j\right)} \right)^{n_j}$$

where $\pi(0)$ is the probability that the system is empty



$$N_3 \,\middle|\, \vec{d}_3 \sim \mathrm{Geom}\left(1 - \frac{\lambda}{\mu}\right)$$

$$N_2 \,\middle|\, \vec{d}_2 \sim \mathrm{Geom}\left(1 - \frac{\lambda_{1,2,3}}{\mu_{1,2,3}}\right)$$

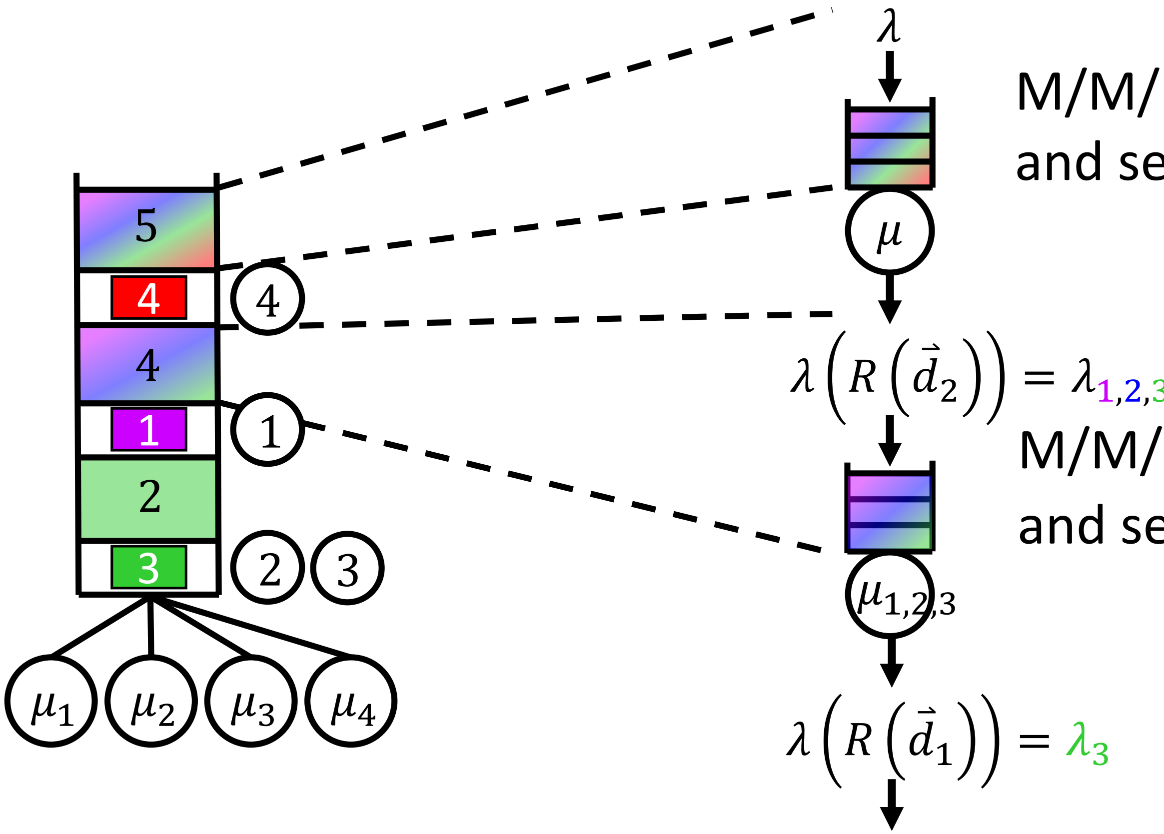


Partial Aggregation: Collaborative Model

Thm:

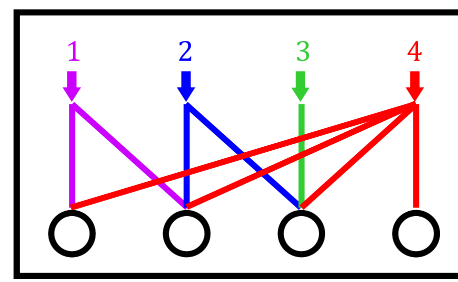
$$\pi\left(\vec{d}_l, \vec{n}_l\right) = \pi(0) \prod_{j=1}^{l} \frac{\lambda_{d_j}}{\mu\left(\vec{d}_j\right)} \left(\frac{\lambda\left(R\left(\vec{d}_j\right)\right)}{\mu\left(\vec{d}_j\right)} \right)^{n_j}$$

where $\pi(0)$ is the probability that the system is empty



M/M/1 with arrival rate λ and service rate μ

M/M/1 with arrival rate $\lambda_{1,2,3}$ and service rate $\mu_{1,2,3,5}$

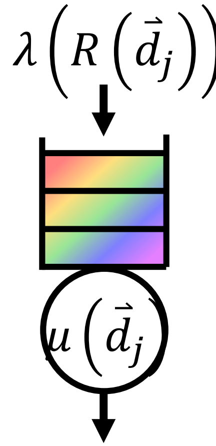


Performance Metrics: Collaborative Model

Corollary:

$$N_j \left| \vec{d}_l =_{st} N_j \right| \vec{d}_j \sim \text{Geom}\left(1 - \frac{\lambda\left(R\left(\vec{d}_j\right)\right)}{\mu\left(\vec{d}_j\right)}\right)$$

$N_j^{(i)}$ = # class-i jobs between jth and $(j+1)$st jobs in service



$$N_j \left| \vec{d}_j \sim \text{Geom}\left(1 - \frac{\lambda\left(R\left(\vec{d}_j\right)\right)}{\mu\left(\vec{d}_j\right)}\right)\right.$$

each is class 3

with probability $\frac{\lambda_3}{\lambda\left(R\left(\vec{d}_j\right)\right)}$

Lemma: If $N \sim \text{Geom}(1-p)$ and $N_j^{(i)} | N \sim \text{Binomial}\left(N, \frac{q_i}{p}\right)$, then

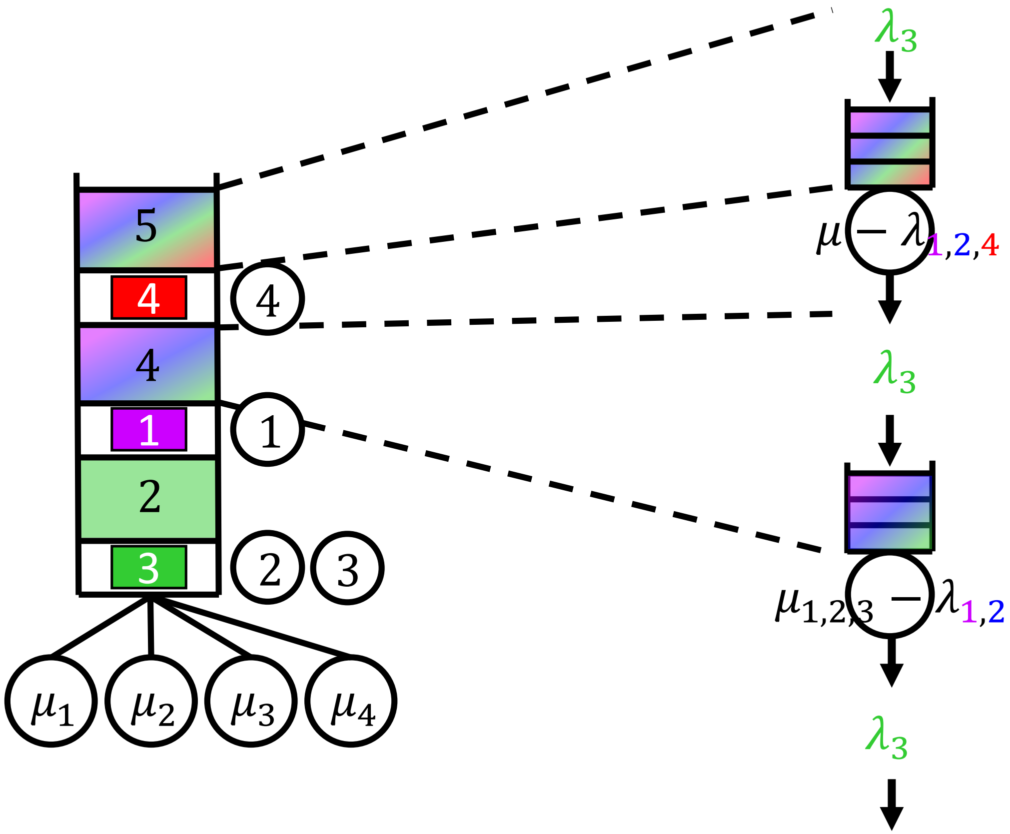
$$N_j^{(i)} \sim \text{Geom}\left(1 - \frac{q_i}{q_i + 1 - p}\right)$$

Let $p = \frac{\lambda\left(R\left(\vec{d}_j\right)\right)}{\mu\left(\vec{d}_j\right)}, \frac{q_i}{p} = \frac{\lambda_i}{\lambda\left(R\left(\vec{d}_j\right)\right)}$

Corollary: $N_j^{(i)} \left| \vec{d}_l =_{st} N_j^{(i)} \right| \vec{d}_j \sim \text{Geom}\left(1 - \frac{\lambda_i}{\mu\left(\vec{d}_j\right) - \lambda\left(R\left(\vec{d}_j\right)\right) + \lambda_i}\right)$

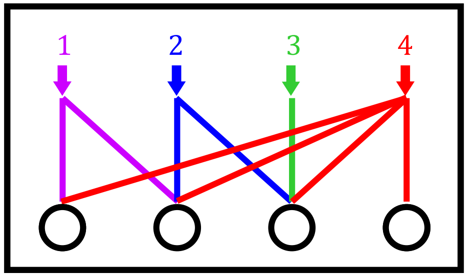
Performance Metrics: Collaborative Model

Corollary: $N_j^{(i)} \,\middle|\, \vec{d}_l =_{st} N_j^{(i)} \,\middle|\, \vec{d}_j \sim \text{Geom}\left(1 - \frac{\lambda_i}{\mu\left(\vec{d}_j\right) - \lambda\left(R\left(\vec{d}_j\right)\right) + \lambda_i}\right)$



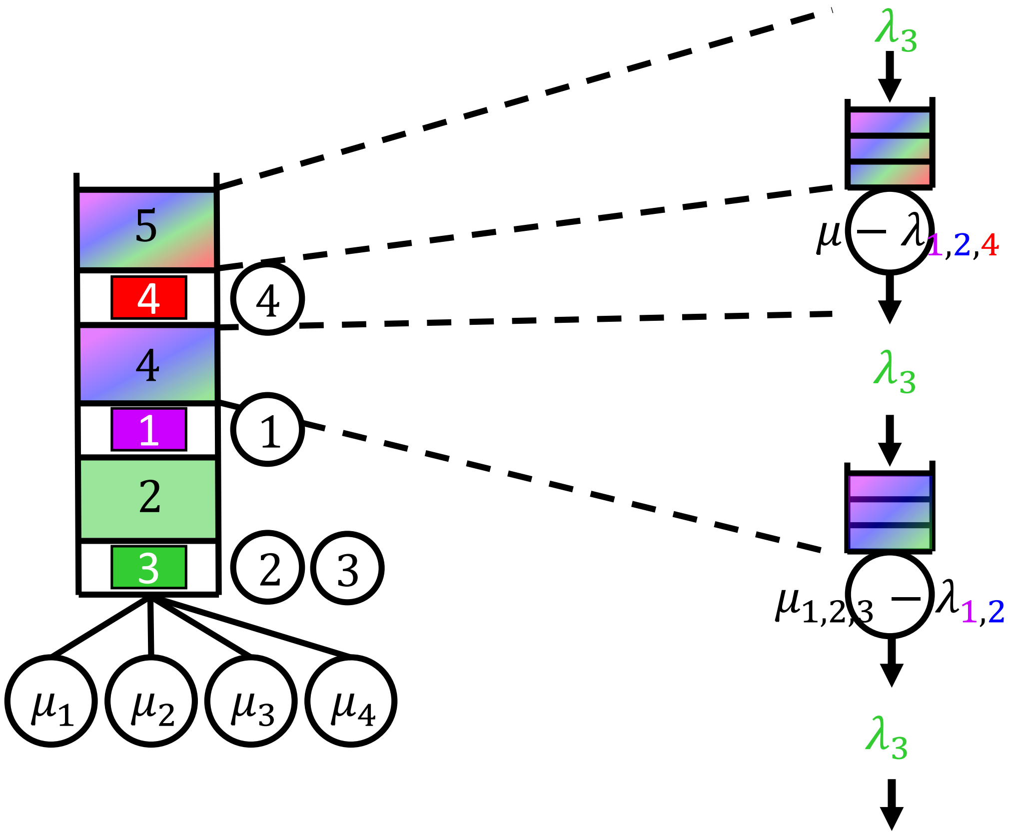
$$N_3^{(3)} \,\middle|\, \vec{d}_3 \sim \text{Geom}\left(1 - \frac{\lambda_3}{\mu - \lambda_{1,2,4}}\right)$$

$$N_2^{(3)} \,\middle|\, \vec{d}_2 \sim \text{Geom}\left(1 - \frac{\lambda_3}{\mu_{1,2,3} - \lambda_{1,2}}\right)$$



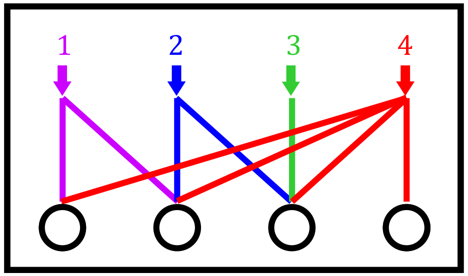
Performance Metrics: Collaborative Model

Corollary: $N_j^{(i)} \,\middle|\, \vec{d}_l =_{st} N_j^{(i)} \,\middle|\, \vec{d}_j \sim \text{Geom}\left(1 - \frac{\lambda_i}{\mu\left(\vec{d}_j\right) - \lambda\left(R\left(\vec{d}_j\right)\right) + \lambda_i}\right)$



Class 3: M/M/1 with arrival rate λ_3 and service rate $\mu - \lambda_{1,2,4}$

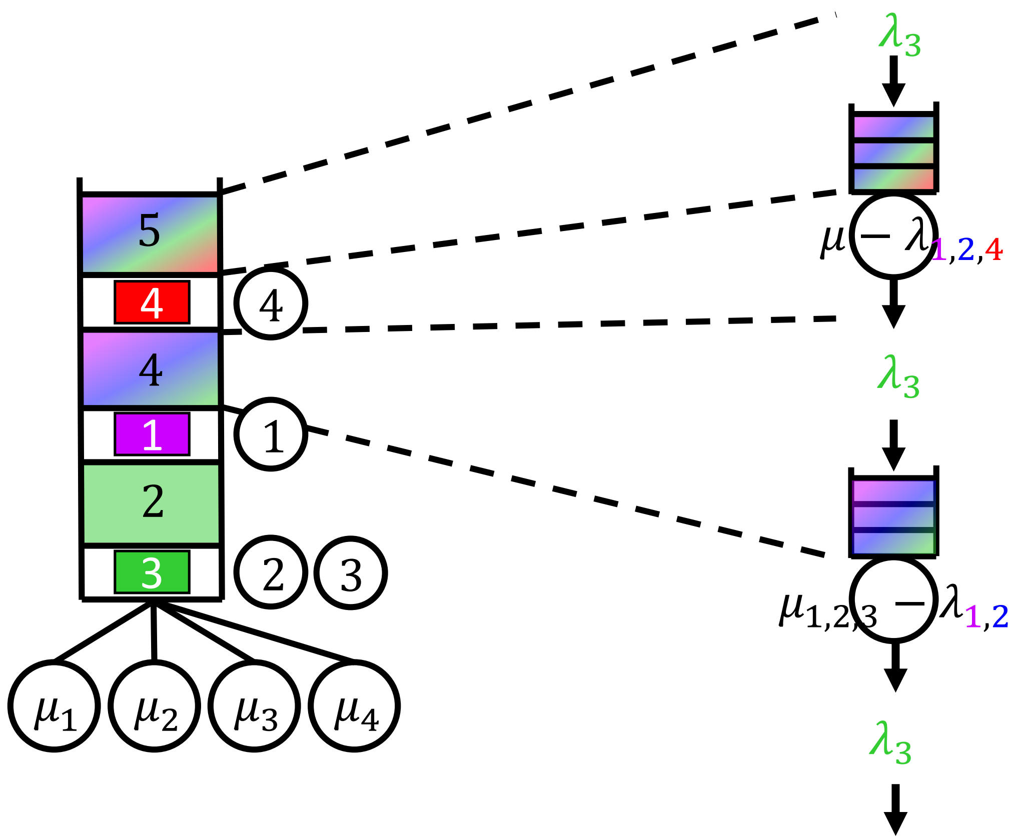
Class 3: M/M/1 with arrival rate λ_3 and service rate $\mu_{1,2,3} - \lambda_{1,2}$



Performance Metrics: Collaborative Model

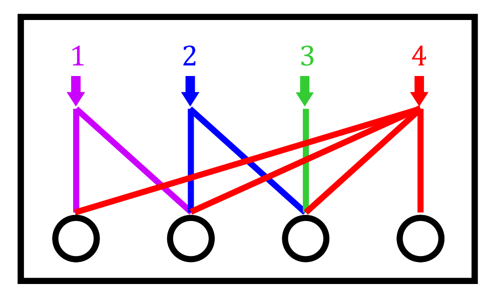
$W_Q^{(i)}\left(\vec{d}_l\right)$ = time in queue for a class-i job, given it sees service profile $\vec{d}_l$

$W_Q^{M/M/1}(\lambda, \mu)$ = time in queue in an M/M/1 with arrival rate λ and service rate μ



$W_Q^{M/M/1}(\lambda_3, \mu - \lambda_{1,2,4})$

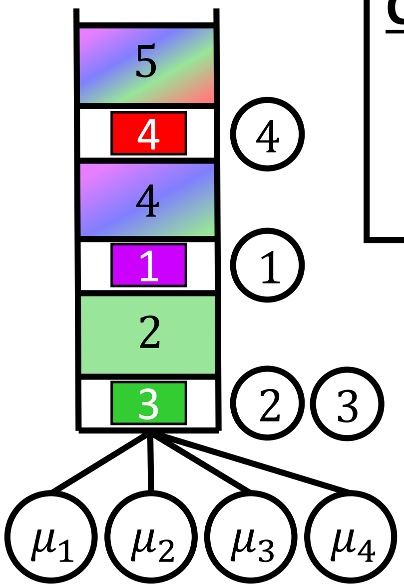
$W_Q^{M/M/1}(\lambda_3, \mu_{1,2,3} - \lambda_{1,2})$



Performance Metrics: Collaborative Model

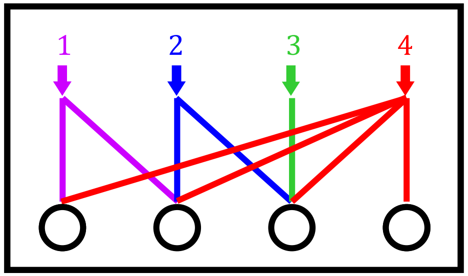
$W_Q^{(i)}\left(\vec{d}_l\right)$ = time in queue for a class-i job, given it sees service profile $\vec{d}_l$

$W_Q^{M/M/1}(\lambda,\mu)$ = time in queue in an M/M/1 with arrival rate λ and service rate μ



Corollary:

$$W_Q^{(i)}\left(\vec{d}_l\right) =_{st} \sum_{j:i\in R\left(\vec{d}_j\right)} W_Q^{M/M/1}\left(\lambda_i, \mu\left(\vec{d}_j\right) - \lambda\left(R\left(\vec{d}_j\right)\right) + \lambda_i\right)$$



Outline

- System description
- Review: reversibility
- Detailed states

  Collaborative model   Noncollaborative model

  Relating the models
- **Partial aggregation**

  Collaborative model   **Noncollaborative model**

  - Special case: fully flexible class
  - Special case: nested systems
- Related product-form systems

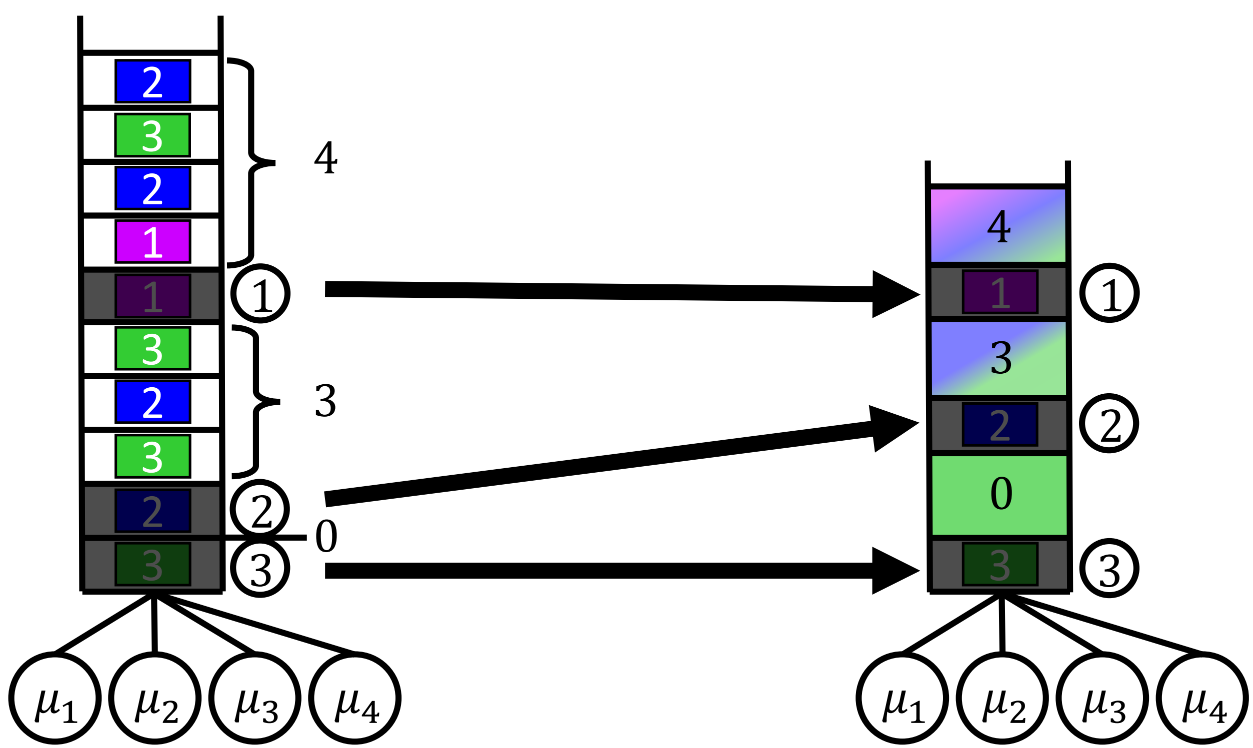
Partial Aggregation: Noncollaborative ALIS

$$\text{state}\left(\vec{n}_l; \vec{b}_l; \vec{s}_{M-l}\right) = \left(n_1, \dots, n_j, \dots, n_l; b_1, \dots, b_j, \dots, b_l; s_1, \dots, s_j, \dots, s_{M-l}\right)$$

number of jobs in the queue between the jth and $(j+1)$st jobs in service

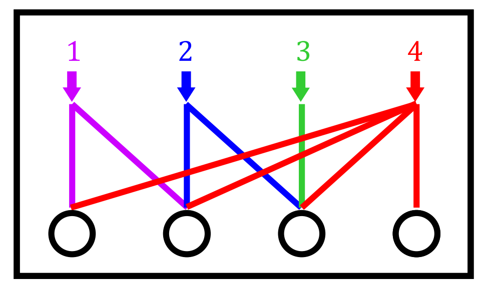
server that has been busy jth longest

server that has been idle jth longest



$\vec{c}_n = (3,3,3,1,2,3,2)$

$\left(\vec{n}_l; \vec{b}_l; \vec{s}_{M-l}\right) = (0,3,4; 3,2,1; 4)$



Partial Aggregation: Noncollaborative

Thm: Under ALIS,

$$\pi\left(\vec{n}_l; \vec{b}_l; \vec{s}_{M-l}\right) = \pi(0; \{b_1, \dots, b_M\}; 0) \prod_{j=1}^{l} \left(\frac{\lambda\left(R\left(\vec{b}_j\right)\right)}{\mu\left(\vec{b}_j\right)}\right)^{n_j} \prod_{j=1}^{l} \frac{1}{\mu\left(\vec{b}_j\right)} \prod_{j=1}^{M-l} \frac{1}{\lambda\left(\vec{s}_j\right)}$$

where $\vec{b}_l$ denotes the set of busy servers ordered by the arrival times of the jobs they are serving.

Thm: Under random assignment (with activation rate condition),

$$\pi\left(\vec{n}_l; \vec{b}_l\right) = \pi(0; 0) \prod_{j=1}^{l} \left(\frac{\lambda\left(R\left(\vec{b}_j\right)\right)}{\mu\left(\vec{b}_j\right)}\right)^{n_j} \prod_{j=1}^{l} \frac{\lambda^A_{b_j}\left(\vec{b}_{j-1}\right)}{\mu\left(\vec{b}_j\right)}$$

where $\vec{b}_l$ denotes the set of busy servers ordered by the arrival times of the jobs they are serving and $\lambda^A_{b_j}\left(\vec{b}_{j-1}\right)$ denotes the activation rate of server b_j, given $\vec{b}_{j-1}$.

[Adan, Visschers, Weiss]

Outline

- System description
- Review: reversibility
- Detailed states

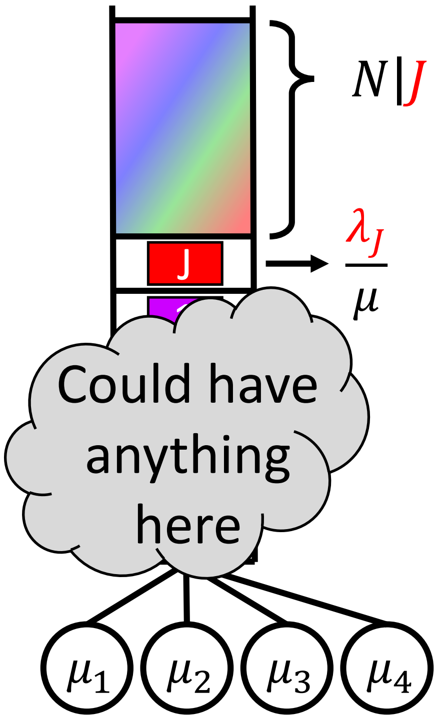
  Collaborative model — Noncollaborative model

  Relating the models
- **Partial aggregation**

  Collaborative model — Noncollaborative model

  - **Special case: fully flexible class**
  - Special case: nested systems
- Related product-form systems

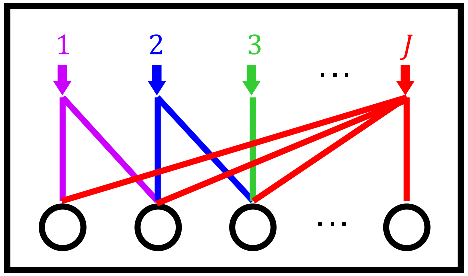
Fully Flexible Class: Collaborative Model



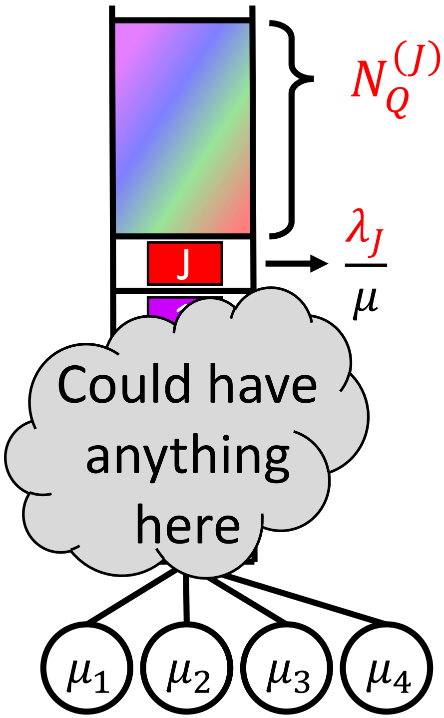
can aggregate over all classes

$$N|J \text{ in system} \sim \text{Geom}\left(1 - \frac{\lambda}{\mu}\right)$$

$\frac{\lambda_J}{\mu}$



Fully Flexible Class: Collaborative Model



$N_Q^{(J)} | J \text{ in system} \sim \text{Geom}\left(1 - \frac{\lambda_J}{\mu - \lambda + \lambda_J}\right)$ (by the same lemma used for partial aggregation)

$$N^{(J)} = \Pr\{J \text{ in system}\} \cdot \left(1 + N_Q^{(J)} | J \text{ in system}\right)$$

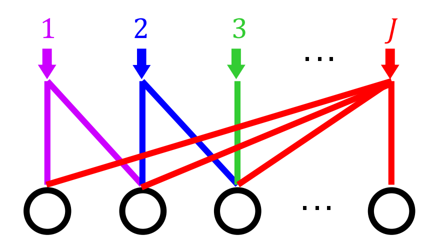
$$\sim \text{Geom}\left(1 - \frac{\lambda_J}{\mu - \lambda + \lambda_J}\right)$$

as long as $\Pr\{J \text{ in system}\} = \rho_J = \frac{\lambda_J}{\mu - \lambda + \lambda_J}$

(short proof, won't show here)

Fully flexible class experiences M/M/1 response time:

$$W^J =_{st} W^{M/M/1}(\lambda_J, \mu - \lambda + \lambda_J) =_{st} W^{M/M/1}(\lambda, \mu)$$



Outline

- System description
- Review: reversibility
- Detailed states

  Collaborative model Noncollaborative model

  Relating the models
- **Partial aggregation**

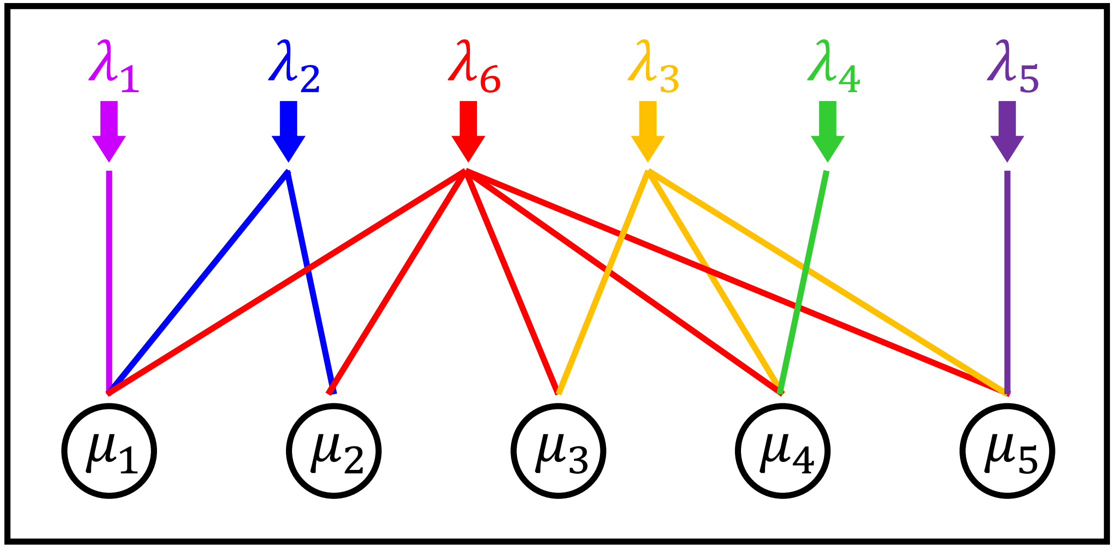
  Collaborative model Noncollaborative model

  - Special case: fully flexible class
  - **Special case: nested systems**
- Related product-form systems

Nested Systems: Collaborative Model

For every pair of job classes i and j, either:

- $S_i \subset S_j$
- $S_j \subset S_i$
- $S_i \cap S_j = \emptyset$

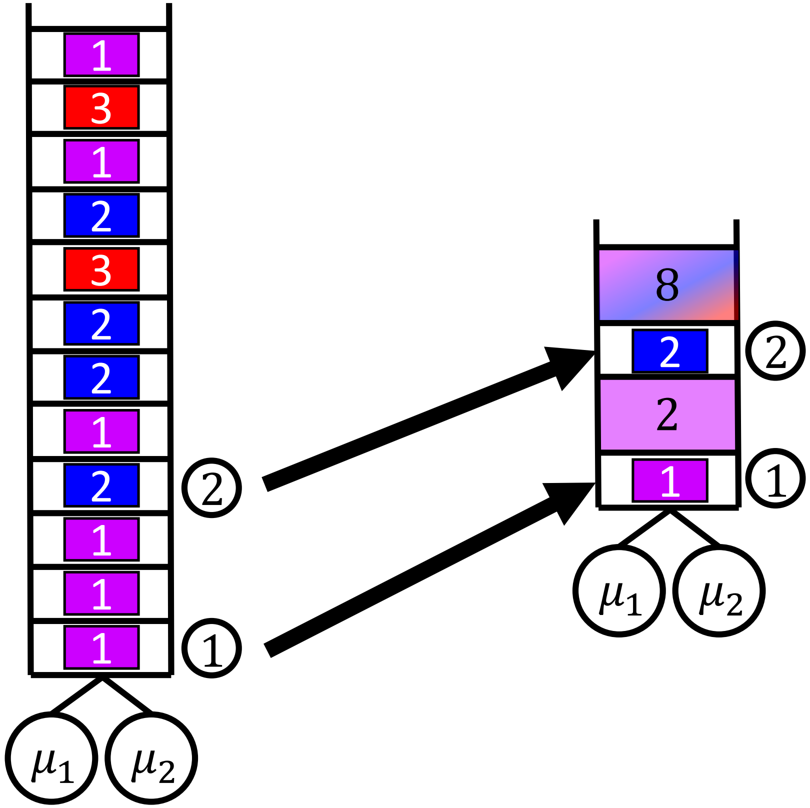


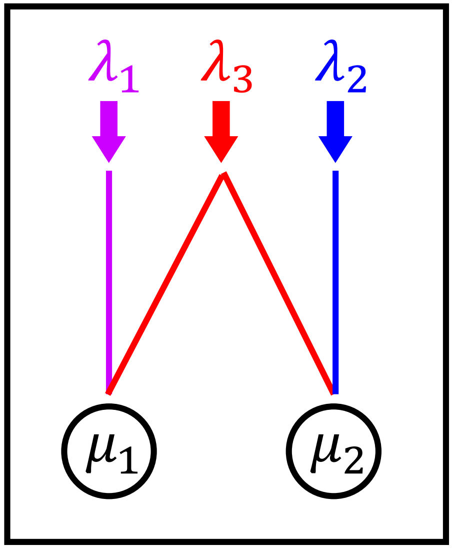
Recursive structure: removing the fully flexible class leaves two smaller nested systems

W System: Collaborative Model

Detailed states

Partially Aggregated States



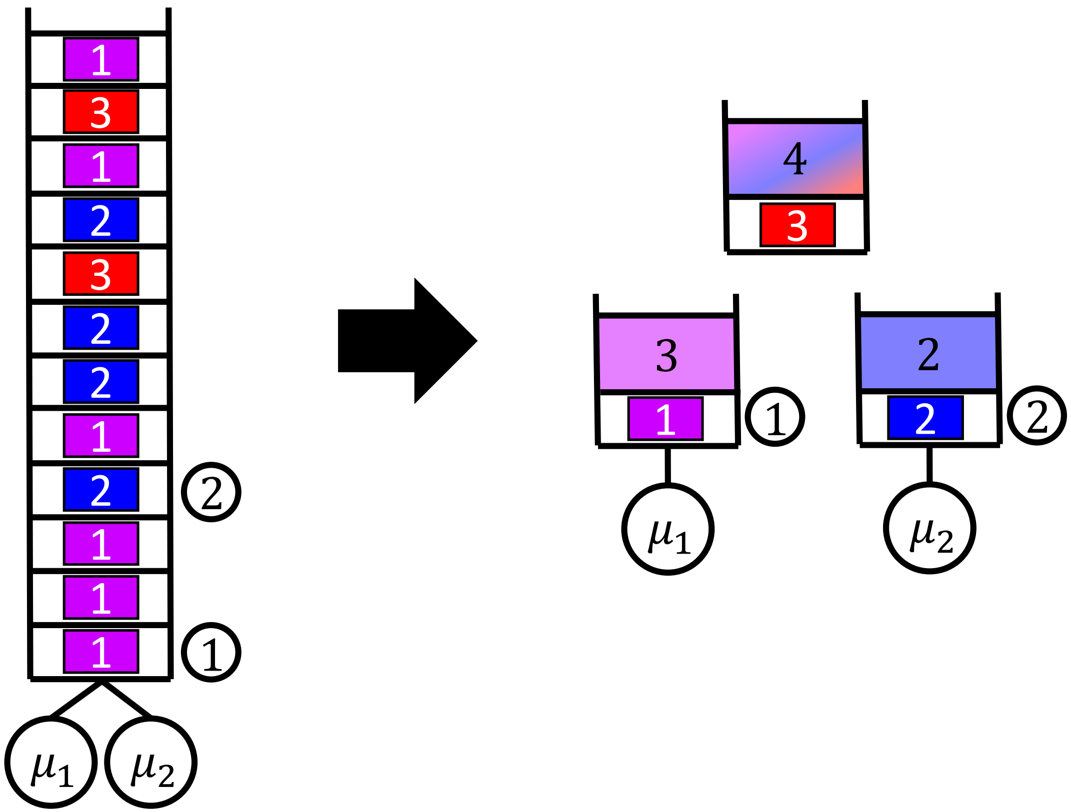


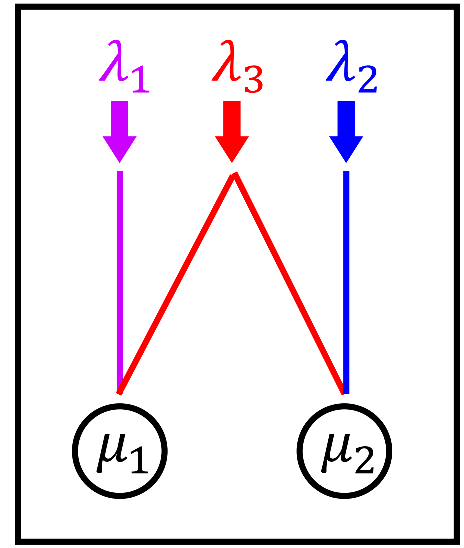
Observation: a class 1 job is never blocked from service by a class 2 job

W System: Collaborative Model

Detailed states

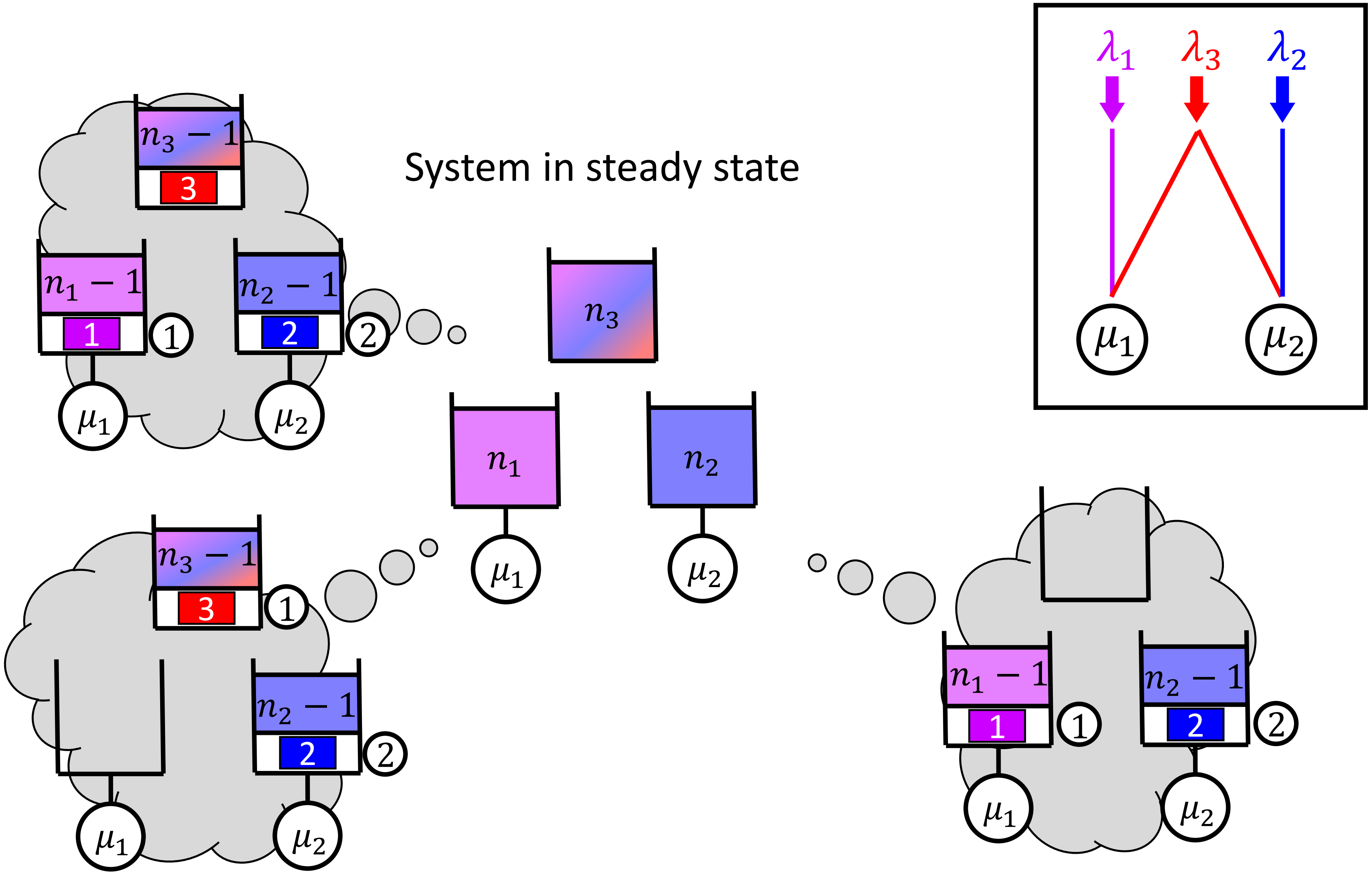
Partially Aggregated States for Nested Systems





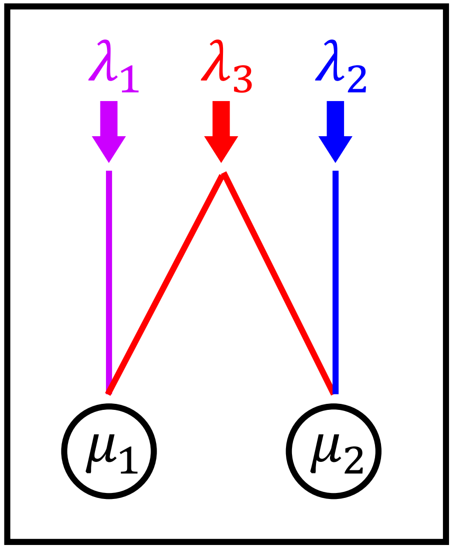
Observation: a class 1 job is never blocked from service by a class 2 job

W System: Collaborative Model



W System: Collaborative Model

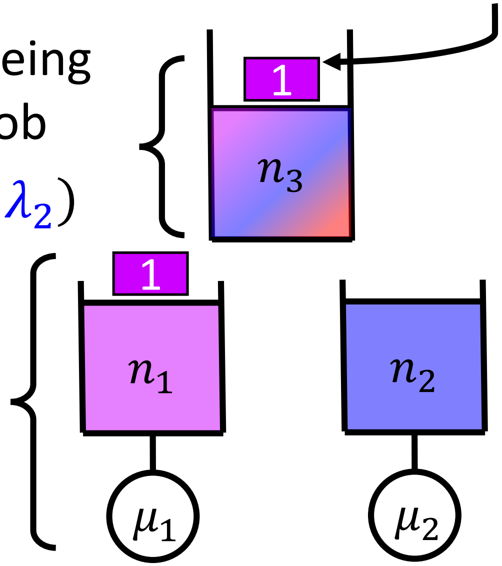
Understanding **class 1** response time



tagged **class 1** arrival

experiences queueing time of a **class 3** job

$W^Q(\lambda_3, \mu - \lambda_1 - \lambda_2)$



experiences response time of a **class 1** job

$W(\lambda_1, \mu_1)$

$$W_1 \sim W^Q(\lambda_3, \mu - \lambda_1 - \lambda_2) + W(\lambda_1, \mu_1)$$

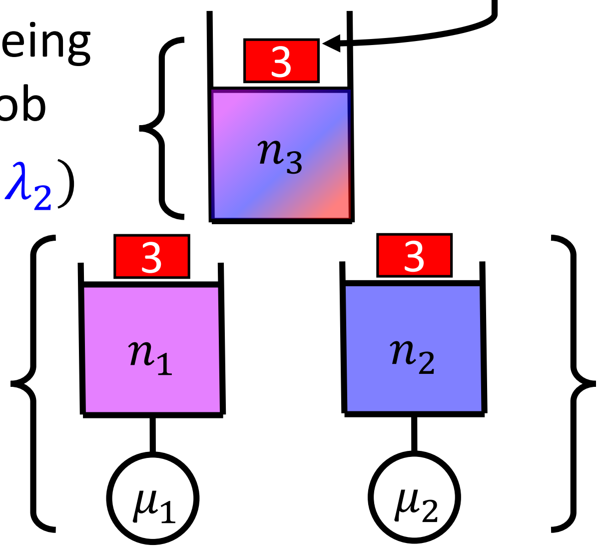
W System: Collaborative Model

Understanding **class 3** response time

tagged **class 3** arrival

experiences queueing time of a **class 3** job

$W^Q(\lambda_3, \mu - \lambda_1 - \lambda_2)$

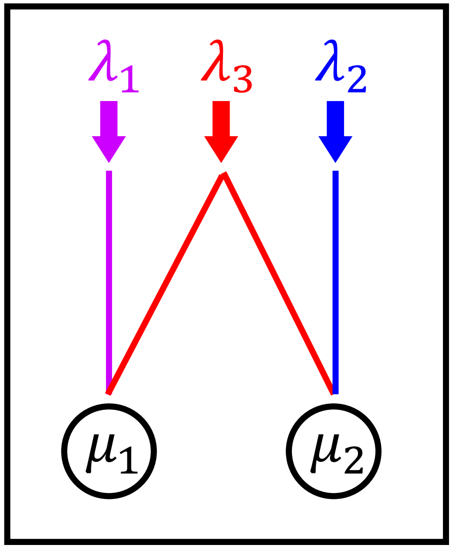


experiences response time of a **class 1** job

$W(\lambda_1, \mu_1)$

experiences response time of a **class 2** job

$W(\lambda_2, \mu_2)$



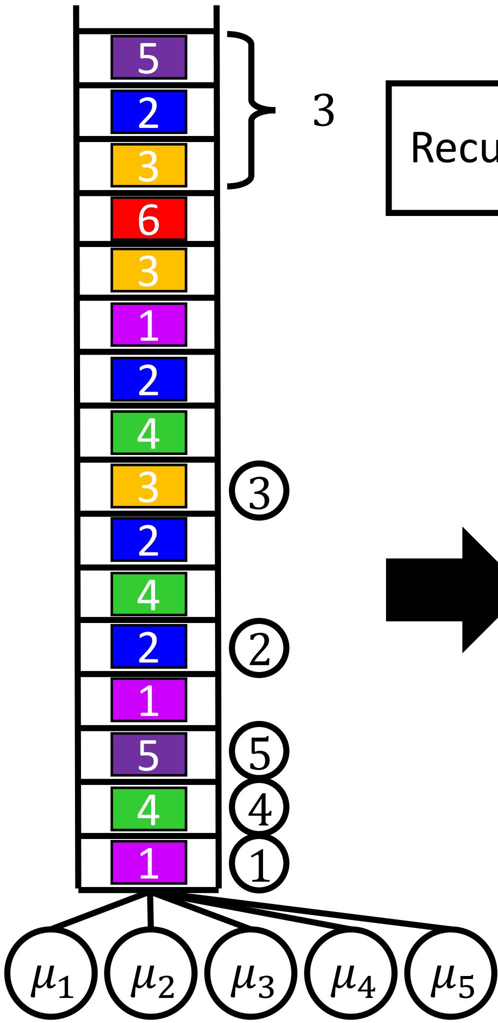
$$W_3 \sim W^Q(\lambda_3, \mu - \lambda_1 - \lambda_2) + \min\{W(\lambda_1, \mu_1), W(\lambda_2, \mu_2)\}$$

$$\sim W^Q(\lambda_3, \mu - \lambda_1 - \lambda_2) + \min\{\mathrm{Exp}(\mu_1 - \lambda_1), \mathrm{Exp}\{\mu_2 - \lambda_2\}\}$$

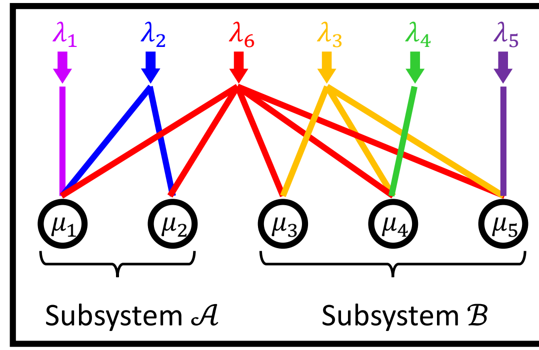
$$\sim W^Q(\lambda_3, \mu - \lambda_1 - \lambda_2) + \mathrm{Exp}(\mu - \lambda_1 - \lambda_2)$$

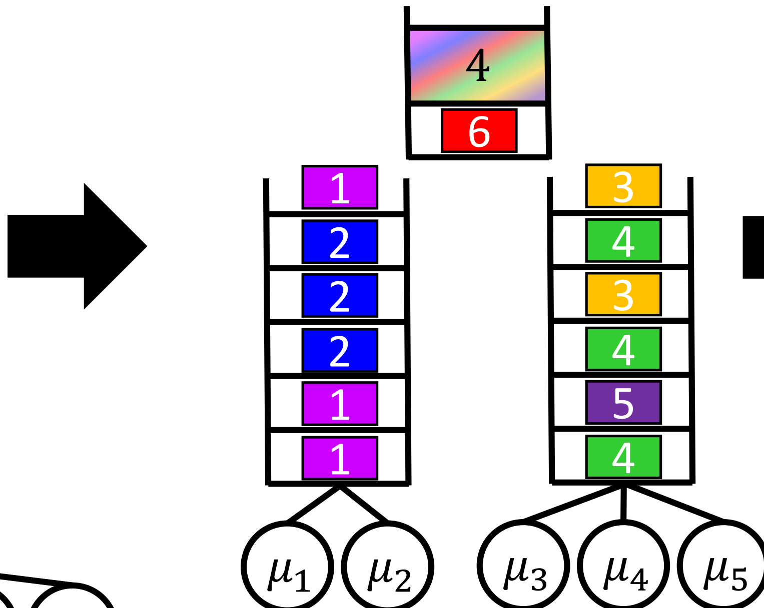
$$\sim W(\lambda_3, \mu - \lambda_1 - \lambda_2)$$

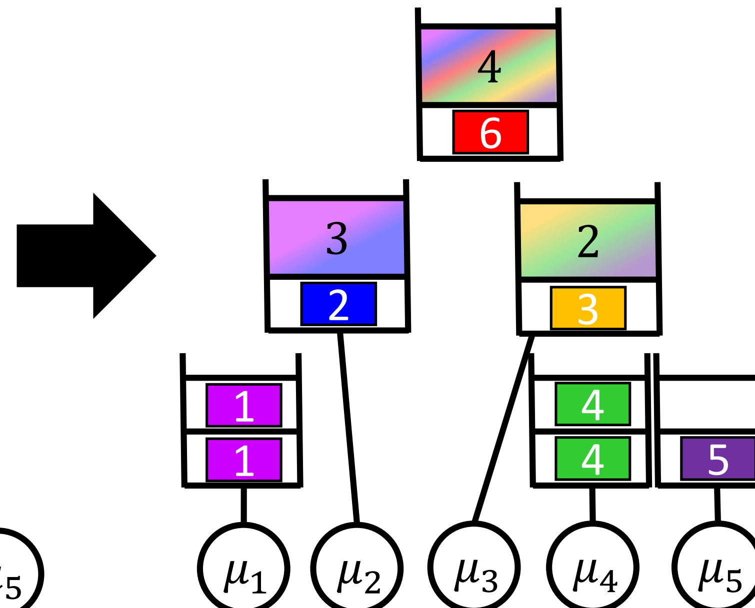
Nested Systems: Collaborative Model



Recursively defined state: $\left(n_J, J, \begin{matrix}(\mathcal{A})\\(\mathcal{B})\end{matrix}\right)$







$$\left(4, 6, \begin{matrix}\left(3, 2, \begin{matrix}1,1\\0\end{matrix}\right)\\ \left(2, 3, \begin{matrix}1,4\\0,5\end{matrix}\right)\end{matrix}\right)$$

$$\vec{c}_{16} = (1,4,5,1,2,4,2,3,4,2,1,3,6,3,2,5)$$

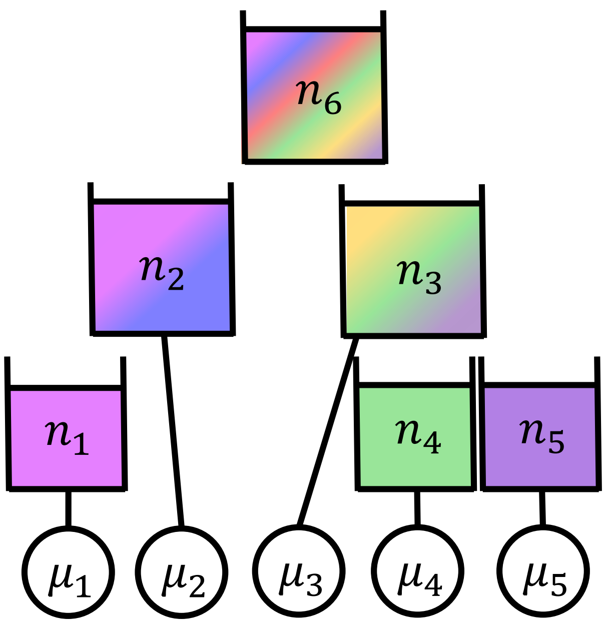
Nested Systems: Collaborative Model

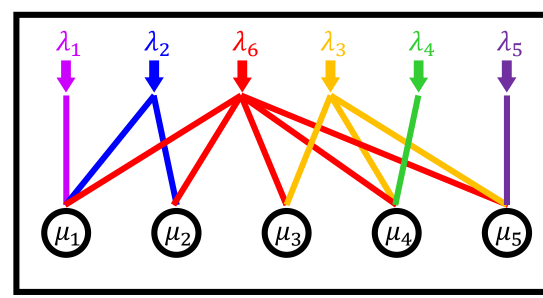
Thm:

$$\pi\left(n_J, J, \begin{matrix}(\mathcal{A})\\(\mathcal{B})\end{matrix}\right) = (1-\rho_J)\left(\frac{\lambda\left(R(S_J)\right)}{\mu(S_J)}\right)^{n_J}\left(\frac{\lambda_J}{\mu(S_J)}\right)\pi(\mathcal{A})\pi(\mathcal{B})$$

where $\rho_i = \frac{\lambda_i}{\mu(S_i) - \lambda\left(R(S_i)\right) + \lambda_i}$

[Gardner, Harchol-Balter, Hyytiä, Righter]



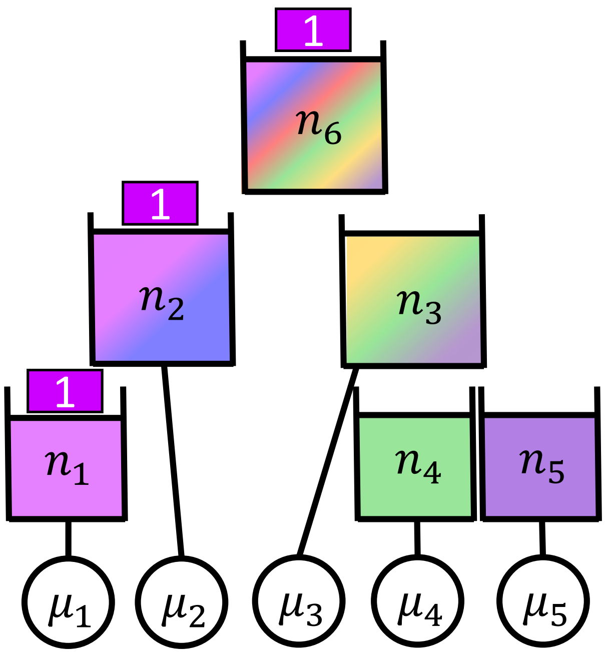


Nested Systems: Collaborative Model

Deriving Performance Metrics

Corollary:

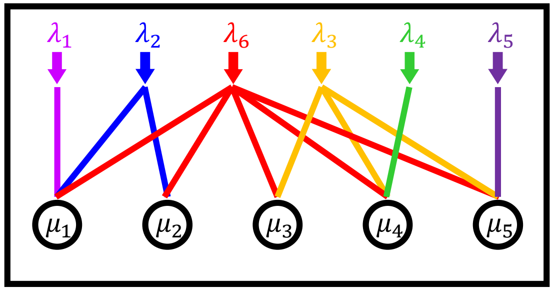
$$W_i =_{st} W\left(\lambda\left(R(S_i)\right), \mu(S_i)\right) + \sum_{j:S_i \subset S_j} W^Q\left(\lambda_j, \mu(S_j) - \lambda\left(R(S_j)\right) + \lambda_j\right)$$



ex. $W_1 =_{st} \quad W^Q(\lambda, \mu - \lambda + \lambda_6)$

$+W^Q(\lambda_2, \mu_{1,2} - \lambda_{1,2} + \lambda_2)$

$+W(\lambda_1, \mu_1)$



Outline

- System description
- Review: reversibility
- Detailed states

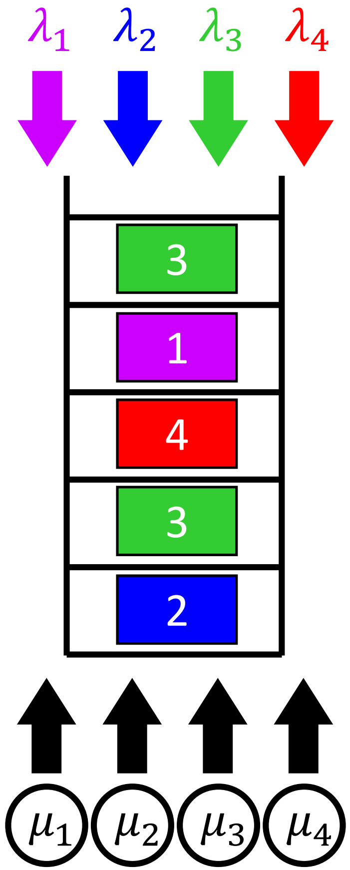
  Collaborative model | Noncollaborative model

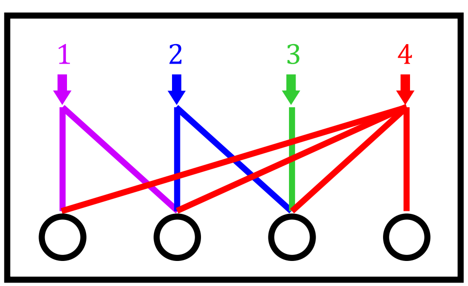
  Relating the models
- Partial aggregation

  Collaborative model | Noncollaborative model

  - Special case: fully flexible class
  - Special case: nested systems
- **Related product-form systems**

Matching Models





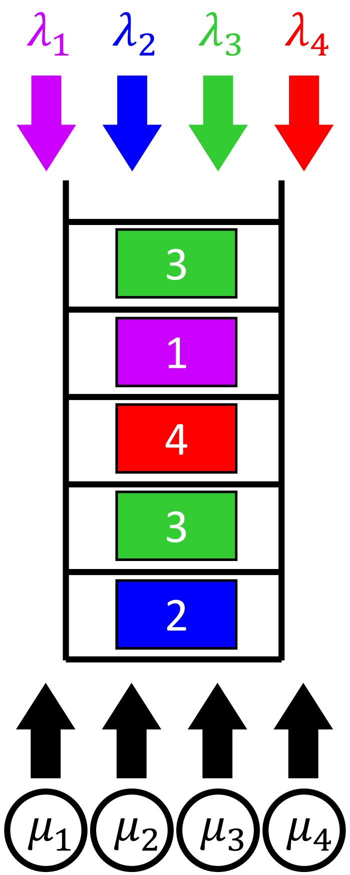
Model 1: arriving jobs wait in the queue for a compatible server, arriving servers match the first compatible job and leave the system (even if unmatched)

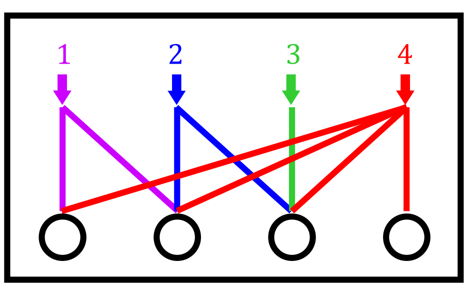
Sample path equivalences for $\vec{c}_n$:

Noncollaborative model given busy (**queue** state) ⟷ Model 1 ⟷ Collaborative model (**system** state)

[Adan, Kleiner, Righter, Weiss]

Matching Models





Other matching models with product form

Model 2 (FCFS infinite bipartite matching): arrivals are job-server pairs, both of which can queue; arriving jobs (servers) match the first compatible server (job) if any, otherwise join queue

[Adan, Busic, Gupta, Mairesse, Weiss]

Thm:

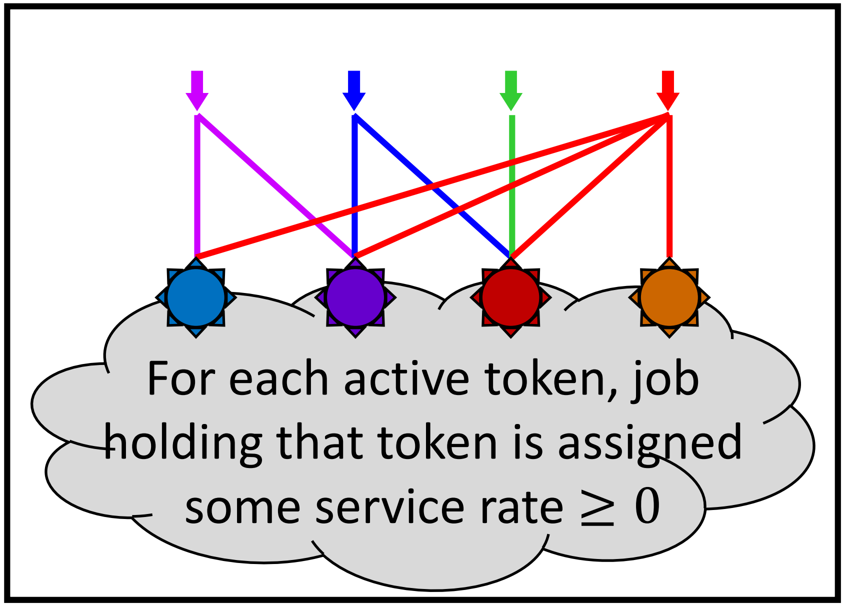
$$\pi(\vec{c}_n;\vec{s}_n) = \pi(0;0)\prod_{i=1}^{n}\frac{\lambda_{c_i}}{\mu(\vec{c}_i)}\frac{\mu_{s_i}}{\lambda(\vec{s}_i)}$$

Model 3: individual arrivals, non-bipartite matching graph, arrivals match the first compatible item in queue if any, otherwise join queue

[Busic, Mairesse, Moyal, Perry]

Unifying Token Model

- Set of tokens, bipartite matching between jobs and tokens
- Arriving job takes compatible token if one is available (assignment condition)
- Only jobs with tokens can be served
- OI condition on tokens defines service process



state $(\vec{c}_n; \vec{a}_m)$

active tokens in the order in which they became active

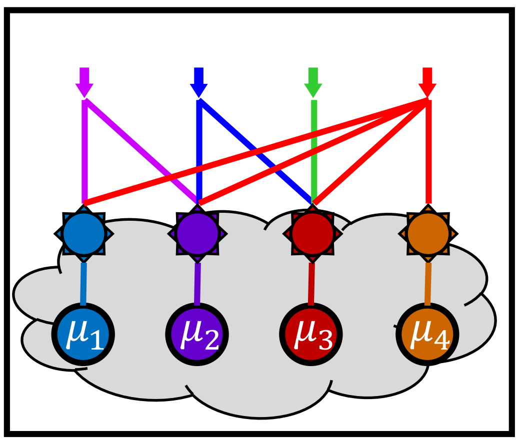
Thm: $$\pi(\vec{c}_n; \vec{a}_m) = B_3 \prod_{i=1}^{n} \frac{\lambda_{c_i}}{\mu(\vec{c}_i)} \prod_{l=1}^{m} \frac{\lambda^A_{a_l}(\vec{a}_{l-1})}{\mu(\vec{a}_l)}$$

[Ayesta, Bodas, Dorsman, Verloop]

Unifying Token Model

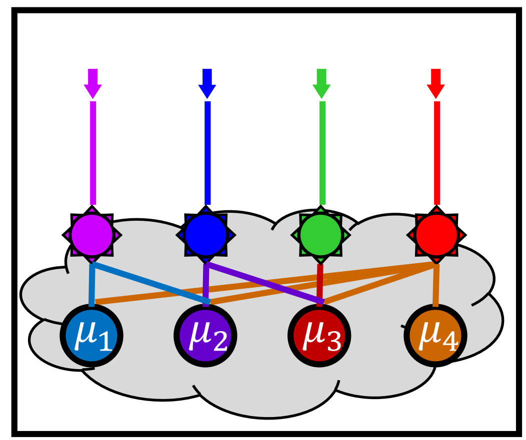
- Set of tokens, bipartite matching between jobs and tokens
- Arriving job takes compatible token if one is available (assignment condition)
- Only jobs with tokens can be served
- OI condition on tokens defines service process

Noncollaborative Model



tokens ↔ servers

Collaborative Model



tokens ↔ job classes

…but much more general than these two models

e.g., MSCCC queue [Ayesta, Bodas, Dorsman, Verloop]

Summary and Conclusions: Part 1

Detailed States

Collaborative (C) model

$\vec{c}_n$

classes of all jobs in the **system**

Noncollaborative (NC) model

ALIS: $(\vec{c}_n; \vec{s}_k)$

RAND: $(\vec{c}_n; b_m)$

classes of all jobs in the **queue**

servers in order they became **idle**

servers in order they became **busy**

Conditioned on all servers being busy in NC, the noncollaborative **queue** has the same stationary distribution as the collaborative **system**

Partially Aggregated States

Collaborative (C) model

$(\vec{d}_l, \vec{n}_l)$

classes of all jobs in **service**

number of jobs in queue between jobs in service

Noncollaborative (NC) model

ALIS: $(\vec{n}_l; \vec{b}_l; \vec{s}_{M-l})$

RAND: $(\vec{n}_l; \vec{b}_l)$

Fully flexible class experiences M/M/1 response time

Easy to derive per-class response time distributions in nested systems (C)