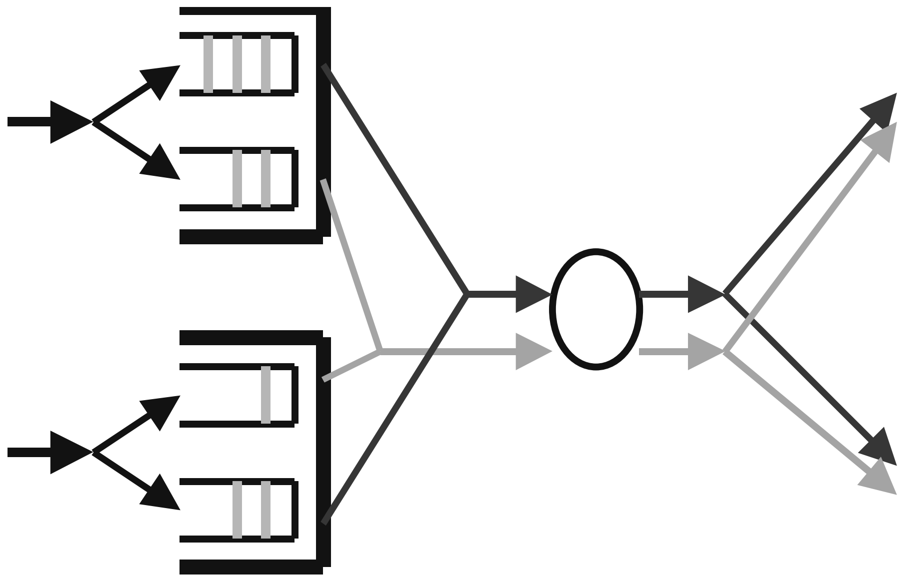
Stochastic Processing Networks



Ruth J. Williams

University of California, San Diego

http://www.math.ucsd.edu/~williams

Maurice Belz (1897-1975)

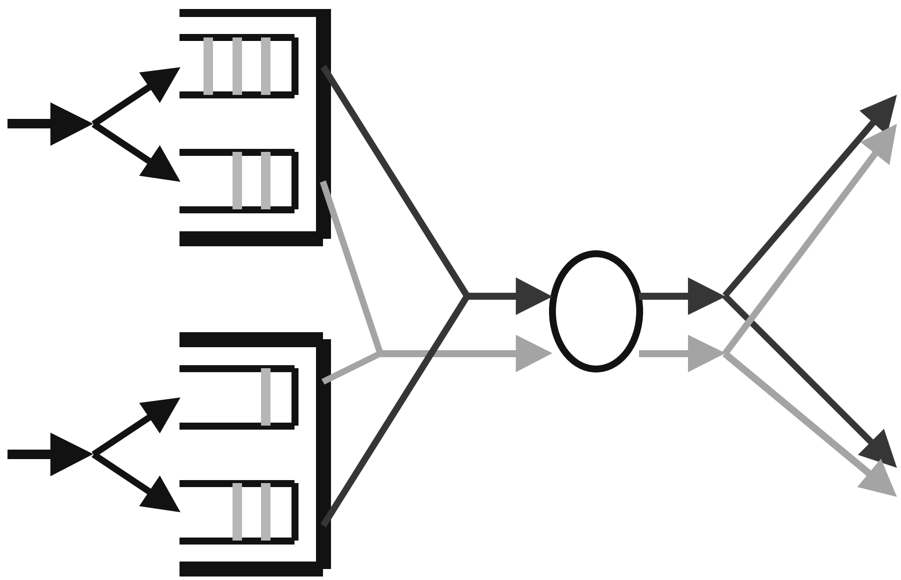
Founding Professor of Statistics, University of Melbourne,1955-1963

Maurice Belz (1897-1975)

Founding Professor of Statistics, University of Melbourne (1955-1963)

Statistical Methods for the Process Industries (1973)

Stochastic Processing Networks: What, Why and How?



Ruth J. Williams

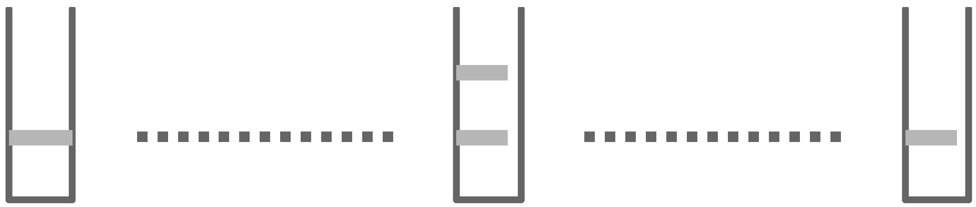
University of California, San Diego

http://www.math.ucsd.edu/~williams

OUTLINE

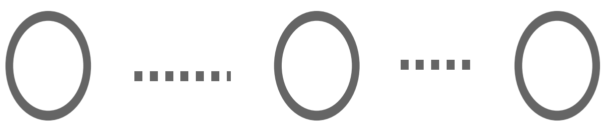
- What is a Stochastic Processing Network?
- Applications
- Questions
- A Simple Example
- Approximations
- Perspective
- Two Motivating Examples
- Main Topics for Remaining Lectures

Stochastic Processing Networks (cf. Harrison ‘00)



$\mathbb{I}$ **buffers (classes)**

$\mathbb{J}$ **activities**

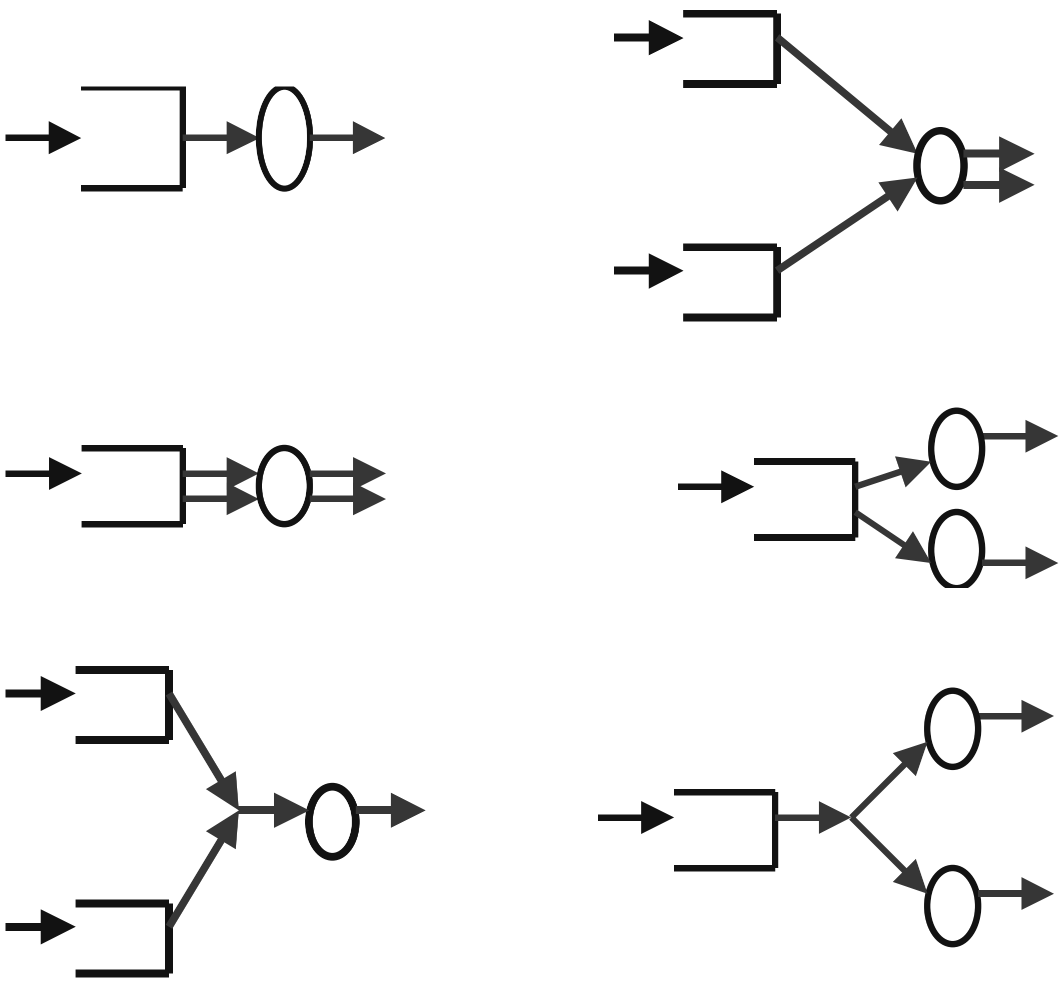


$\mathbb{K}$ **servers (resources)**

An activity consumes from certain classes, produces for certain (possibly different) classes, and uses certain servers.

Stochastic Processing Networks

SPN Activities are Very General

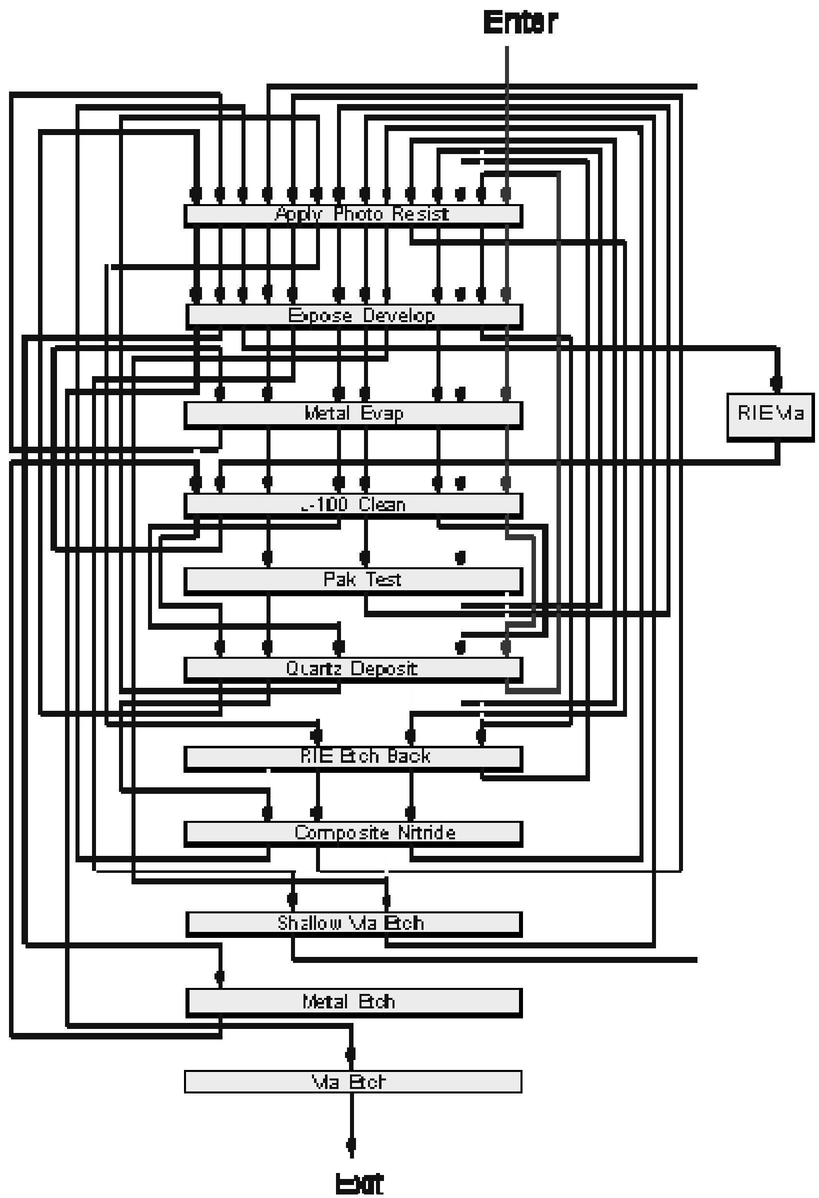


Queueing network

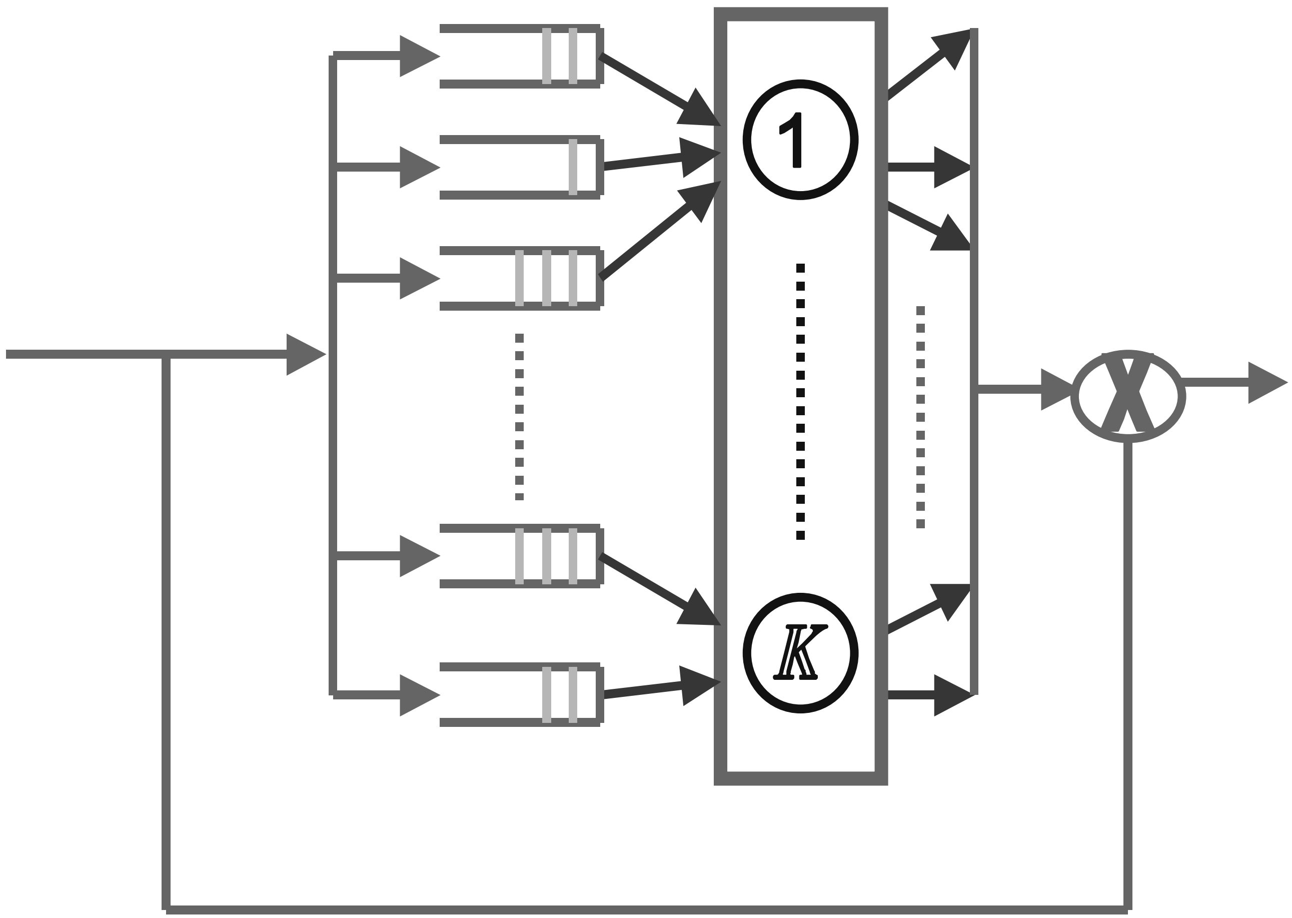
Flexible servers, alternate routing

Simultaneous actions

Semiconductor Wafer Fab: P. R. Kumar



Multiclass Queueing Network

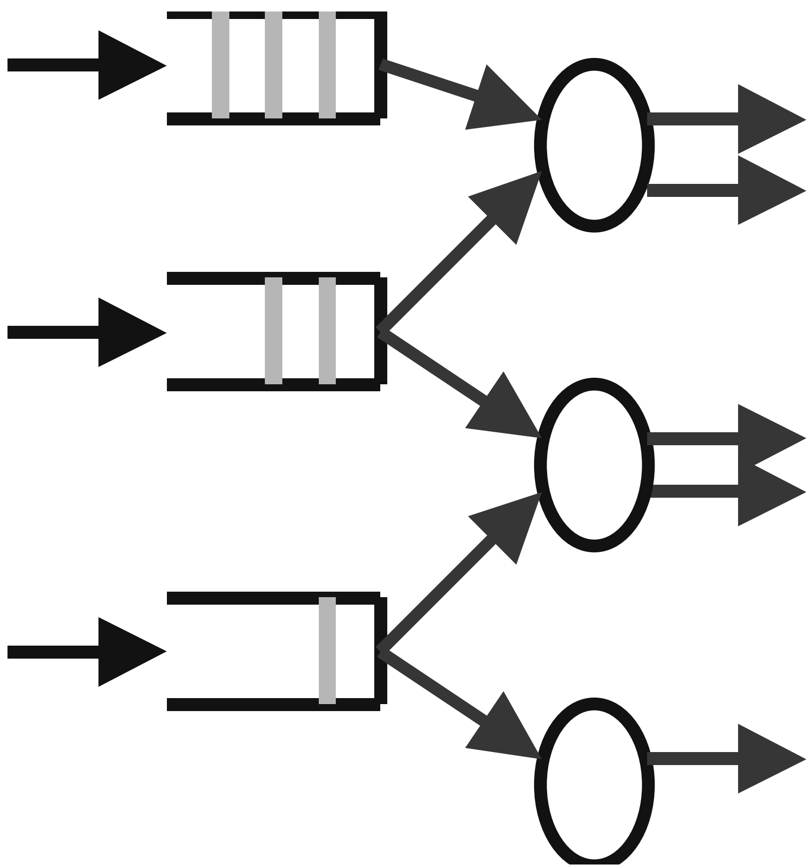


Call Center: First Direct (branchless retail banking)

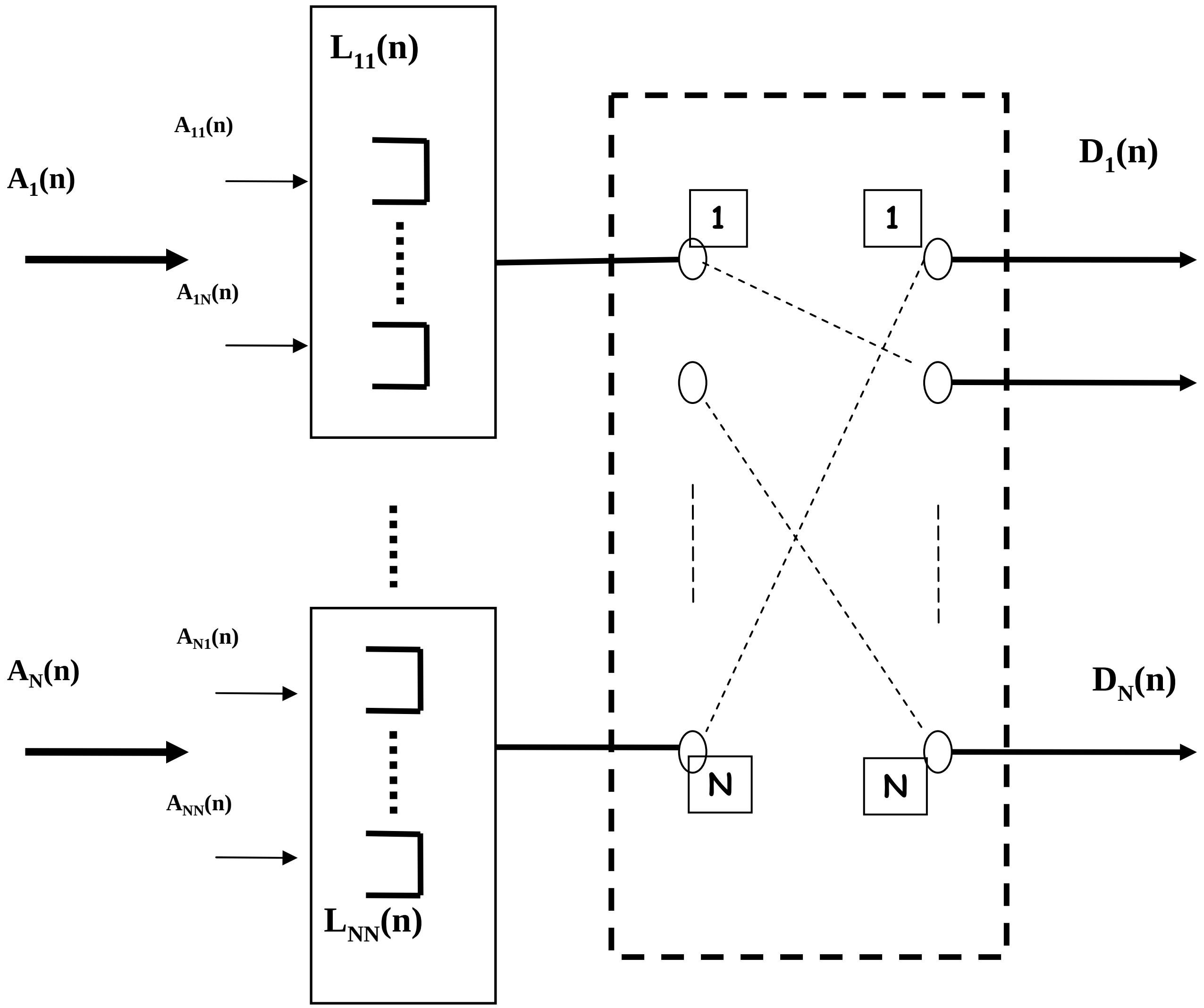
Larreche et al., INSEAD ‘97 (see also Gans, Koole, Mandelbaum ‘93)

Differentiated Service Center

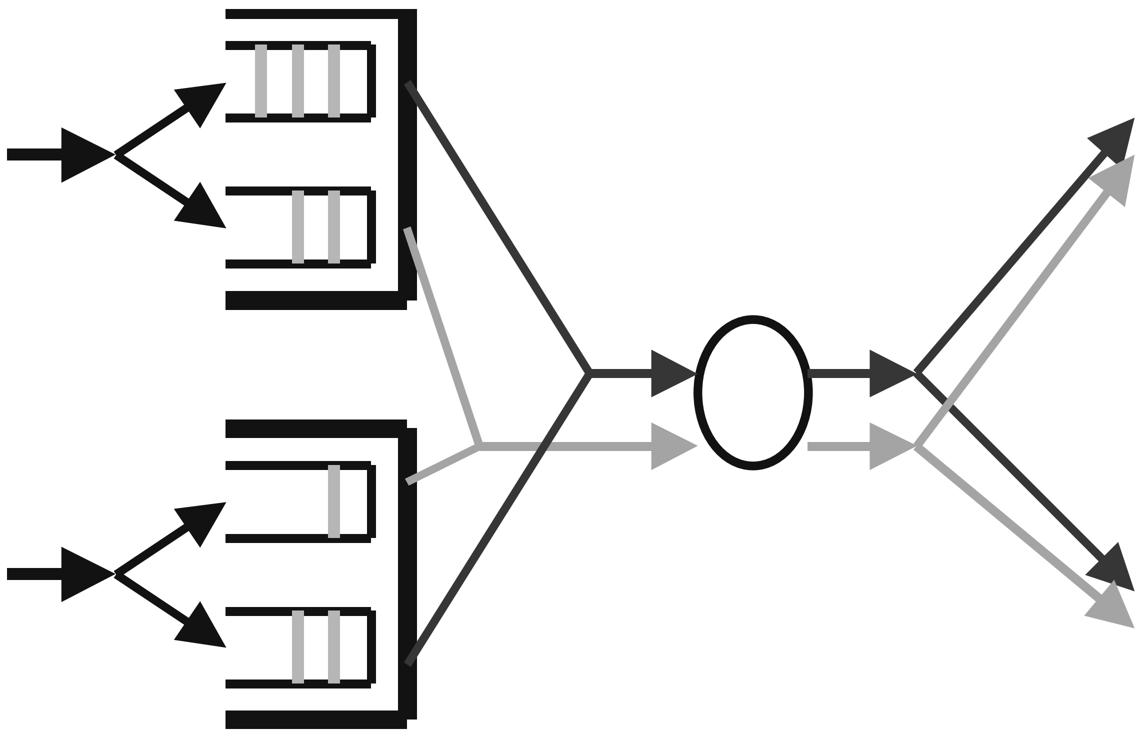
(Parallel server system, alternate routing)



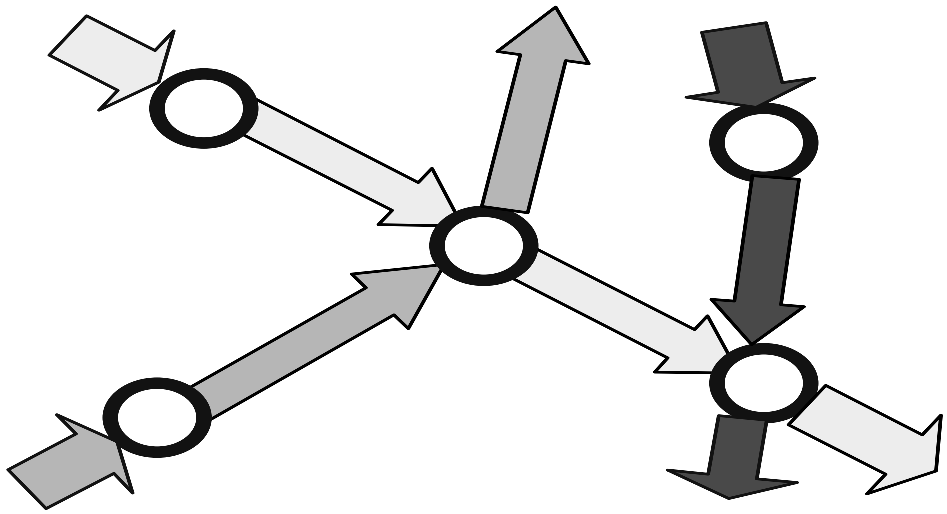
NxN Input Queued Packet Switch: Prabhakar



2x2 Input Queued Packet Switch



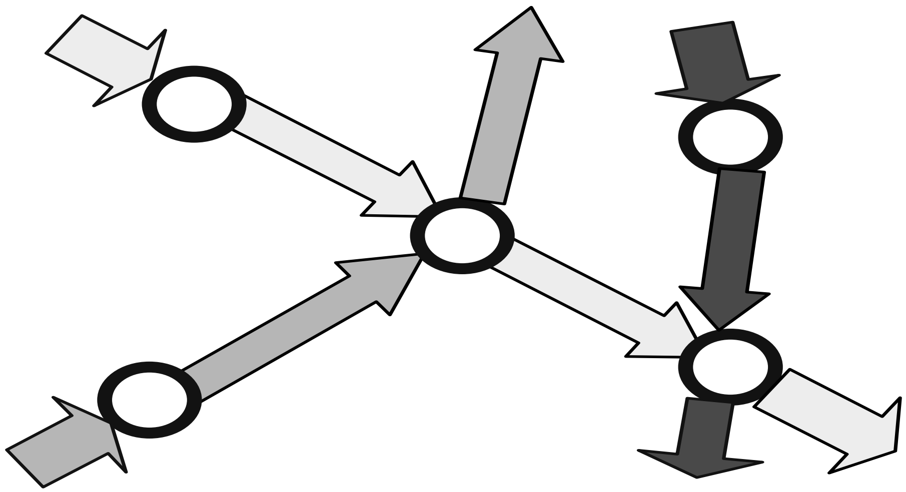
Data Network (Roberts and Massoulie, ‘00)

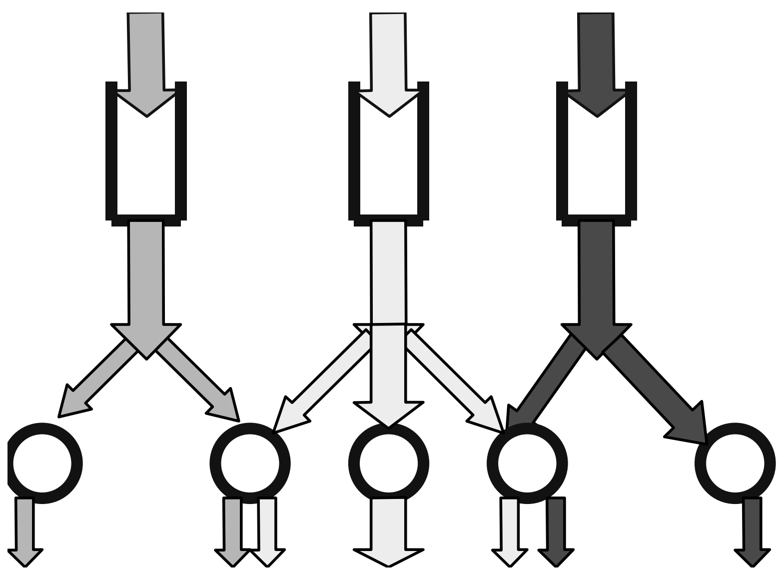


Link

Route

Simultaneous Resource Possession





Stochastic Processing Networks

- APPLICATIONS

Complex manufacturing, telecommunications, computer systems, service networks

- FEATURES

Multiclass, service discipline, alternate routing, complex feedback, heavily loaded

- PERFORMANCE MEASURES

Queuelength, workload and server idletime

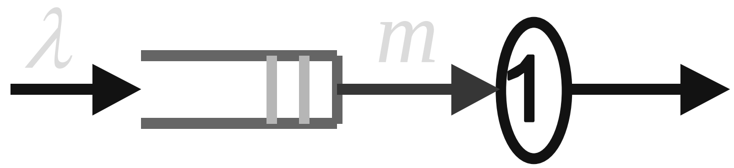
QUESTIONS

- STABILITY
- PERFORMANCE ANALYSIS (when heavily loaded)
- CONTROL (involves performance analysis for "good" controls)

A SIMPLE EXAMPLE:

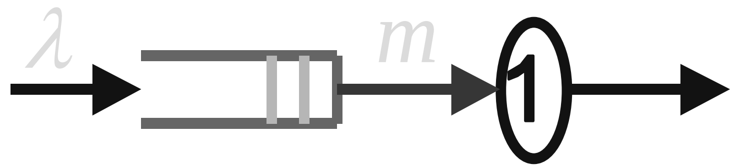
SINGLE SERVER QUEUE

M/M/1 Queue



- Poisson arrivals at rate λ (independent of service times)
- i.i.d. exponential service times mean m
- FIFO order of service, infinite buffer

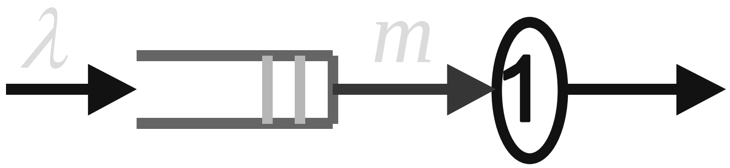
M/M/1 Queue



- Poisson arrivals at rate λ (independent of service times)
- i.i.d. exponential service times mean m
- FIFO order of service, infinite buffer

- Traffic intensity $\rho = \lambda m$

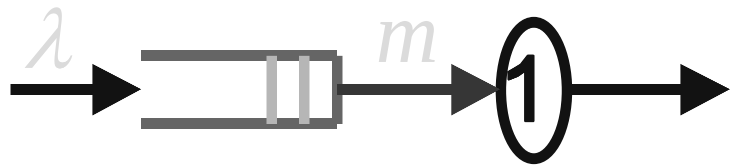
M/M/1 Queue



- Poisson arrivals at rate λ (independent of service times)
- i.i.d. exponential service times mean m
- FIFO order of service, infinite buffer

- Traffic intensity $\rho = \lambda m$
- Queuelength is a birth-death process (Markov)

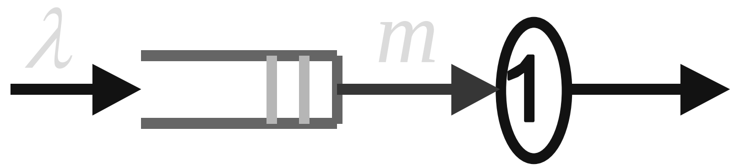
M/M/1 Queue



- Poisson arrivals at rate λ (independent of service times)
- i.i.d. exponential service times mean m
- FIFO order of service, infinite buffer

- Traffic intensity $\rho = \lambda m$
- Queuelength is a birth-death process (Markov)
- Positive recurrent (stable) iff $\rho < 1$

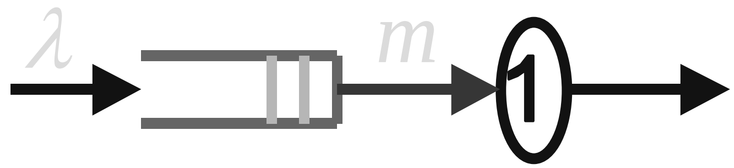
M/M/1 Queue



- Poisson arrivals at rate λ (independent of service times)
- i.i.d. exponential service times mean m
- FIFO order of service, infinite buffer

- Traffic intensity $\rho = \lambda m$
- Queuelength is a birth-death process (Markov)
- Positive recurrent (stable) iff $\rho < 1$
- Stationary distribution $\pi_i = \rho^i (1-\rho), \quad i = 0,1,2,\ldots$
- Mean steady-state queuelength $L = \rho/(1-\rho)$

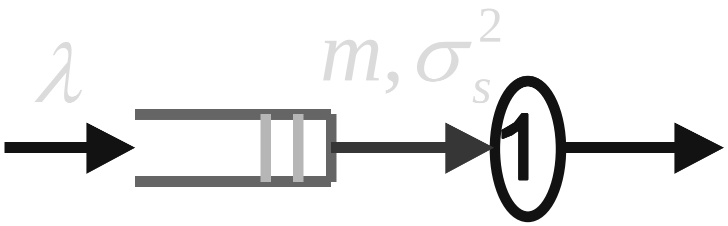
M/M/1 Queue



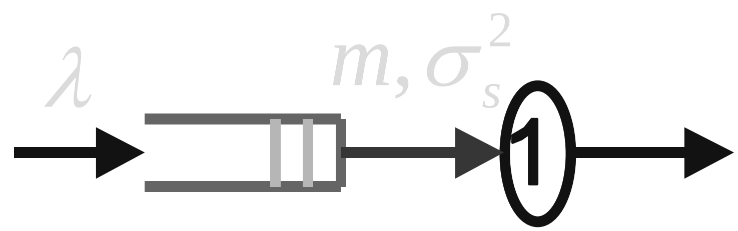
- Poisson arrivals at rate λ (independent of service times)
- i.i.d. exponential service times mean m
- FIFO order of service, infinite buffer

- Traffic intensity $\rho = \lambda m$
- Queuelength is a birth-death process (Markov)
- Positive recurrent (stable) iff $\rho < 1$
- Stationary distribution $\pi_i = \rho^i (1-\rho), \quad i = 0,1,2,\ldots$
- Mean steady-state queuelength $L = \rho/(1-\rho) = \lambda W$

M/GI/1 Queue



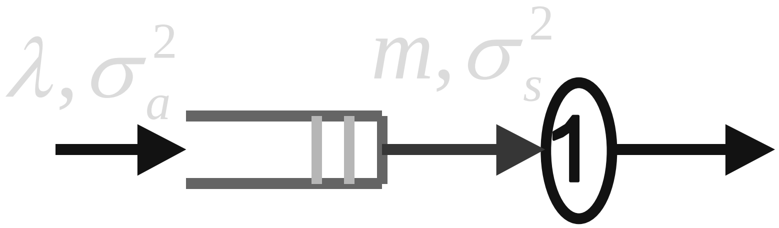
M/GI/1 Queue



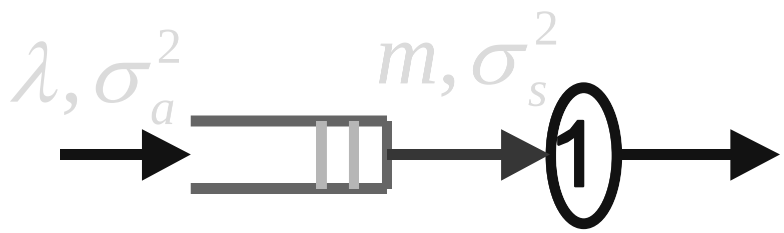
- Mean steady-state queuelength

$$L = \rho + \frac{\rho^2 + \lambda^2 \sigma_s^2}{2(1-\rho)}$$ (Pollaczek-Khintchine)

GI/GI/1 Queue (+mild reg. assumptions)



GI/GI/1 Queue (+mild reg. assumptions)

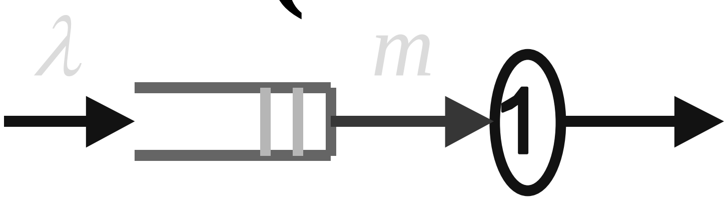


$$(1-\rho)L \approx \frac{\lambda^2(\sigma_a^2+\sigma_s^2)}{2} \qquad \text{for } \rho \simeq 1$$

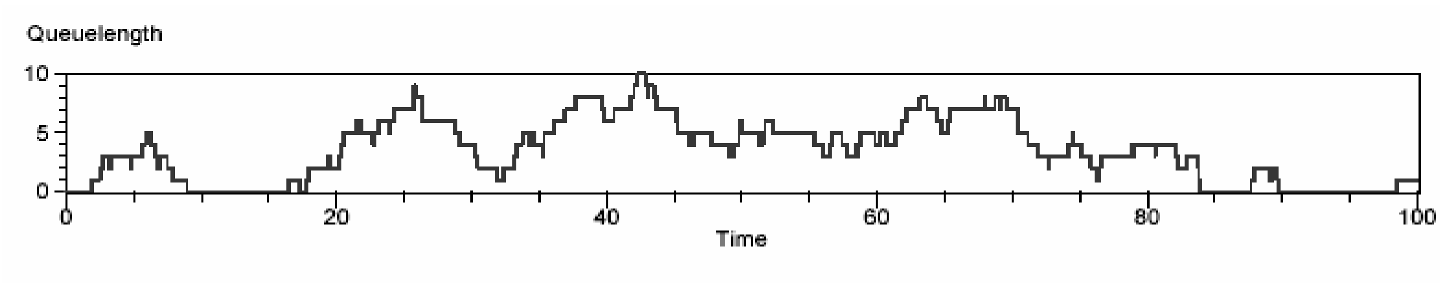
(Smith ‘53, Kingman ‘61)

M/M/1 Queue

(Simulation of Dynamics)

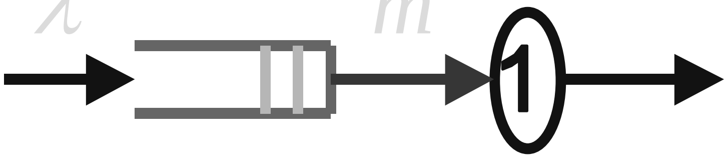


$$\rho = \lambda = 0.9524$$

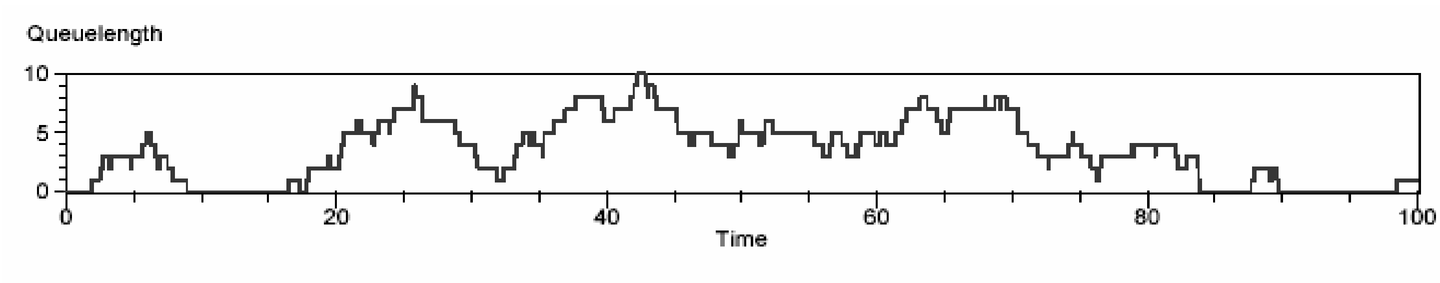


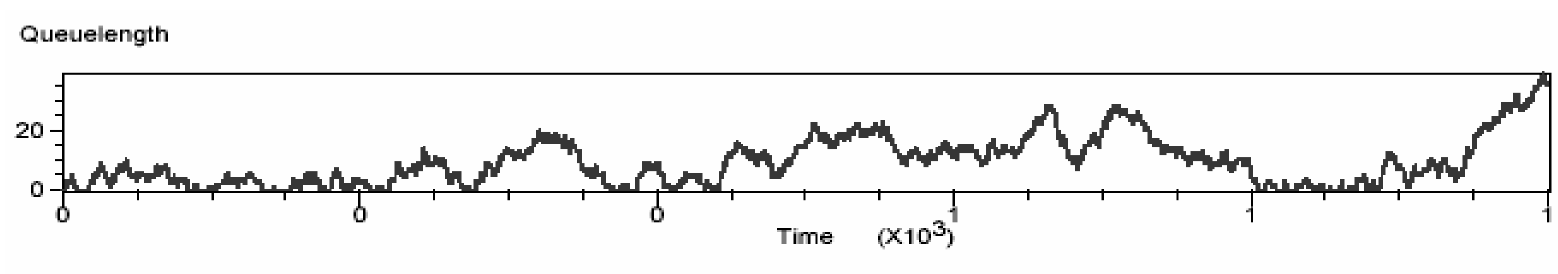
M/M/1 Queue

(Simulation of Dynamics)



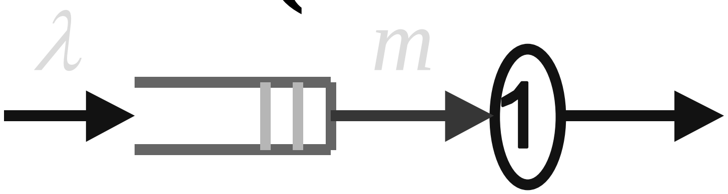
$$\rho = \lambda = 0.9524$$



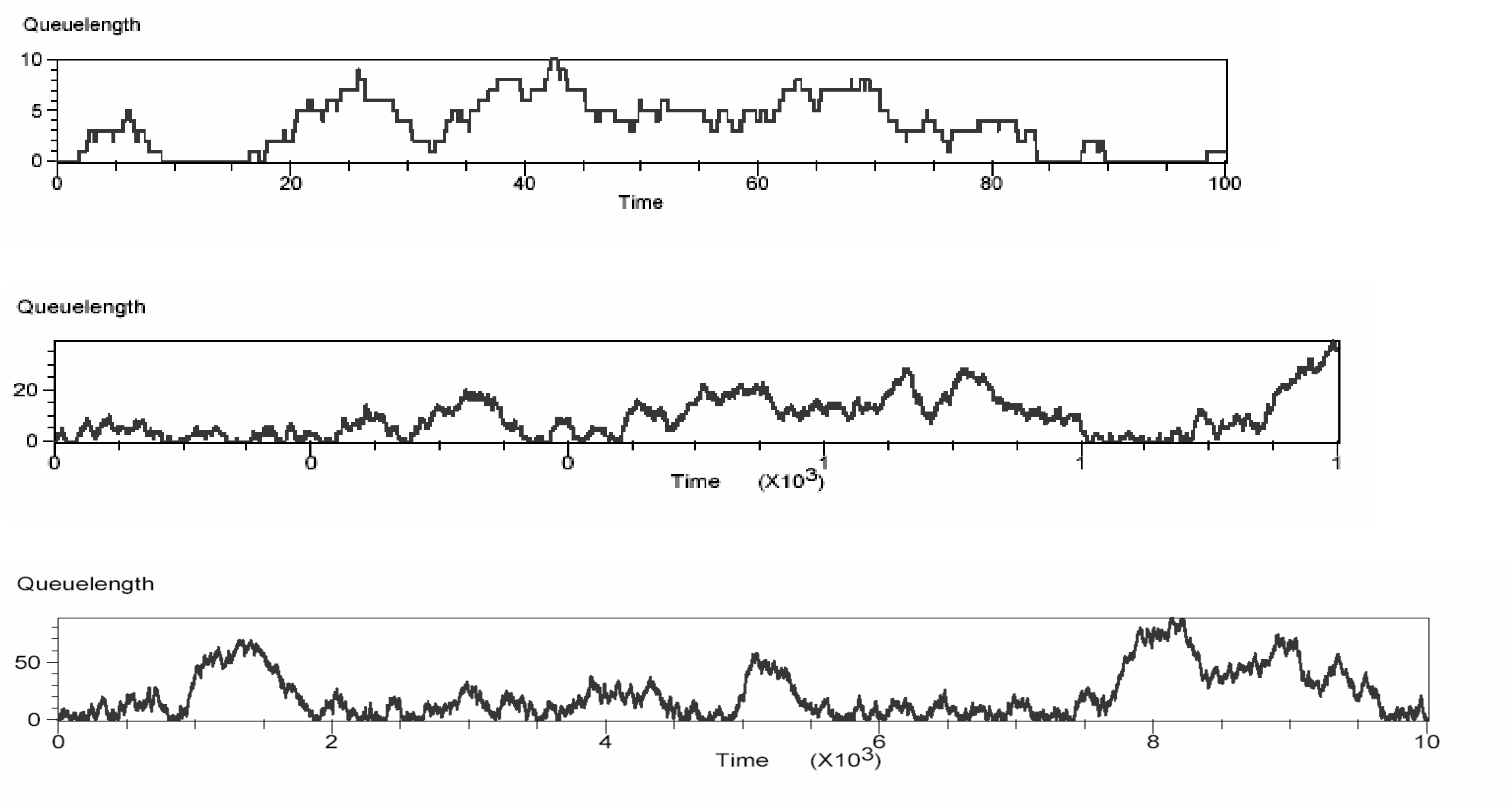


M/M/1 Queue

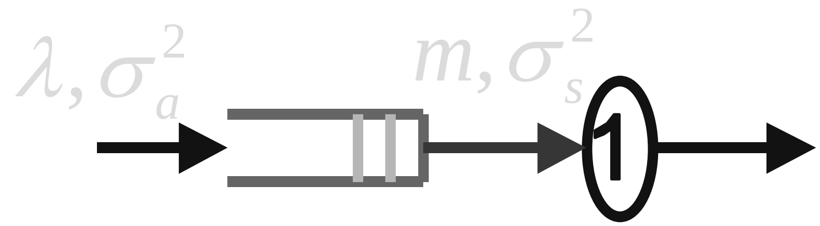
(Simulation of Dynamics)



$\rho = \lambda = 0.9524$



GI/GI/1 Queue (Dynamics)



Q(t) = queuelength at time *t*

Start system empty (for simplicity)

Theorem (A. Borovkov '67, Iglehart-Whitt '70): For $\rho \approx 1$,

$(1-\rho)Q(\cdot/(1-\rho)^2) \approx Q^*(\cdot)$ where $Q^*(\cdot)$

is a one-dimensional reflecting Brownian motion

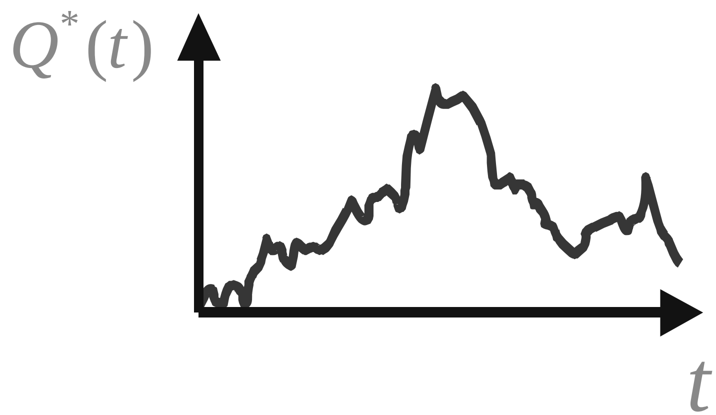
with drift $-m^{-1}$ and variance parameter $\lambda^3\sigma_a^2 + m^{-3}\sigma_s^2$

One-dimensional Reflecting Brownian Motion

$$Q^*(t) = X^*(t) + Y^*(t)$$

$$Y^*(t) = \sup\{-X^*(s) : 0 \leq s \leq t\}$$

$$X^* = \text{Brownian motion}$$



APPROXIMATE DYNAMIC MODELS

- Most SPNs cannot be analyzed exactly
- Consider approximate models (valid under some scaling limit, e.g., heavily loaded, many sources, many servers, large networks)
- Two main classes of approximate models:
  - Fluid models (functional law of large numbers)
  - Diffusion models (functional central limit theorem)

ANSWERS
(OPEN MULTICLASS HL QUEUEING NETWORKS)

Last 15 years: development of a theory for establishing stability and heavy traffic diffusion approximations for open multiclass queueing networks with non-idling head-of-the-line (HL) service disciplines.

Head-of-the-line: service allocated to a buffer goes to the job at the head-of-the-line (jobs within buffers are in FIFO order).

PERSPECTIVE

	MQN	SPN
HL	Sufficient conditions for stability and diffusion approximations	e.g., parallel server system, packet switch
Non-HL	e.g., LIFO, Processor Sharing (single station, PS: network stability)	e.g., Internet congestion control / bandwidth sharing model

MOTIVATING EXAMPLES

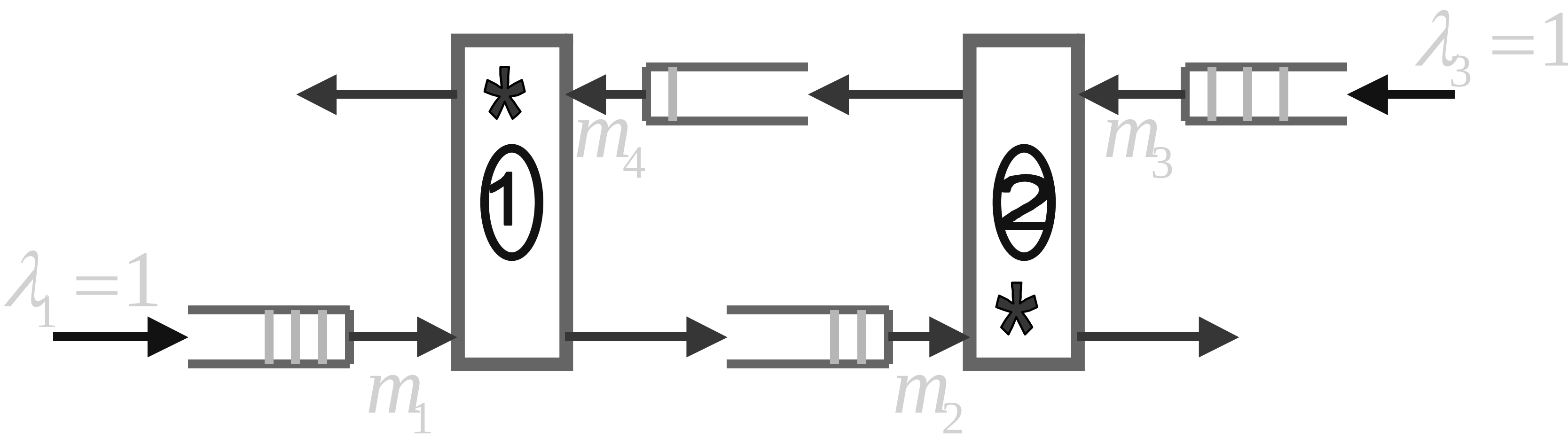
Stability

Performance

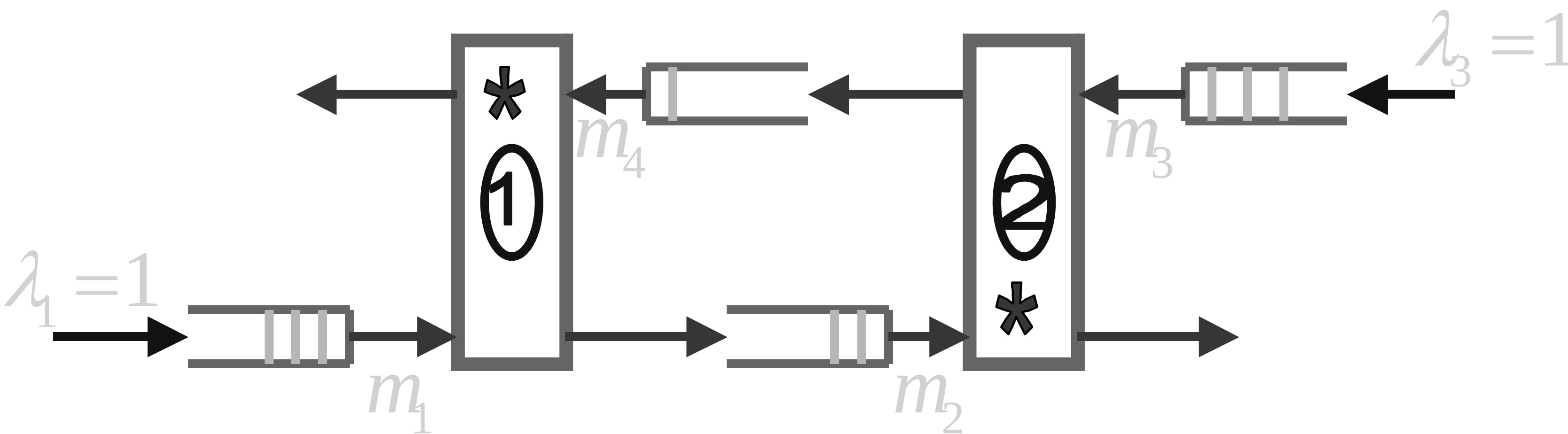
Control

Two-Station Priority Queueing Network

(Rybko-Stolyar ‘92)



Two-Station Priority Queueing Network
(Rybko-Stolyar ‘92)



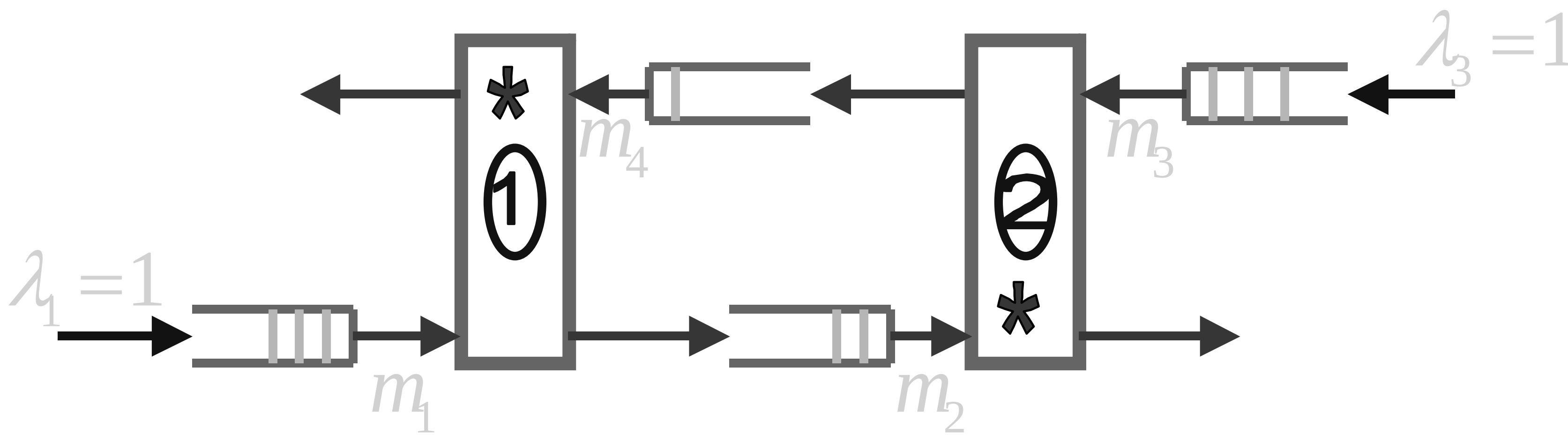
•Poisson arrivals at rate *1* to buffers *1* and *3*

•Exponential service times: m_i mean rate of service for buffer *i*

•Preemptive resume priority: * denotes high priority classes

Two-Station Priority Queueing Network

(Rybko-Stolyar '92)



•Poisson arrivals at rate *1* to buffers *1* and *3*

•Exponential service times: m_i mean rate of service for buffer *i*

•Preemptive resume priority: * denotes high priority classes

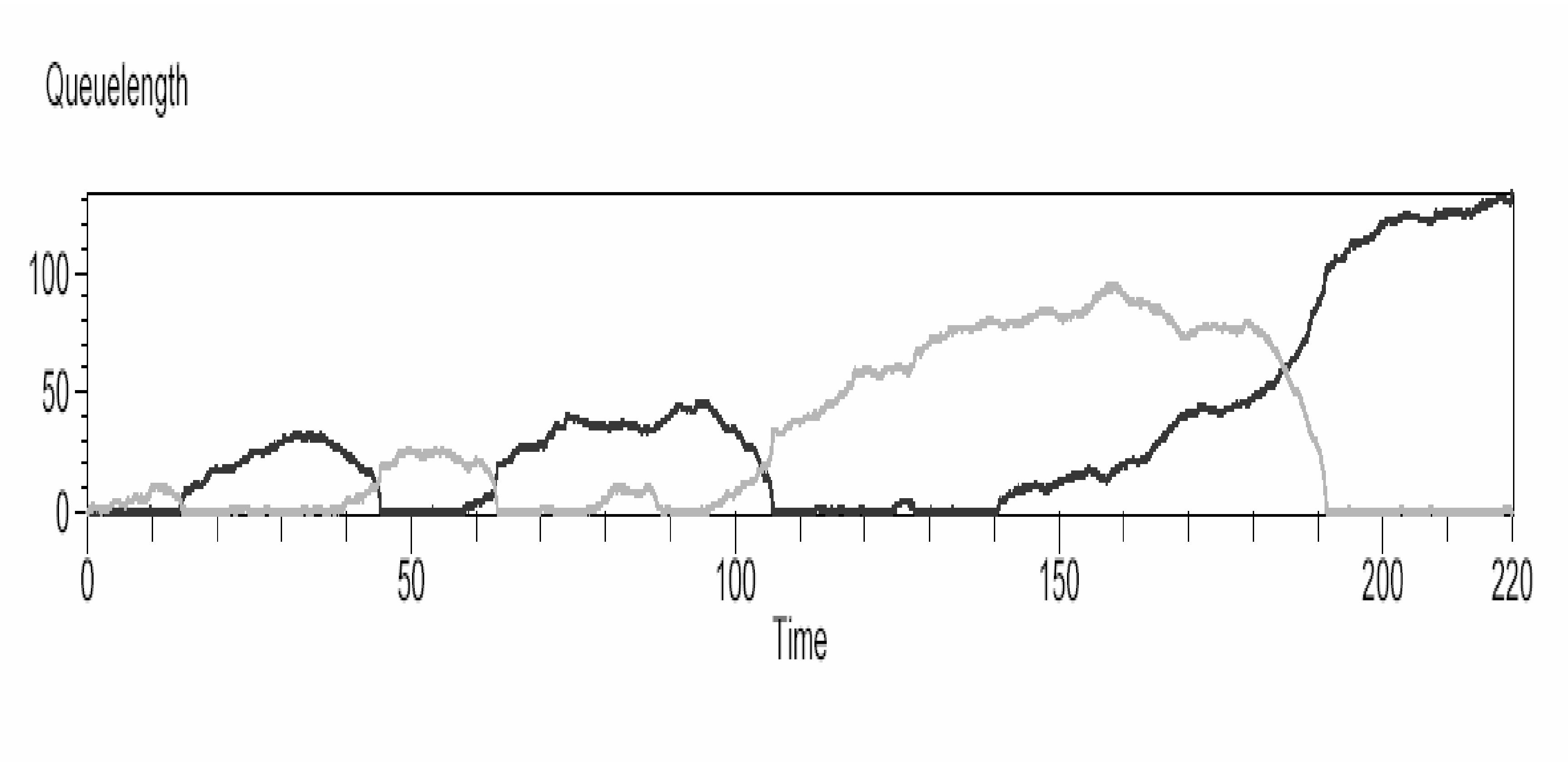
•Simulation: $m_1 = m_3 = 0.33,\ m_2 = m_4 = 0.66$

•Traffic intensities: $\rho_1 = m_1 + m_4 = 0.99 \qquad \rho_2 = m_2 + m_3 = 0.99$

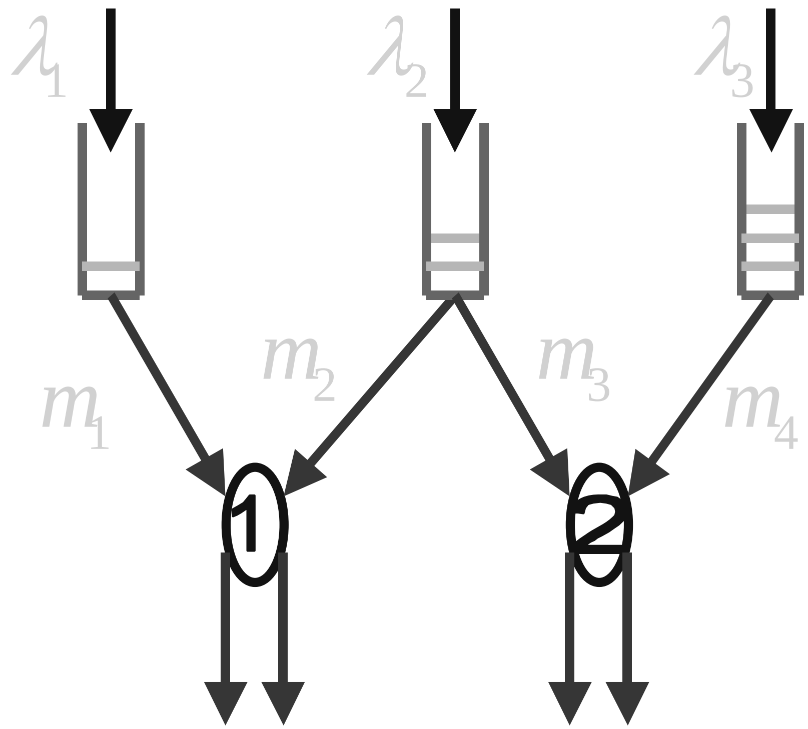
Two-Station Priority Queueing Network

(Rybko-Stolyar '92)

--- Server 1 (sum of queues 1 & 4) --- Server 2 (sum of queues 2 & 3)



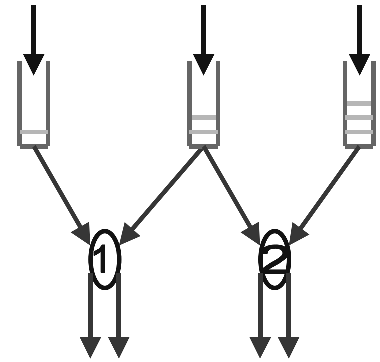
Parallel Server System



$$\lambda_1 = 0.05,\ \lambda_2 = 1.2,\ \lambda_3 = 0.35$$

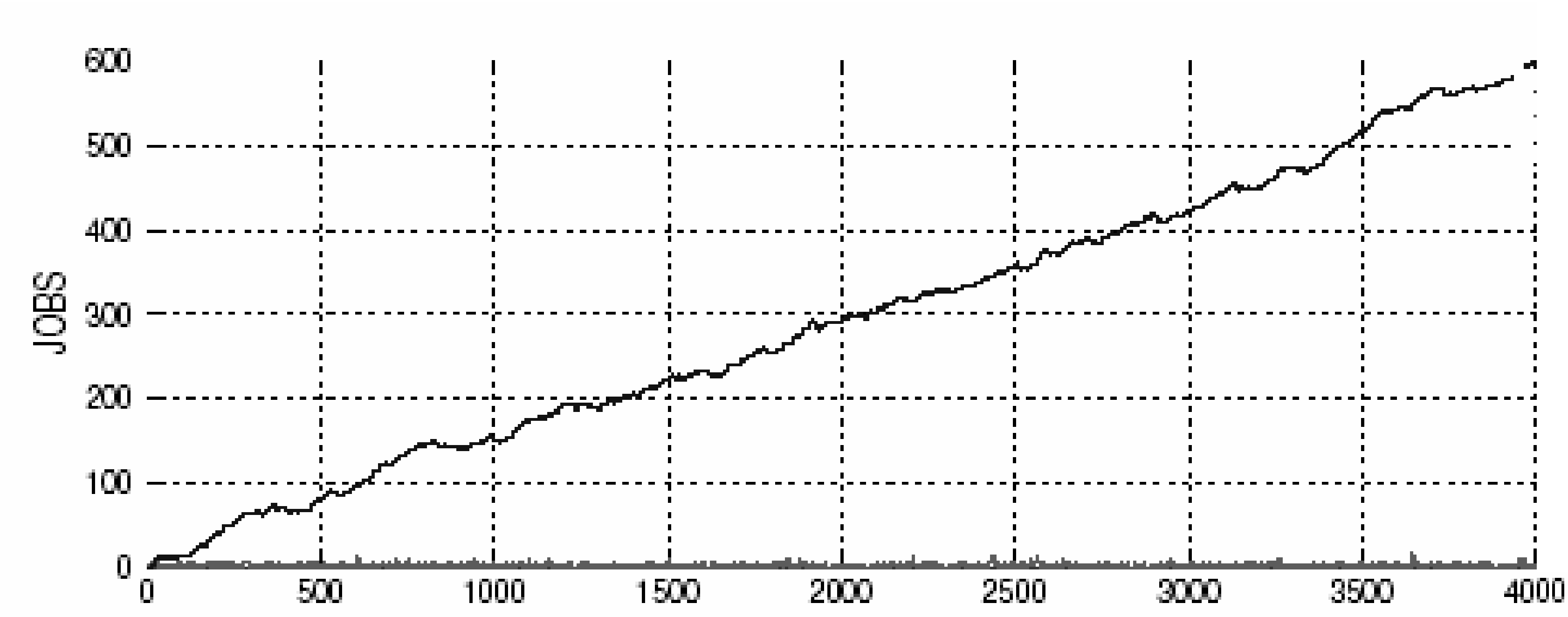
$$m_1 = 0.5,\ m_2 = 1,\ m_3 = 1,\ m_4 = 2$$

Parallel Server System



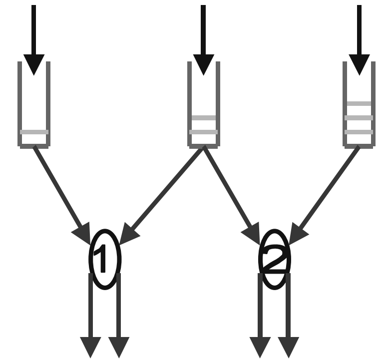
Simulation with static priority discipline:

server 1 gives priority to buffer 1, server 2 gives priority to buffer 2



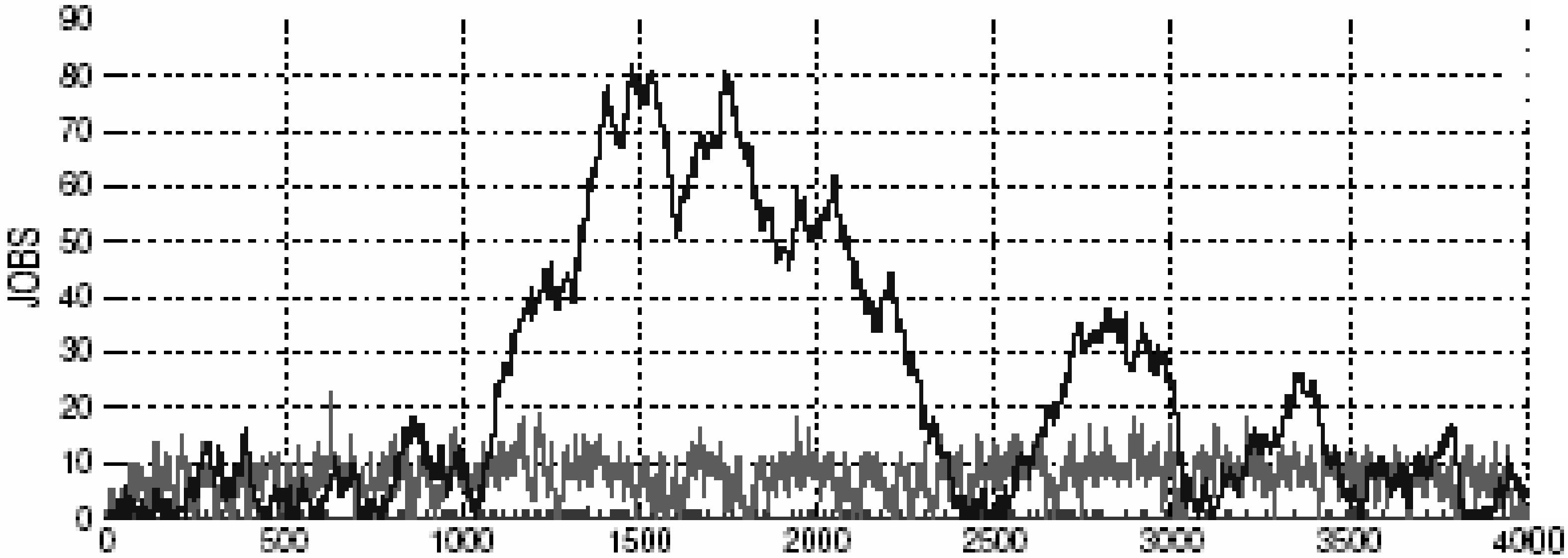
Queuelengths for buffer 1 ---, buffer 2 ---, buffer 3 --- versus time

Parallel Server System



Simulation with dynamic priority discipline:

server 1 gives priority to buffer 1, server 2 gives priority to buffer 2, except when queue 2 goes below threshold of size 10



Queuelengths for buffer 1 ---, buffer 2 ---, buffer 3 --- versus time

MAIN TOPICS FOR REMAINING LECTURES

- Open Multiclass HL Queueing Networks: Stability and Performance
- Control of Stochastic Processing Networks: Some Theory and Examples