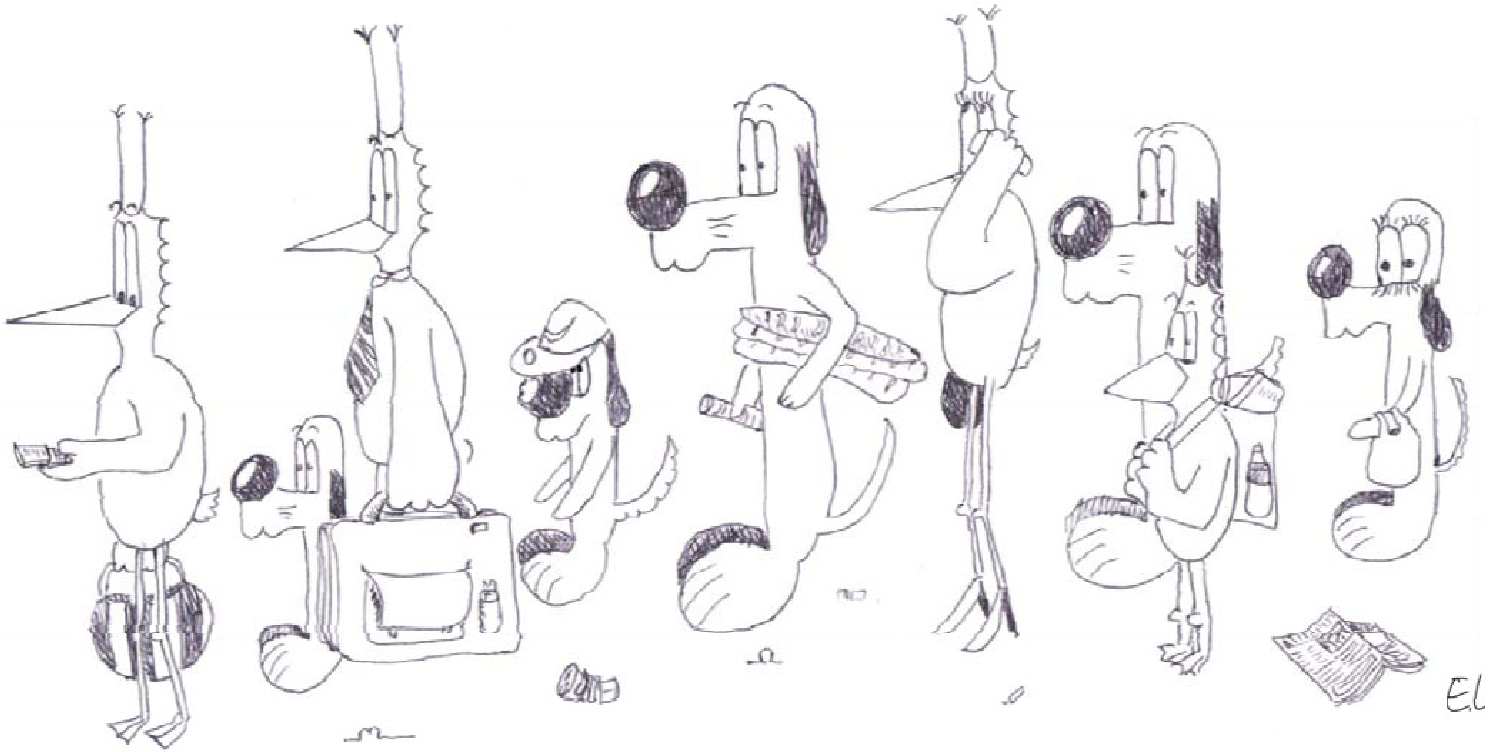
Queuing Networks
Mean Value Analysis



Jean-Yves Le Boudec

Goal of This Module

- Learn on an example how to solve a product-form queuing network

Reminder : A queuing network is called «Product Form» if...

- Markov routing
- One or several classes
- External arrivals, if any are Poisson

Station are either

- FIFO
  - with one or more servers (possibly with exclusion constraints - MSCCC)
  - exponential service times, independent of class
- Or insensitive station:
  - delay, processor sharing, LCFS among others
  - Service time is arbitrary, with finite mean – may depend on class

A product-form queuing network...

- ...is **stable** when the natural stability condition holds
- The stationary distribution of state and of number of customers has product form (Theorem 8.7):

Normalizing constant

$$P(\vec{n}_1, \dots, \vec{n}_S) = K\, p_1(\vec{n}_1) \dots p_S(\vec{n}_S)$$

Station 1

Station S

Depends only on station and visit rates, not on the othernetwork around

- $p_s(\vec{n}_s) = f_s(\vec{n}_s)\theta_{1,s}^{n_{1,s}} \dots \theta_{C,s}^{n_{C,s}}$

Visit rate for class C at station s

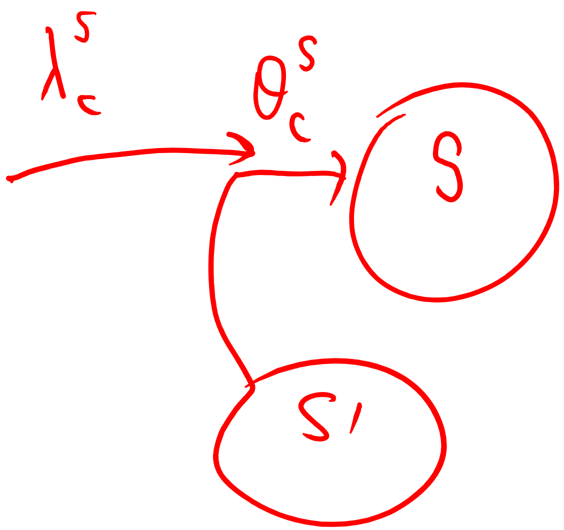
Station function depends only on station in isolation

Visit Rates

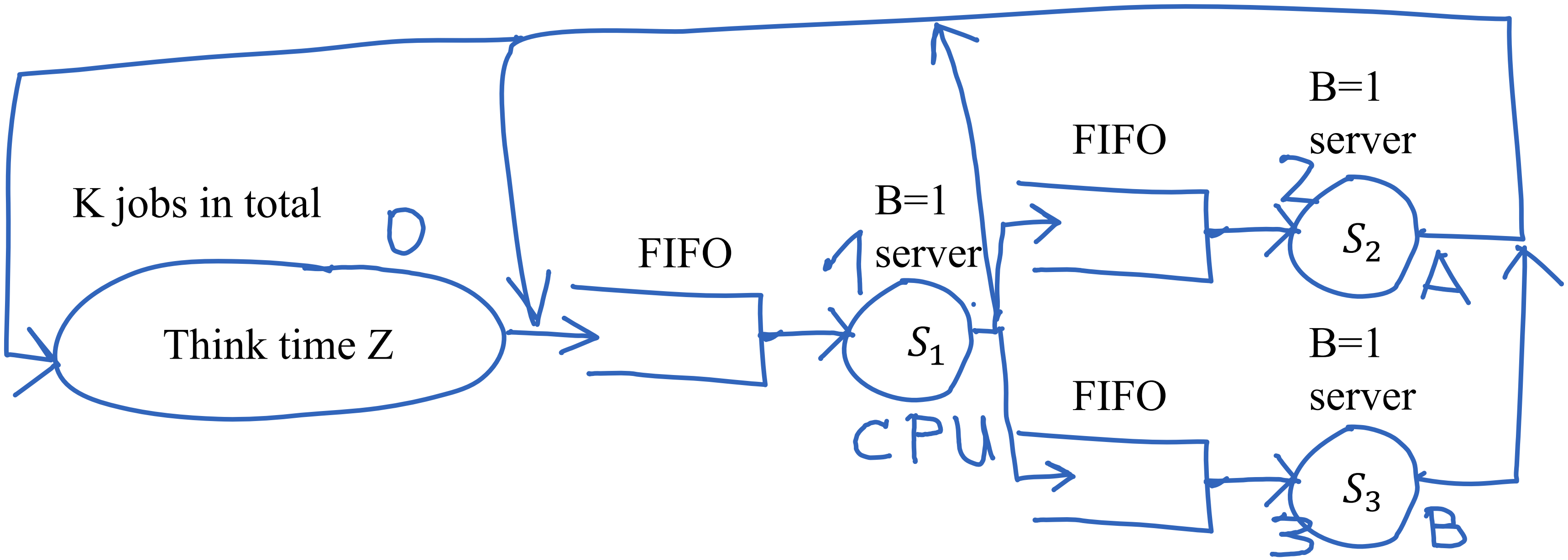
We define the numbers θ_c^s (*visit rates*) as one solution to

$$\theta_c^s = \sum_{s',c'} \theta_{c'}^{s'} q_{c',c}^{s',s} + \nu_c^s \tag{8.24}$$

If the network is open, this solution is unique and θ_c^s can be interpreted[12] as the number of arrivals per time unit of class-c customers at station s. If c belongs to a closed chain, θ_c^s is determined only up to one multiplicative constant per chain. We assume that the array $(\theta_c^s)_{s,c}$ is one non identically zero, non negative solution of Eq.(8.24).



Let us apply these results to this network



- Single class; closed
- Stations 1,2,3 are FIFO; station 0 is delay;
- Markov routing : visit rates $\theta_0 = 1; \theta_1 = 102; \theta_2 = 30; \theta_3 = 17$
- Product-Form ?

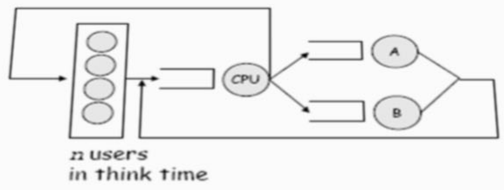
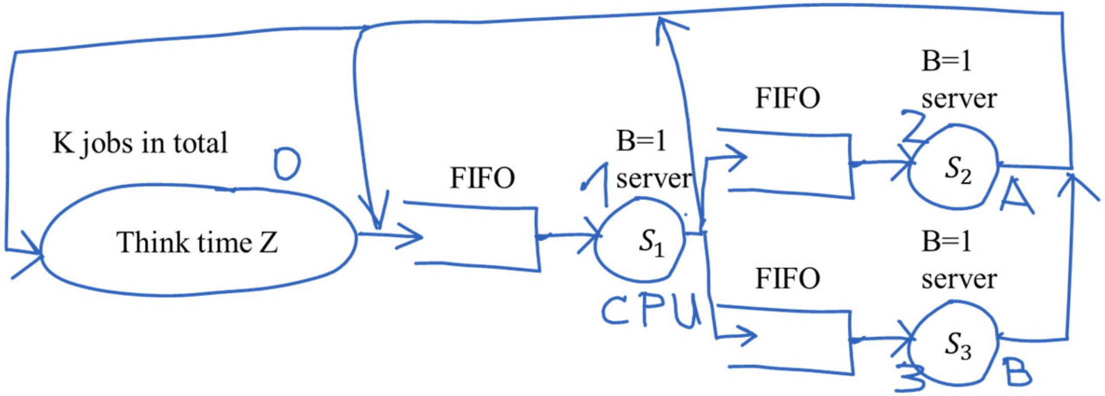


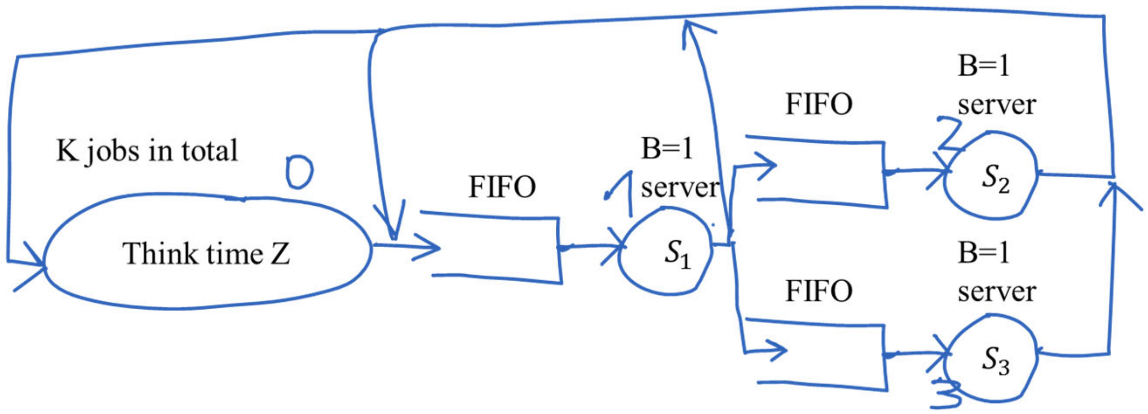
Figure 8.5: Network example used to illustrate bottleneck analysis. n attendants serve customers. Each transaction uses CPU, disk A or disk B. Av. numbers of visits per transaction: $V_{CPU} = 102, V_A = 30, V_B = 17$; av. service time per transaction: $\bar{S}_{CPU} = 0.004\ s$, $\bar{S}_A = 0.011\ s$, $\bar{S}_B = 0.013\ s$; think time $Z = 1\ s$.

Let us apply these results to this network



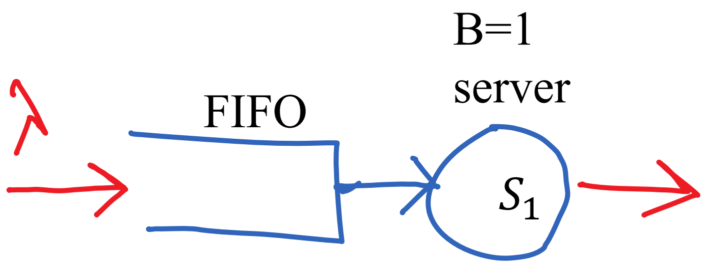
- Single class; closed
- Stations 1,2,3 are FIFO; station 0 is delay;
- Markov routing : visit rates $\theta_0 = 1; \theta_1 = 102; \theta_2 = 30; \theta_3 = 17$
- Product-Form ?
  Yes if service time at stations 1,2,3 (FIFO) are $\sim$ exponential
  No condition for station 0

The Product-Form



K jobs in total

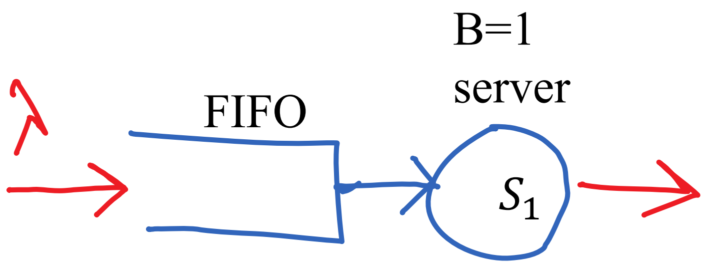
- Network is always stable (because closed)
- Product-form ⇒ $P(n_1, n_2, n_3) = \frac{1}{\eta(K)} p_1(n_1) p_2(n_2) p_3(n_3) p_0(K - n_1 - n_2 - n_3)$
- $p_1(n) = f_1(n)$ where f_1 depends on station 1 only –idem for station 2
- Let us compute f_i



Station function f_1

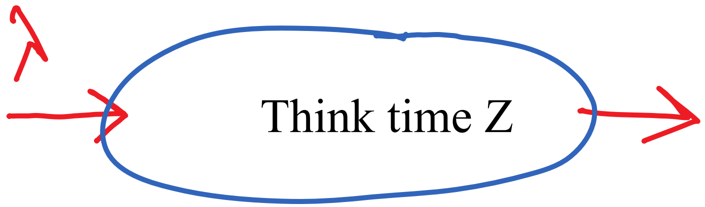
M/M/1

- Let us consider the simplest possible product-form queuing network: station 1 with Poisson arrivals
- This is a product-form network, with visit rate $\theta = \lambda$
  Therefore $P(n) = \frac{1}{\eta} f_1(n)\lambda^n$
- But this is a well-known system: M/M/1
  $P(n) = (1-\rho)\rho^n$ with $\rho = \lambda S_1$
  $P(n) = (1-\rho)S_1^n\lambda^n$
- Compare and obtain:



Station function f_1

- Let us consider the simplest possible product-form queuing network: station 1 with Poisson arrivals
- This is a product-form network, with visit rate $\theta = \lambda$
  Therefore $P(n) = \frac{1}{\eta} f_1(n)\lambda^n$
- But this is a well-known system: M/M/1
  $P(n) = (1-\rho)\rho^n$ with $\rho = \lambda S_1$
  $P(n) = (1-\rho)S_1^n\lambda^n$
- Compare and obtain: $f_1(n) = S_1^n$



Station function f_0

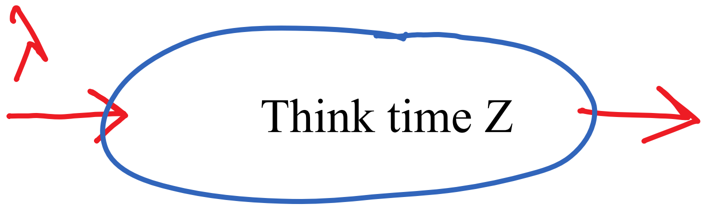
- Let us consider the simplest possible product-form queuing network: station 2 with Poisson arrivals
- This is a product-form network, with visit rate $\theta = \lambda$

  Therefore $P(n) = \frac{1}{\eta} f_2(n) \lambda^n$

- f_2 does not depend on the distribution of service time, but only on its mean (insensitive station). To obtain f_2, we may thus consider the case where the service time is exponential.
- We obtain a well-known system: M/M/∞

  $P(n) = e^{-\rho} \frac{\rho^n}{n!}$ with $\rho = \lambda Z$

Station function f_0



- Let us consider the simplest possible product-form queuing network: station 2 with Poisson arrivals
- This is a product-form network, with visit rate $\theta = \lambda$

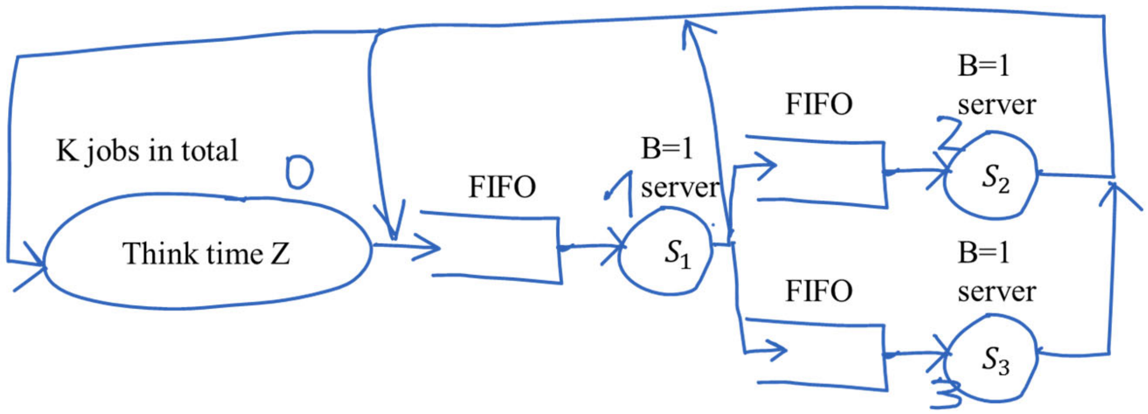
  Therefore $P(n) = \frac{1}{\eta} f_2(n) \lambda^n$

- f_2 does not depend on the distribution of service time, but only on its mean (insensitive station). To obtain f_2, we may thus consider the case where the service time is exponential.
- We obtain a well-known system: M/M/∞

  $P(n) = e^{-\rho} \frac{\rho^n}{n!}$ with $\rho = \lambda Z$

  $P(n) = e^{-\rho} \frac{Z^n}{n!} \lambda^n$

- Compare and obtain: $f_0(n) = \frac{Z^n}{n!}$



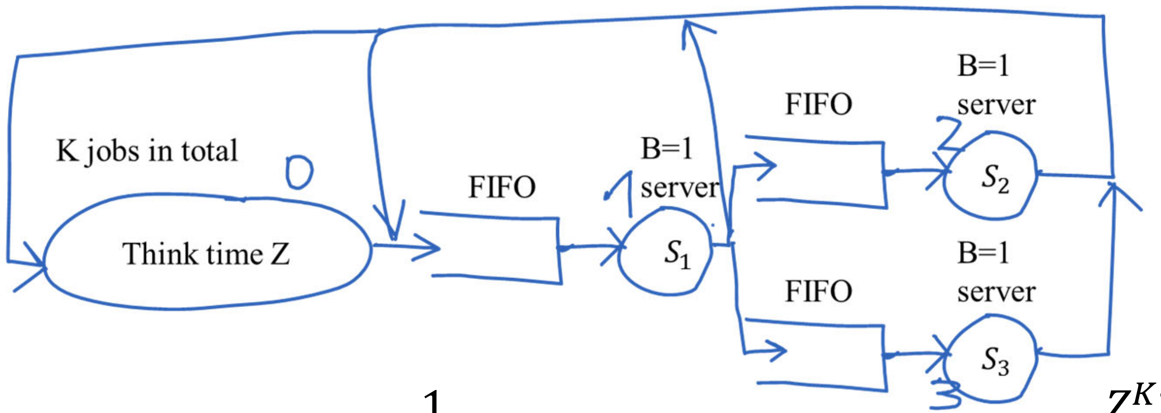
The Product-Form

- Network is always stable (because closed)
- Product-form $\Rightarrow P(n_1, n_2, n_3) = \frac{1}{\eta(K)} p_1(n_1) p_2(n_2) p_3(n_3) p_0(K - n_1 - n_2 - n_3)$

$$p_1(n_1) = (S_1\theta_1)^{n_1}$$
$$p_2(n_2) = (S_2\theta_2)^{n_2}$$
$$p_3(n_3) = (S_3\theta_3)^{n_3}$$
$$p_0(n_0) = \frac{Z^{n_0}}{n_0!}, \qquad n_0 = K - n_1 - n_2 - n_3$$

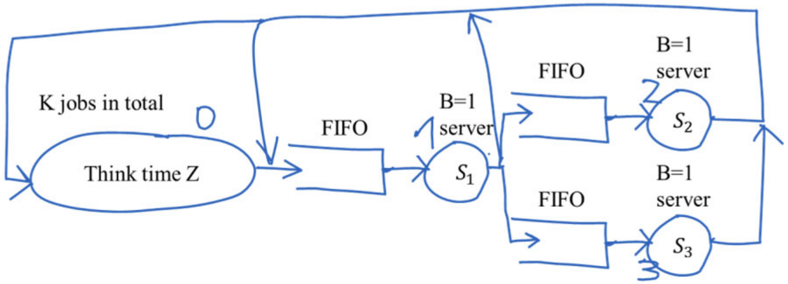
$$P(n_1, n_2, n_3) = \frac{1}{\eta(K)} (S_1\theta_1)^{n_1} (S_2\theta_2)^{n_2} (S_3\theta_3)^{n_3} \frac{Z^{K-n_1-n_2-n_3}}{(K - n_1 - n_2 - n_3)!}$$

Mean Value Analysis



$$P(n_1, n_2, n_3) = \frac{1}{\eta(K)} (S_1\theta_1)^{n_1} (S_2\theta_2)^{n_2} (S_3\theta_3)^{n_3} \frac{Z^{K-n_1-n_2-n_3}}{(K - n_1 - n_2 - n_3)!}$$

- Assume we want to compute: throughput, mean response time at station 1
- We can use direct computations but need to evaluate $\eta(K)$
  - Numerical problems for large K
  - Combinatorial explosion of number of states
- The Mean Value Algorithms does this in a smarter way



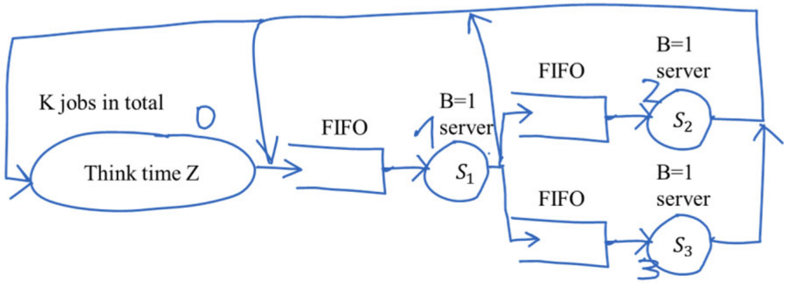
Arrival Theorem

- The distribution of customers at an arbitrary point in time is

  $P(n_1, n_2, n_3) =$

- The distribution of customers seen by a customer just before arriving at station 1 (excluding herself)

  $P^0(n_1, n_2, n_3) =$



Arrival Theorem

- The distribution of customers at an arbitrary point in time is

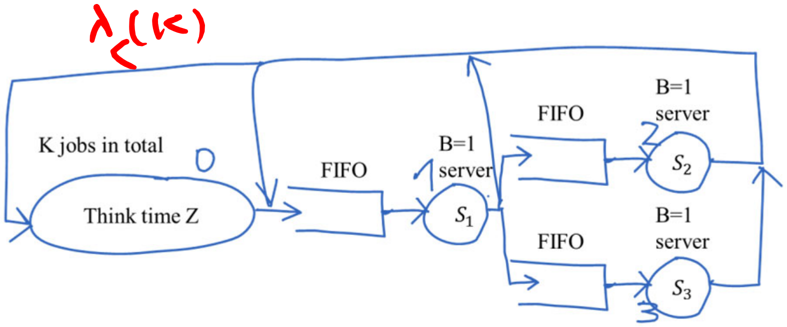
$$P(n_1, n_2, n_3) = \frac{1}{\eta(K)} (S_1\theta_1)^{n_1} (S_2\theta_2)^{n_2} (S_3\theta_3)^{n_3} \frac{Z^{K-n_1-n_2-n_3}}{(K - n_1 - n_2 - n_3)!}$$

  for $n_1 \geq 0, n_2 \geq 0, n_3 \geq 0$ and $n_1 + n_2 + n_3 \leq K$
  (and 0 otherwise)

- The distribution of customers seen by a customer just before arriving at station 1 (excluding herself)

$$P^0(n_1, n_2, n_3) = \frac{1}{\eta(K-1)} (S_1\theta_1)^{n_1} (S_2\theta_2)^{n_2} (S_3\theta_3)^{n_3} \frac{Z^{K-1-n_1-n_2-n_3}}{(K - 1 - n_1 - n_2 - n_3)!}$$

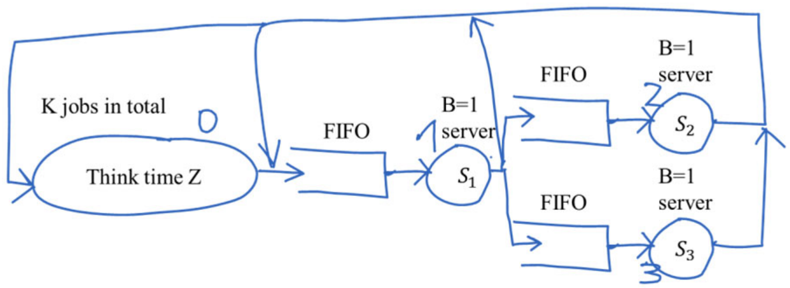
  for $n_1 \geq 0, n_2 \geq 0, n_3 \geq 0$ and $n_1 + n_2 + n_3 \leq K - 1$
  (and 0 otherwise)



Mean Value Analysis applied to our Network

- Avoids the numerical problems due to computation of normalizing constant
- Iterates on population K
  - Variables : $R_i(K)$ (response time at station i)
    $N_i(K)$ (mean number of jobs, station i)
    $\lambda(K)$ (throughput at station 0)
- Uses :
  - Arrival theorem: $R_i(K) = \left(1 + N_i(K-1)\right)S_i$ for $i = 1,2,3$
    $R_0(K) = Z$
  - Little's formula : $N_0(K) = \lambda(K)\, Z$
    $N_i(K) = \lambda(K)\theta_i\, R_i(K)$
  - Conservation of total number of customers

$$N_0(K) + N_1(K) + N_2(K) + N_3(K) = K$$



Mean Value Analysis applied to our Network

- Arrival theorem: $R_i(K) = \big(1 + N_i(K-1)\big)S_i$ for $i = 1$
  $R_0(K) = Z$
- Little's formula : $N_0(K) = \lambda(K)\, Z$
  $N_i(K) = \lambda(K)\theta_i\, R_i(K)$
- Conservation of total number of customers
  $$N_0(K) + N_1(K) + N_2(K) + N_3(K) = K$$

- Iterates on K
- At every step:
  - set $\lambda = 1$ and compute N_i
  - Obtain λ by the conservation of number of customers

$N_0 = N_1 = N_2 = N_3 = 0$;
for $k = 1{:}K$
  for $i = 1{:}3$
    $N_i = \theta_i(1 + N_i)S_i$;
  end
  $N_0 = Z$;
  $\lambda = \frac{K}{N_0+N_1+N_2+N_3}$;
  $(N_0, N_1, N_2, N_3) = \lambda(N_0, N_1, N_2, N_3)$;
end
return $(\lambda, N_0, N_1, N_2, N_3)$

Algorithm 7 MVA Version 1: Mean Value Analysis for a single chain closed multi-class product form queuing network containing only constant rate FIFO and IS stations, or stations with same station functions.

```
K = population size
λ = 0                                                        ▷ throughput
Q^s = 0 for all station s ∈ FIFO      ▷ total number of customers at station s, Q^s = Σ_c N̄_c^s
Compute the visit rates θ_c^s using Eq.(8.24) and Σ_{c=1}^{C} θ_c^1 = 1
θ^s = Σ_c θ_c^s for every s ∈ FIFO
h = Σ_{s∈IS} Σ_c θ_c^s S̄_c^s + Σ_{s∈FIFO} θ^s S̄^s                  ▷ constant term in Eq.(8.75)
for k = 1 : K do
    λ = k / (h + Σ_{s∈FIFO} θ^s Q^s S̄^s)                         ▷ Eq.(8.75)
    Q^s = λ θ^s S̄^s (1 + Q^s) for all s ∈ FIFO
end for
The throughput at station 1 is λ
The throughput of class c at station s is λθ_c^s
The mean number of customers of class c at FIFO station s is Q^s θ_c^s / θ^s
The mean number of customers of class c at IS station s is λ θ_c^s S̄_c^s
```

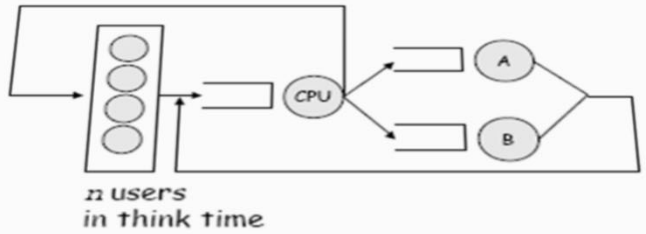


Figure 8.5: Network example used to illustrate bottleneck analysis. n attendants serve customers. Each transaction uses CPU, disk A or disk B. Av. numbers of visits per transaction: $V_{CPU} = 102, V_A = 30, V_B = 17$; av. service time per transaction: $\bar{S}_{CPU} = 0.004\,s$, $\bar{S}_A = 0.011\,s$, $\bar{S}_B = 0.013\,s$; think time $Z = 1\,s$.

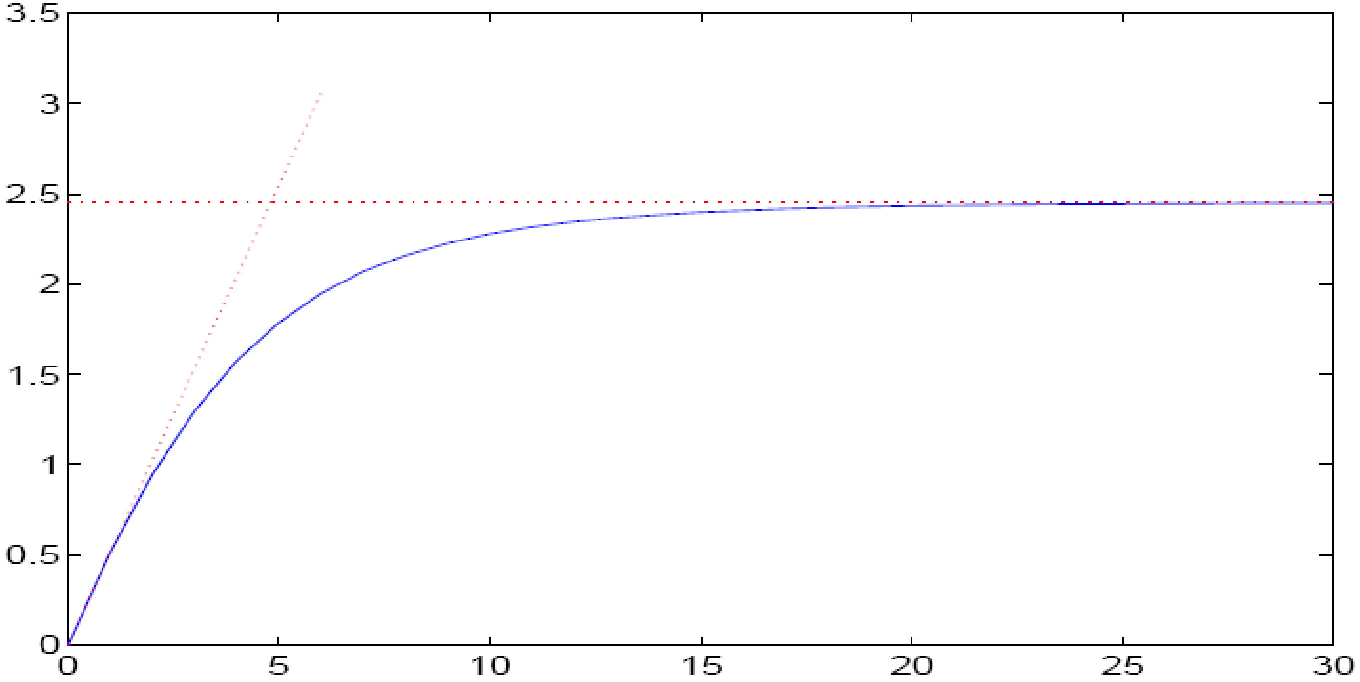


Figure 8.15: Throughput in transactions per second versus number of users, computed with MVA for the network in Figure 8.5. The dotted lines are the bounds of bottleneck analysis in Figure 8.6.

The algorithm we just used is called Mean Value Analysis (MVA) version 1

- It applies to closed product form networks where all stations are
  - FIFO or Delay
  - or equivalent (i.e. have the same function f_i)

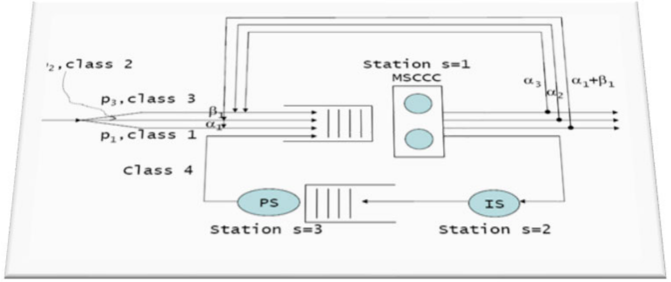
MVA Version 2

- Applies to more general networks;
- Gives not only means but also full distribs
- Uses the decomposition and complement network theorems

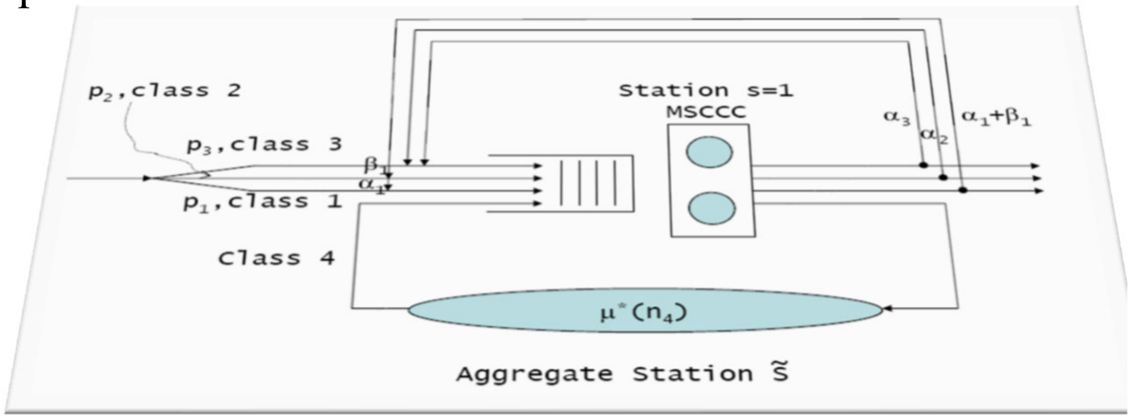
THEOREM 8.6.7. *(Decomposition Theorem [78])*
Consider a multi-class network that satisfies the hypotheses of the product form theorem 8.5.1. Any subnetwork $\mathcal{S}$ *can be replaced by its equivalent station* $\tilde{\mathcal{S}}$*, with one class per chain and station function defined by Eq.(8.80). In the resulting equivalent network* $\tilde{\mathcal{N}}$*, the stationary probability and the throughputs that are observable are the same as in the original network.*
Furthermore, if $\mathcal{C}$ *effectively visits* $\mathcal{S}$*, the equivalent service rate to chain* $\mathcal{C}$ *(closed or open) at the equivalent station* $\tilde{\mathcal{S}}$ *is*

$$\mu_{\mathcal{C}}^{*\mathcal{S}}(\vec{k}) = \lambda_{\mathcal{C}}^{*\mathcal{S}}(\vec{k}) \tag{8.81}$$

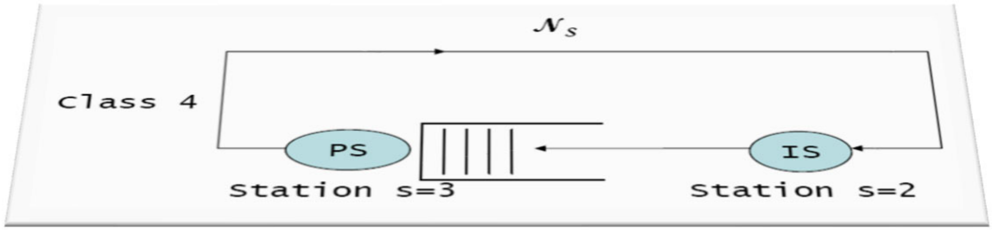
where $\lambda_{\mathcal{C}}^{*\mathcal{S}}(\vec{k})$ *is the throughput of chain* $\mathcal{C}$ *for the subnetwork in short-circuit* $\tilde{\mathcal{N}}_{\mathcal{S}}$ *when the population vector for all chains (closed or open) is* $\vec{k}$.

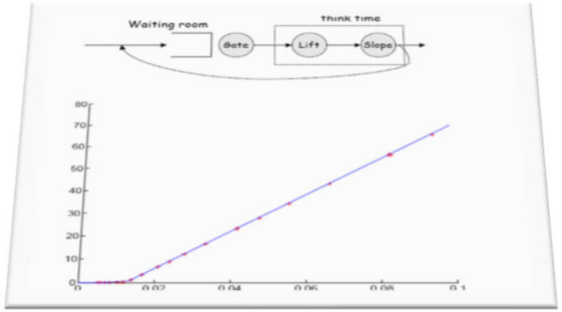


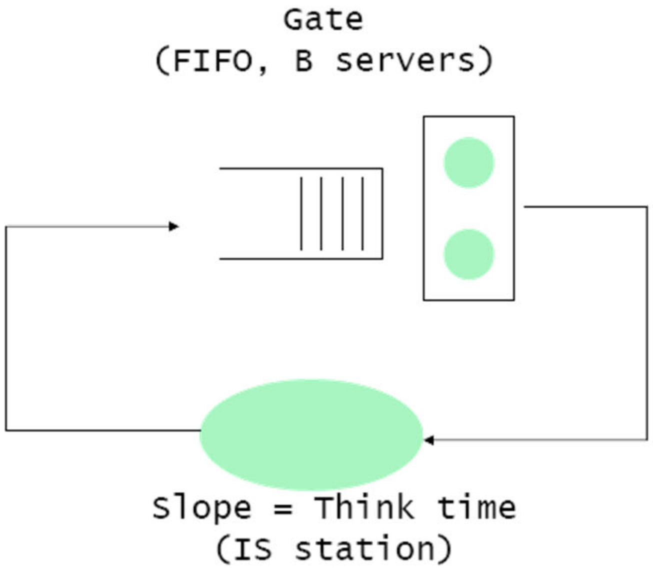
is equivalent to:



where the service rate $\mu^*(n_4)$ is the throughput of







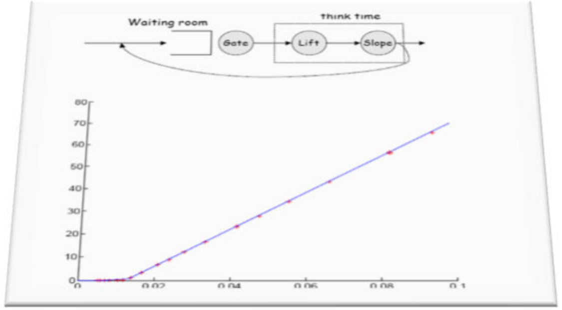
We compute $\lambda(K)$ by mean value analysis, which avoids computing the normalizing constants and the resulting overflow problems. Let $P(n|K)$ be the stationary probability that there are n customers present (in service or waiting) at the FIFO station, when the total number of customers is K. The mean value analysis equations are (Section 8.6.5):

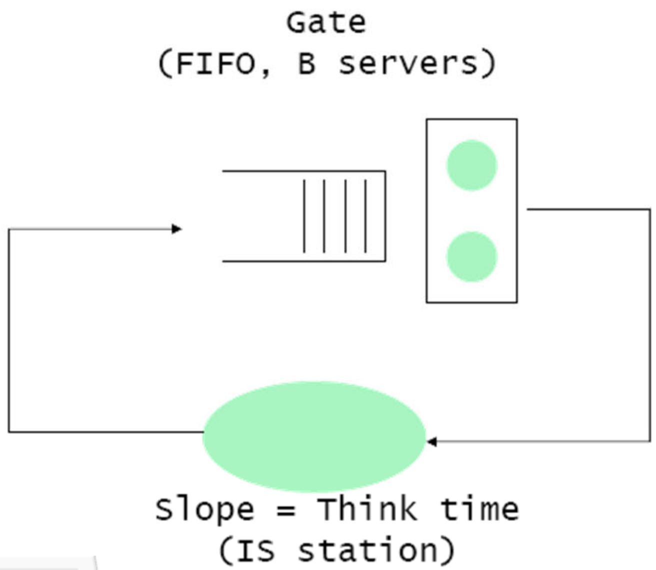
$$P(n|K) = P(n-1|K-1)\frac{\lambda(K)}{\mu^*(n)} \text{ if } n \geq 1 \quad (8.104)$$

$$P(0|K) = P(0|K-1)\frac{\lambda(K)}{\lambda^{[1]}(K)} \quad (8.105)$$

$$\sum_{n=0}^{K} P(n|K) = 1 \quad (8.106)$$

where $\mu^*(n)$ is the equivalent service rate of the FIFO station and $\lambda^{[1]}(K)$ the throughput of the complement of this station. By Table 8.1:





$$
\begin{aligned}
P(n|K) &= P(n-1|K-1)\frac{\lambda(K)}{\mu^*(n)} \text{ if } n \geq 1 \\
P(0|K) &= P(0|K-1)\frac{\lambda(K)}{\lambda^{[1]}(K)} \\
\sum_{n=0}^{K} P(n|K) &= 1
\end{aligned}
$$

$$\mu^*(n) = \frac{\min(n, B)}{\bar{S}}$$

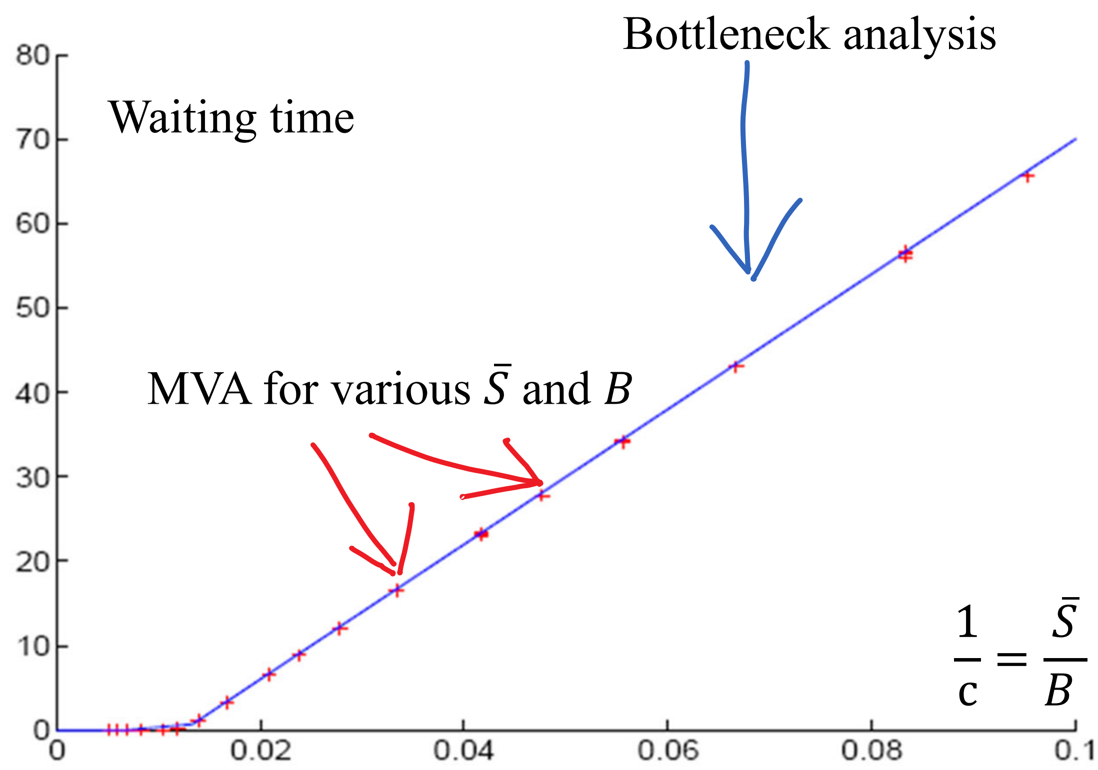
The complement network is obtained by short circuiting the FIFO station; it consists of the IS station alone. Thus

$$\lambda^{[1]}(K) = \frac{K}{\bar{Z}}$$

$$
\begin{aligned}
P(n|K) &= P(n-1|K-1)\frac{\lambda(K)}{\mu^*(n)} \text{ if } n \geq 1 \\
P(0|K) &= P(0|K-1)\frac{\lambda(K)}{\lambda^{[1]}(K)} \\
\sum_{n=0}^{K} P(n|K) &= 1
\end{aligned}
$$

Algorithm 8 Implementation of MVA Version 2 to the network in Figure 8.24.

K =: population size
$p(n)$, $n = 0...K$: probability that there are n customers at the FIFO station
λ: throughput
$p(0) = 1$, $p(n) = 0$, $n = 1...K$
for $k = 1 : K$ **do**
    $p^*(n) = p(n-1)\bar{Z} \;/\; \min(n, B)$, $n = 1...k$ ▷ Unnormalized $p(n|k)$, Eq.(8.104)
    $p^*(0) = p(0)\bar{Z}/k$ ▷ Unnormalized $p(0|k)$, Eq.(8.105)
    $\lambda = 1/\sum_{n=0}^{k} p^*(n)$
    $p(n) = p^*(n)/\lambda$, $n = 0...k$
end for



Bottleneck analysis
Waiting time
MVA for various $\bar{S}$ and B
$\frac{1}{c} = \frac{\bar{S}}{B}$
80
70
60
50
40
30
20
10
0
0
0.02
0.04
0.06
0.08
0.1

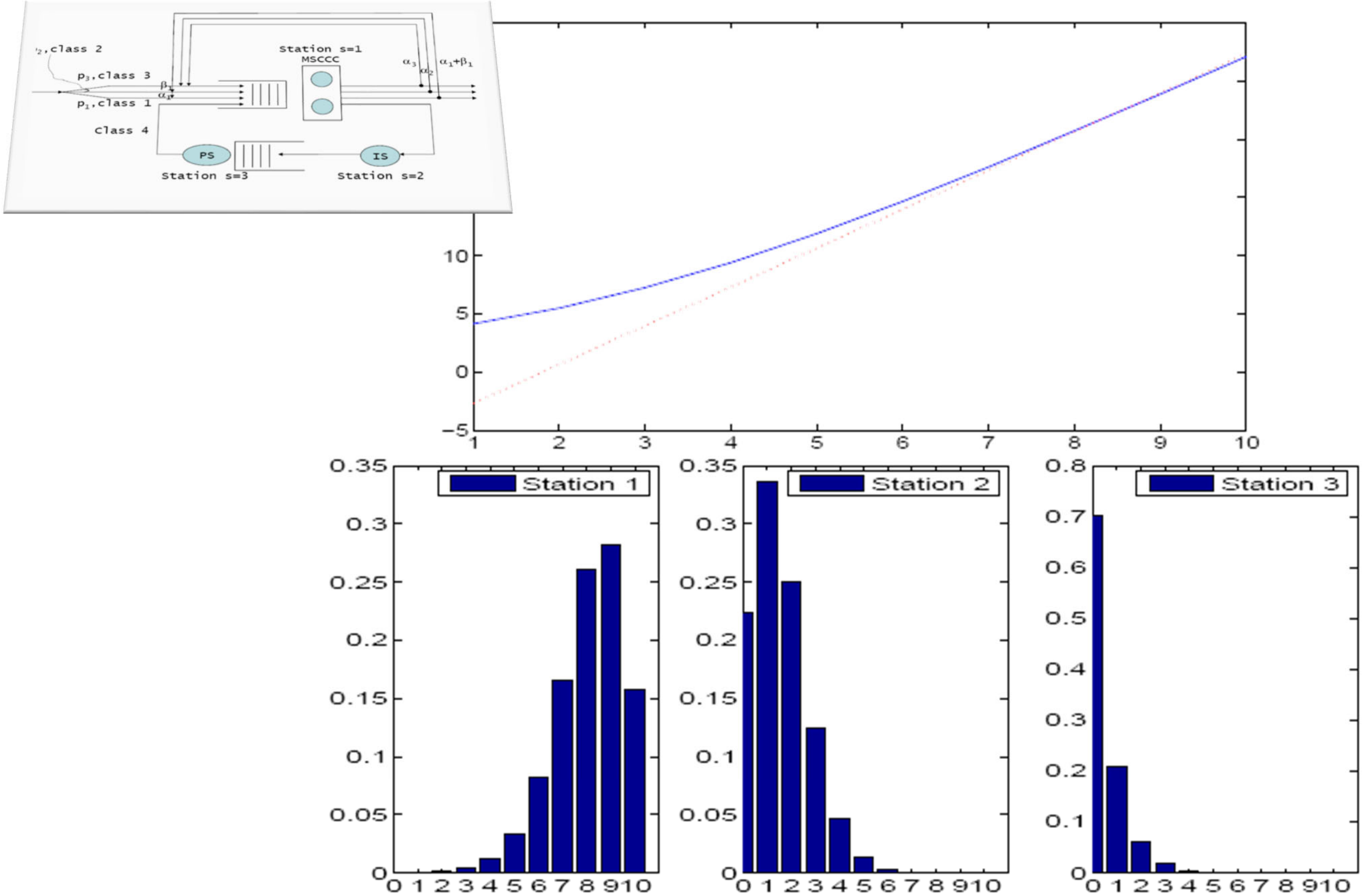


Figure 8.14: First panel: Mean Response time for internal jobs at the dual core processor, in millisecond, as a function of the number K of internal jobs. Second panel: stationary probability distribution of the number of internal jobs at stations 1 to 3, for $K = 10$. (Details of computations are in Examples 8.10 and 8.11; $\bar{S}^1 = 1, \bar{S}^2 = 5, \bar{S}^3 = 1$msec, $x = 0.7$, $y = 0.8$.)

Conclusions

- Product-form queueing networks can be analyzed with very efficient algorithms