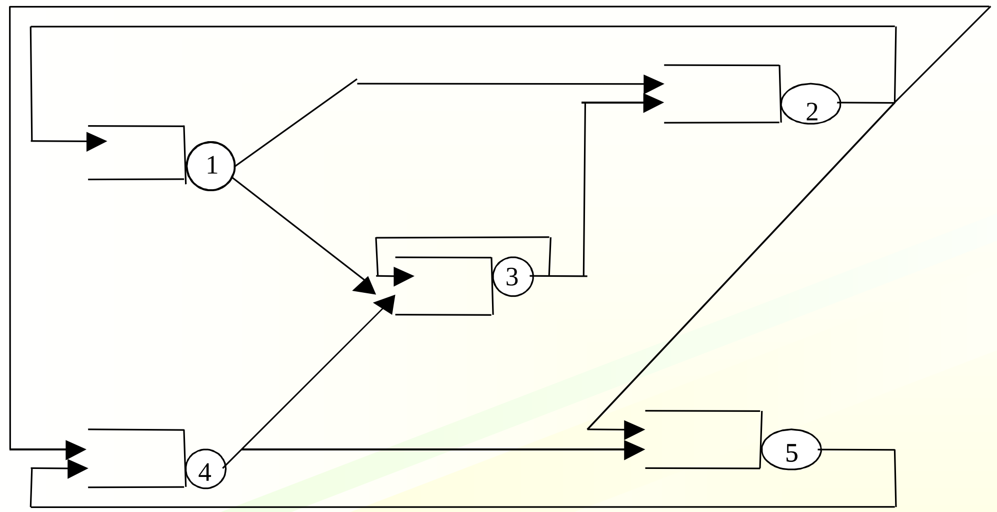
Lecture 7 : *Networks of queues*

Prof. Hee Yong Youn
College of Software
Sungkyunkwan University
Suwon, Korea
youn7147@skku.edu

Open/closed queuing network

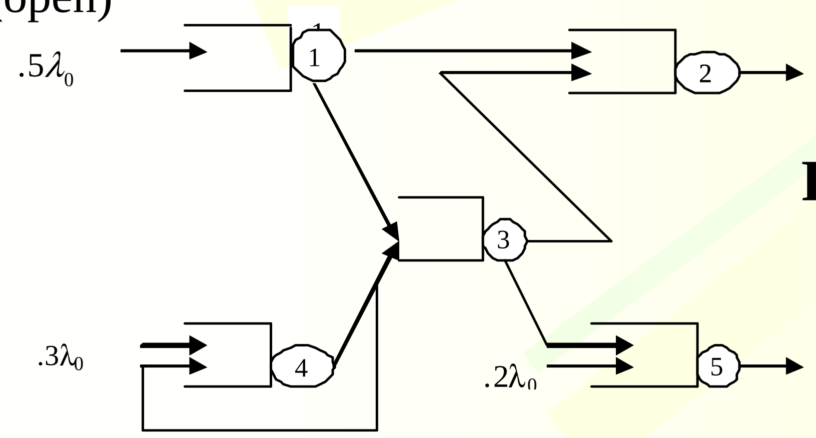
(closed)



(Routing chain)

$$P = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 1/2 & 1/2 & 0 & 0 \\ 2 & 1/3 & 0 & 0 & 1/3 & 1/3 \\ 3 & 0 & 1/2 & 1/2 & 0 & 0 \\ 4 & 0 & 0 & 1/2 & 0 & 1/2 \\ 5 & 0 & 0 & 0 & 1 & 0 \end{array}$$

(open)



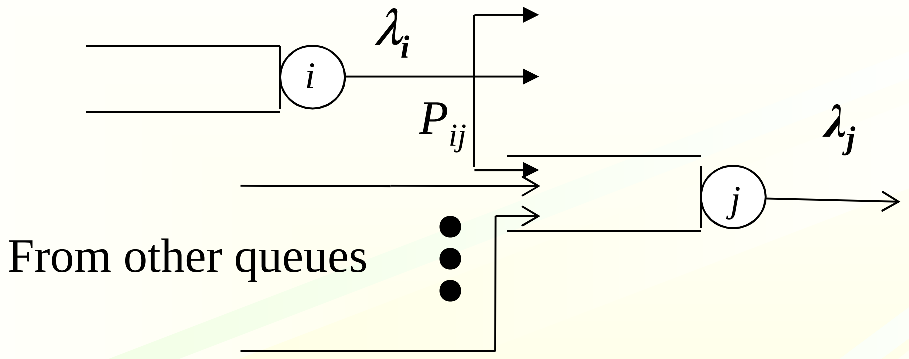
$$P = \begin{array}{c|cccccc} & 0 & 1 & 2 & 3 & 4 & 5 \\ \hline 0 & 0 & 1/2 & 0 & 0 & 3/10 & 1/5 \\ 1 & 0 & 0 & 1/2 & 1/2 & 0 & 0 \\ 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 1/2 & 0 & 0 & 1/2 \\ 4 & 0 & 0 & 0 & 1/2 & 1/2 & 0 \\ 5 & 1 & 0 & 0 & 0 & 0 & 0 \end{array}$$

$\leftarrow$ Input (virtual queue)

$\uparrow$ output

Relative throughputs and utilities

- Using routing chain, we obtain relative throughputs, visit ratios, relative utilities

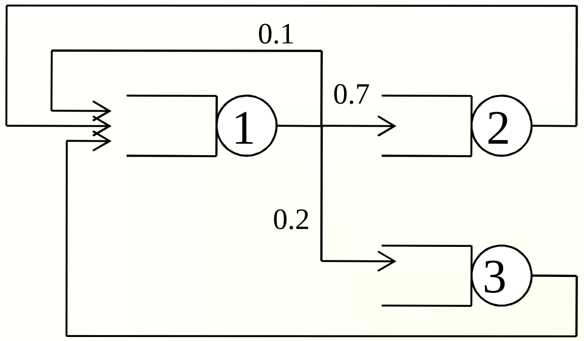


$$\vec{\lambda} P = \vec{\lambda}$$

- $\lambda_j = \lambda_1 P_{1j} + \lambda_2 P_{2j} + \ldots + \lambda_i P_{ij} + \lambda_j P_{jj} + \ldots + \lambda_m P_{mj}$ **(if there exist *m* queues)**

Relative throughput, r_i, and visit ratio, v_i

- (ex)



$$(r_1\ \ r_2\ \ r_3)\begin{bmatrix}0.1 & 0.7 & 0.2\\ 1 & 0 & 0\\ 1 & 0 & 0\end{bmatrix} = (r_1\ \ r_2\ \ r_3)$$

$$\left.\begin{array}{l}0.1r_1 + r_2 + r_3 = r_1\\ 0.7\,r_1 = r_2\\ 0.2\,r_1 = r_3\end{array}\right\} \qquad \lambda = (\alpha, 0.7\alpha, 0.2\alpha)$$

$$V_i = \frac{\lambda_i}{\lambda_1} = \frac{r_i}{r_1}$$ (larger throughput implies more _____)

Relative utility, u_i

$$\rho_i = \frac{\lambda_i}{\mu_i}, \lambda_i = \mu_i \rho_i \Leftrightarrow r_i = \mu_i u_i$$

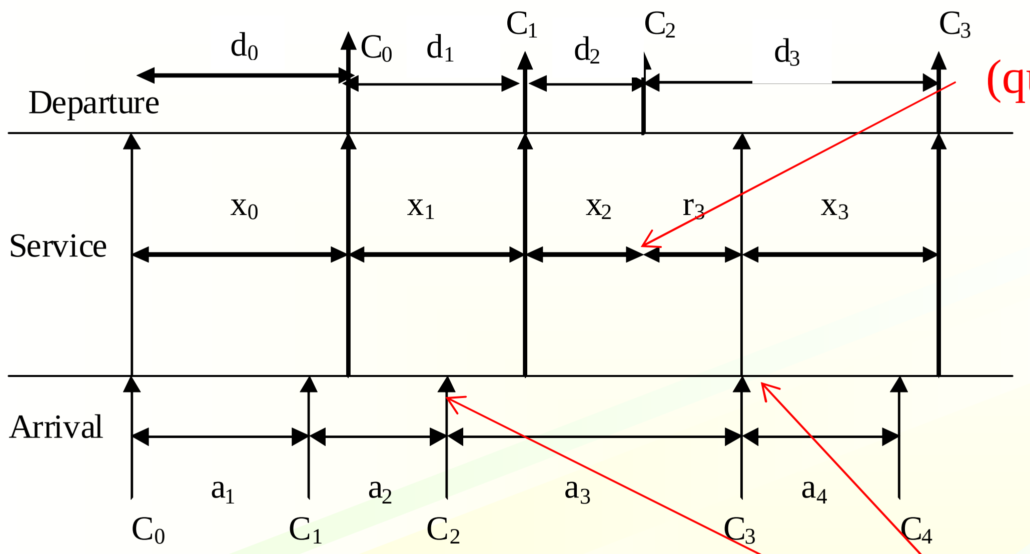
$$\sum_{i=1}^{M} r_i P_{ij} = r_j \Leftrightarrow \sum_{i=1}^{M} \mu_i u_i P_{ij} = \mu_j u_j$$

$$u_i = \frac{r_i}{\mu_i} \text{ (here } r_i \text{ is solved, } \mu_i \text{ is given)}$$

M → M property

- To study the output process of a queue
- Output process = *f*(input process, queuing discipline, service process)
- A queue has the M $\rightarrow$ M property if when the input is a Poisson process, so is the output. Then timing and order may be different (jobs may be delayed and priority orders become different) but both input and output have same Poisson parameter (λ)

FCFS M/M/1 queue



(queue becomes ______)

(Inter-departure time PDF, *d*(*t*))

$$D^*(s|busy) = \frac{\mu}{\mu+s}$$ **; (ex) $d_2 = x_2$ (when C_2 arrives, server is busy)**

$$D^*(s|idle) = \frac{\mu}{\mu+s}\cdot\frac{\lambda}{\lambda+s}$$ **; (ex) $d_3 = r_3 + x_3$ (when C_3 arrives, server is idle)**

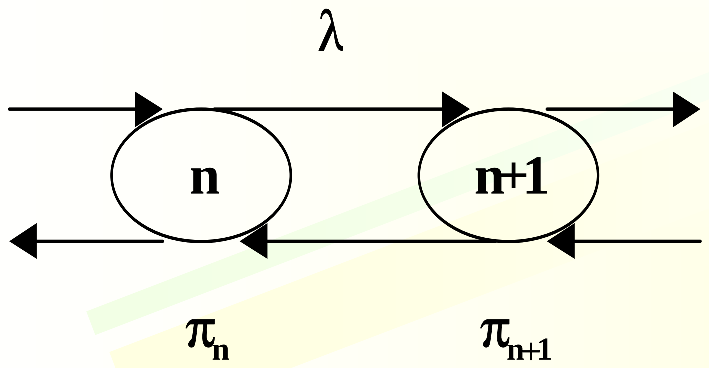
(due to ______________)

$$D^*(s) = \rho\frac{\mu}{\mu+s} + (1-\rho)[\frac{\mu}{\mu+s}\frac{\lambda}{\lambda+s}] = \frac{\lambda}{\lambda+s}$$

So M / M / 1 queue has the M $\Rightarrow$ M property

Easier checking for M $\rightarrow$ M property

- Without deriving the output process
- At steady-state



$\pi_{n+1} R(n+1 \rightarrow n) = \pi_n \lambda, \quad \forall n$
(R is the rate of departure)

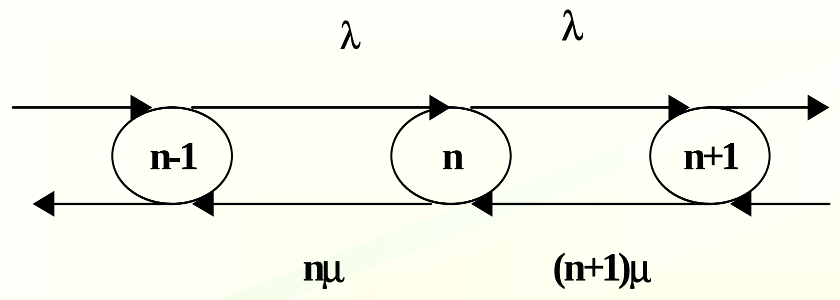
"No memory in the departure process indicates that even though the departure rate in any state is higher than the arrival rate, it cannot be compensated by a ________ number state"

- Checking is done by obtaining the steady-state probability relationship and then substituting it into the state balance equation to check the consistency

Easier checking for M ➔ M property

(ex) M/M/∞ queue

$R(n + 1 \rightarrow n) = (n + 1)\mu$



$$\pi_{n+1} = \frac{\lambda}{(n+1)\mu}\pi_n$$

$$\pi_1 = \frac{\lambda}{\mu}\pi_0, \pi_2 = \frac{\lambda}{2\mu}\pi_1 = \frac{\lambda^2}{2\mu^2}\pi_0 \ldots\ldots\ldots \pi_n = (\frac{\lambda}{\mu})^n \frac{1}{n!}\pi_0$$

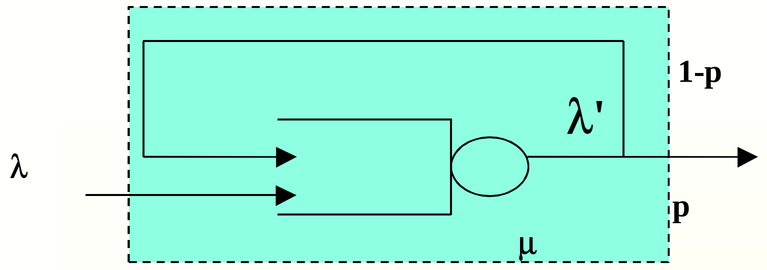
check

$$(\lambda + n\mu)\pi_n = (n+1)\mu\pi_{n+1} + \lambda\pi_{n-1}$$

$$(\lambda + n\mu)(\frac{\lambda}{\mu})^n \frac{1}{n!}\pi_0 = (n+1)\mu(\frac{\lambda}{\mu})^{n+1}\frac{\pi_0}{(n+1)!} + \lambda(\frac{\lambda}{\mu})^{n-1}\frac{\pi_0}{(n-1)!}$$

$$= (\frac{\lambda}{\mu})^n \frac{1}{n!}\pi_0(\lambda + n\mu)$$

Feedback



- Even with Poisson I/O ($M \Rightarrow M$), a feedback changes the process to hyperexponential

$$\lambda = p\lambda' \qquad \lambda' = \frac{\lambda}{p} \qquad \rho = \frac{\lambda'}{\mu} = \frac{\lambda}{p\mu}$$

- **For a stable queue,** $\rho < 1 \to \lambda < p\mu$

Local balance

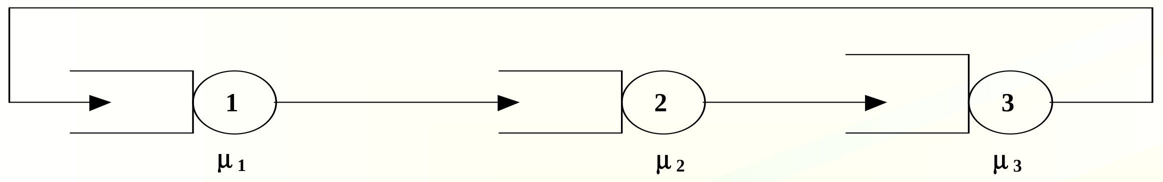
- **State of a network is the ___________ of the states of the individual queues**

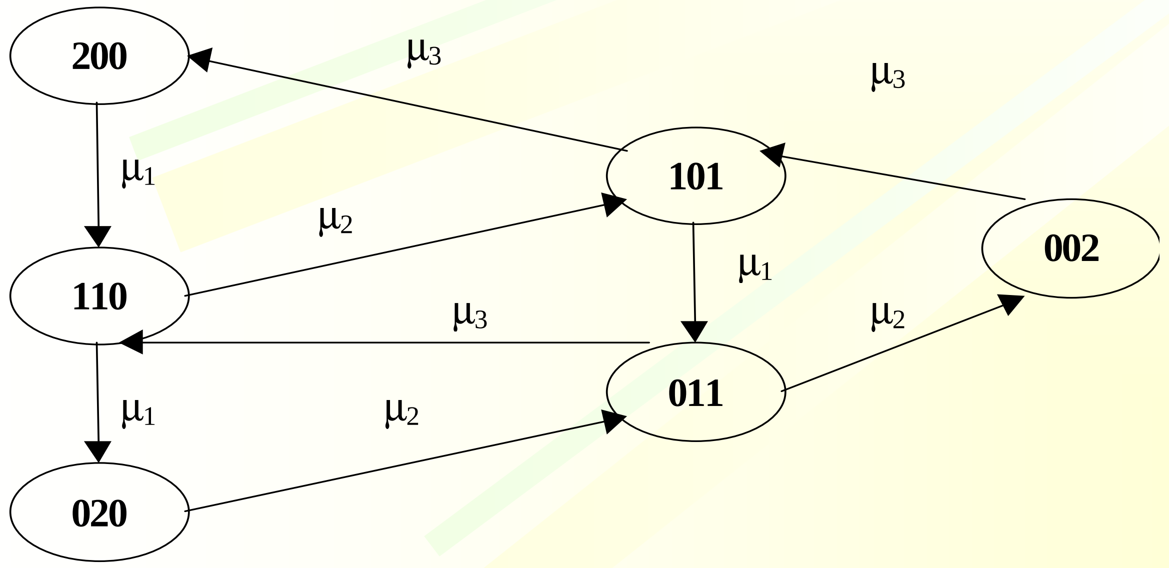
- **Jackson’s theorem: Joint PDF of the number of customers in each queue in the network in steady-state is the _________ of the marginal PDF’s irrespective of the type of input processes to the queues** (r.v. of the number of customers are ____________)

- **Local balance: the property allowing the product form of the solution**

Local balance

(ex) 2 jobs circulating in M/M/m queueing network

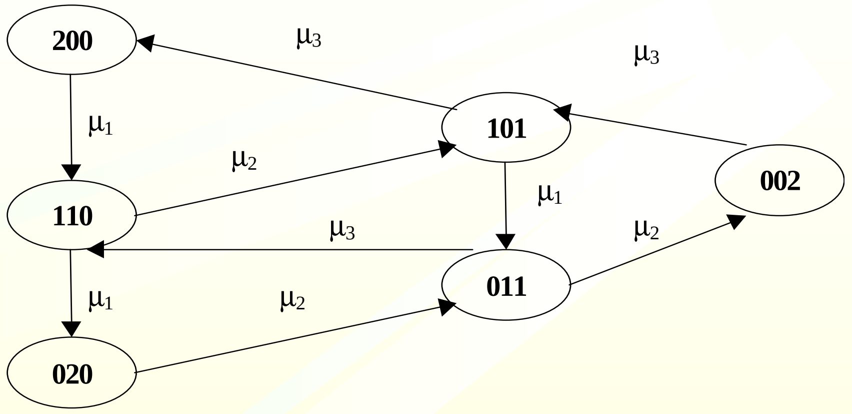




Local balance

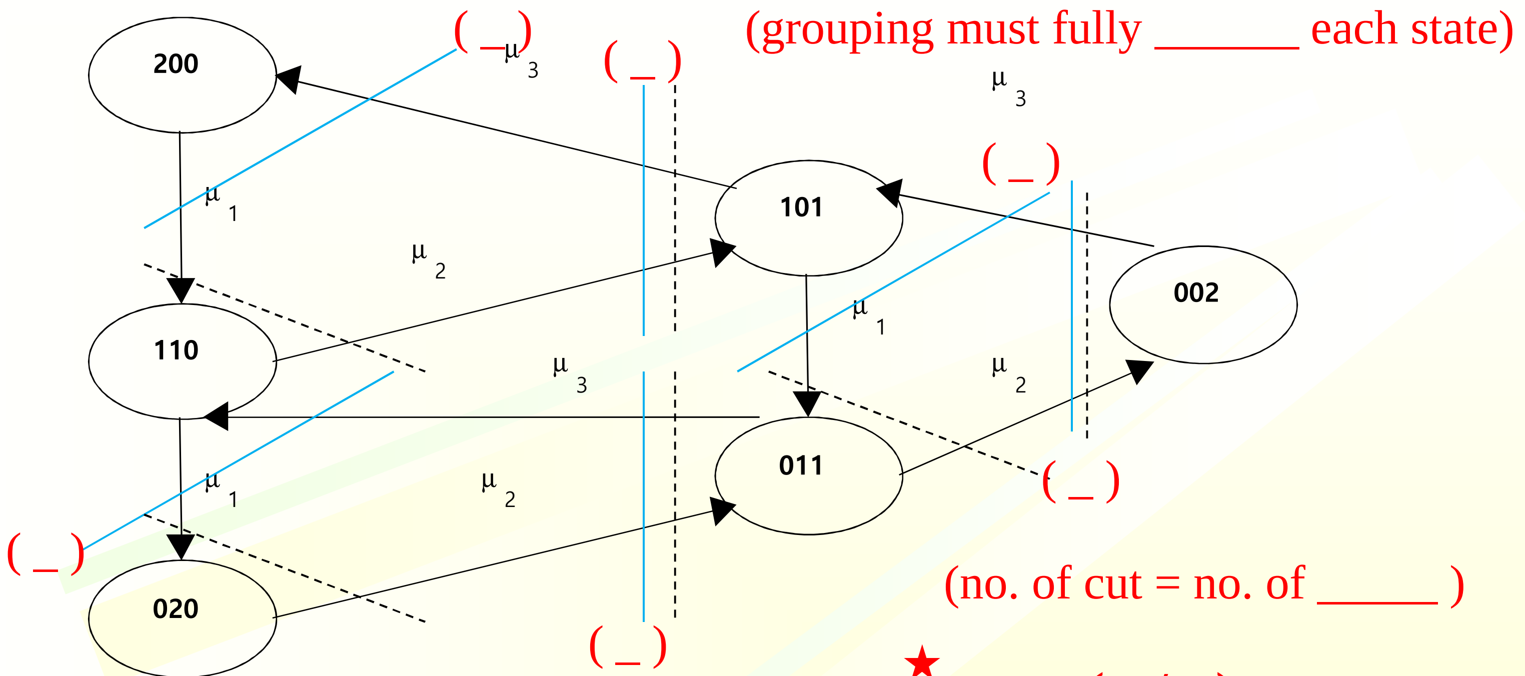
- Analysis using global balance via $\pi Q = 0$

$$\begin{cases} \mu_1\pi_{200} = \mu_3\pi_{101} \\ (\mu_1+\mu_2)\pi_{110} = \mu_1\pi_{200} + \mu_3\pi_{011} \\ \vdots \\ \mu_3\pi_{002} = \mu_2\pi_{011} \end{cases}$$



- Analysis using local balance by cutting 2 edges per state that completely isolates all the states, and check the balances → stronger requirement than global balance check

Local balance



200
101
002
110
011
020
μ_1
μ_2
μ_3
(_)
(grouping must fully ______ each state)
(no. of cut = no. of _____)

$$\begin{cases} \mu_1\pi_{200} = \mu_3\pi_{101} & \text{(a)} \\ \mu_3\pi_{011} = \mu_1\pi_{110} & \text{(b)} \\ \mu_3\pi_{011} = \mu_2\pi_{020} & \text{(c)} \\ \mu_1\pi_{101} = \mu_3\pi_{002} & \text{(d)} \\ \mu_3\pi_{002} = \mu_1\pi_{101} & \text{(e)} \\ \mu_3\pi_{002} = \mu_2\pi_{011} & \text{(f)} \\ \Sigma\pi = 1 \end{cases}$$

★ $\pi_{101} = (\mu_1/\mu_3)\,\pi_{200}$

$\pi_{020} = (\mu_3/\mu_2)\,\pi_{011} = (\mu_3/\mu_2)^2\pi_{200}$

$\pi_{002} = (\mu_1/\mu_3)\,\pi_{101} = (\mu_1/\mu_3)^2\pi_{200}$

$\pi_{110} = (\mu_3/\mu_2)\,\pi_{101} = (\mu_1/\mu_2)\pi_{200}$

$\pi_{011} = (\mu_3/\mu_2)\,\pi_{002} = (\mu_1{}^2/\mu_2\,\mu_3)\pi_{200}$

Local balance

$$\pi_{200} = \frac{\overset{200 \rightarrow}{\frac{1}{\mu_1^2}}}{\frac{1}{\mu_1^2}+\frac{1}{\mu_1\mu_3}+\frac{1}{\mu_1\mu_2}+\frac{1}{\mu_2\mu_3}+\frac{1}{\mu_2^2}+\frac{1}{\mu_3^2}} = \frac{\frac{1}{\mu_1^2}}{G(2)}$$

$$\pi_{110} = \frac{\frac{1}{\mu_1}\frac{1}{\mu_2}}{G(2)}, \ldots, \qquad — \qquad —$$

$\pi_{200} + \pi_{110} + \pi_{020} + \pi_{101} + \pi_{011} + \pi_{002} = 1$
$\pi_{200} + (\mu_1/\mu_2)\pi_{200} + (\mu_3/\mu_2)^2\pi_{200} + (\mu_1/\mu_3)$
$\pi_{200} + (\mu_1^{\,2}/\mu_2\,\mu_3)\pi_{200} + (\mu_1^{\,2}/\mu_2\,\mu_3)\pi_{200} = 1$

$$\pi_{n_1 n_2 n_3} = \frac{\left(\frac{1}{\mu_1}\right)^{n_1}\left(\frac{1}{\mu_2}\right)^{n_2}\left(\frac{1}{\mu_3}\right)^{n_3}}{G(2)}, \ n_i = 0,1,2 \ \& \sum_i n_i = 2$$

- $\pi_{n_1\, n_2\, n_3}$ **will satisfy the global balance equations**

(ex) Check with $(\mu_1+\mu_2)\,\pi_{110} = \mu_1\,\pi_{200} + \mu_3\pi_{011}$

Product form solution

- Local balance solution technique used in networks is similar to the technique used for birth-death systems

- General B-D queue: $\pi_k = (\prod_{j=0}^{k-1} \frac{\lambda_j}{\mu_{j+1}})\pi_0$

  M/M/1 queue: $\pi_k = \rho^k \pi_0 = \rho^k (1-\rho)$

  M/M/∞ queue: $\pi_k = \frac{\rho^k}{k!}\pi_0 = \frac{\rho^k}{k!} e^{-\frac{\lambda}{\mu}}$

Product form solution

- General queueing network of single server queues satisfying local balance

$$\pi_{n_1 n_2 \ldots n_M} = \frac{\rho_1^{n_1} \rho_2^{n_2} \ldots \rho_M^{n_M}}{C}$$ (C is normalization constant for making it a PDF)

- $$C = \begin{cases} \dfrac{1}{\pi_{00\ldots0}} & \text{(for open NWs)} \\ \text{P[N customers in NW] (for closed NWs)} \end{cases}$$

Product form solution

- Open NW of M/M/1 queues

$$\pi_{n_1 n_2 \ldots n_M} = \underbrace{\overbrace{(1-\rho_1)}^{\bigstar}\rho_1^{n_1}}_{\_\_\_} \overbrace{(1-\rho_2)}^{\bullet\bullet\bullet}\rho_2^{n_2} \ldots \overbrace{(1-\rho_M)}^{= \_\_\_\_\_}\rho_M^{n_M}$$

(calculated from _______ ________, as λ_M/μ_M)

- Open NW of M/M/S_i queues

  S_i: no. of servers in queue-i

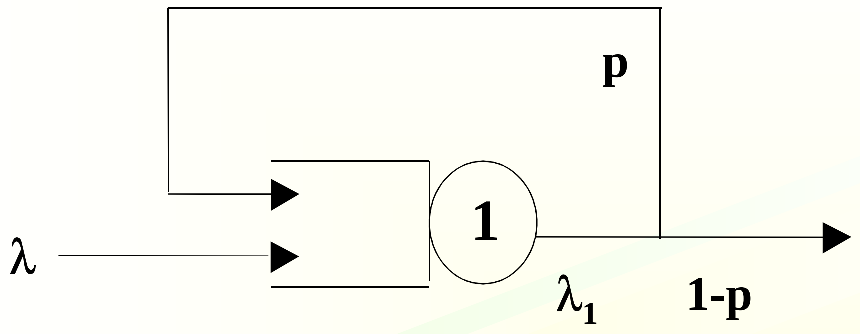
$$f_i(n_i) \equiv \begin{cases} \dfrac{n_i!}{S_i^{n_i}} & n_i < S_i \\ \dfrac{S_i!}{S_i^{S_i}} & n_i \geq S_i \end{cases} \qquad \pi_{n_1 n_2 \ldots n_M} = \frac{\rho_1^{n_1}\rho_2^{n_2}\ldots\rho_M^{n_M}}{f_1(n_1)f_2(n_2)\ldots f_M(n_M)C}$$

Product form solution summary

- Short cut solutions (product form) possible with local balance NWs
- Any NW composed of queues, each having the M $\Rightarrow$ M property, will satisfy local balance
- NWs constructed from M $\Rightarrow$ M queues permit product form solutions
- The queue types yielding product form NWs
  - FCFS with exp. dist. service time.
  - LCFS preemptive-resume, processor sharing, infinite server with Coxian distribution service time

Solving open networks

(ex)



Avg. no. of customers in the queue

$$\textbf{E[n] = N}_1 = \frac{\rho_1}{1-\rho_1} = \frac{\frac{\lambda_1}{\mu_1}}{1-\frac{\lambda_1}{\mu_1}} = \frac{\frac{\lambda}{\mu_1(1-p)}}{1-\frac{\lambda}{\mu_1(1-p)}} = \frac{\lambda}{\mu_1(1-p)-\lambda}$$

Solving open networks

T_1 = Avg. time in the queue considering only the queue and server

$$= \frac{N_1}{\lambda_1} = \frac{\lambda}{\frac{\lambda}{1-p}(\mu_1(1-p)-\lambda)} = \frac{1}{\mu_1 - \frac{\lambda}{1-p}} = \frac{1-p}{\mu_1(1-p)-\lambda}$$

T_1 = Avg. time in the queue considering the feedback also

$$= \frac{N_1}{\lambda} = \frac{1}{\mu_1(1-p)-\lambda}$$

Solving open networks

T_1 = Avg. time in the queue considering only the queue and server

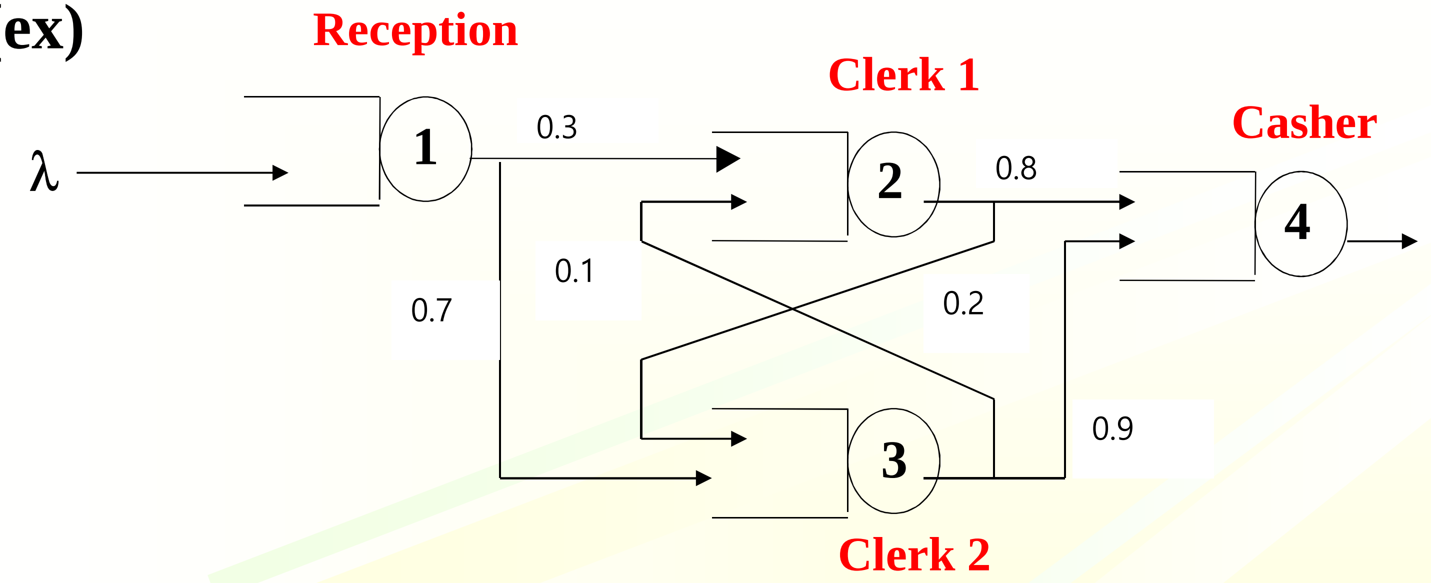
$$= \frac{N_1}{\lambda_1} = \frac{\lambda}{\frac{\lambda}{1-p}(\mu_1(1-p)-\lambda)} = \frac{1}{\mu_1 - \frac{\lambda}{1-p}} = \frac{1-p}{\mu_1(1-p)-\lambda}$$

T = Avg. time in the queue considering the feedback also

$$= \frac{N_1}{\lambda} = \frac{1}{\mu_1(1-p)-\lambda}$$

Solving open networks

(ex)



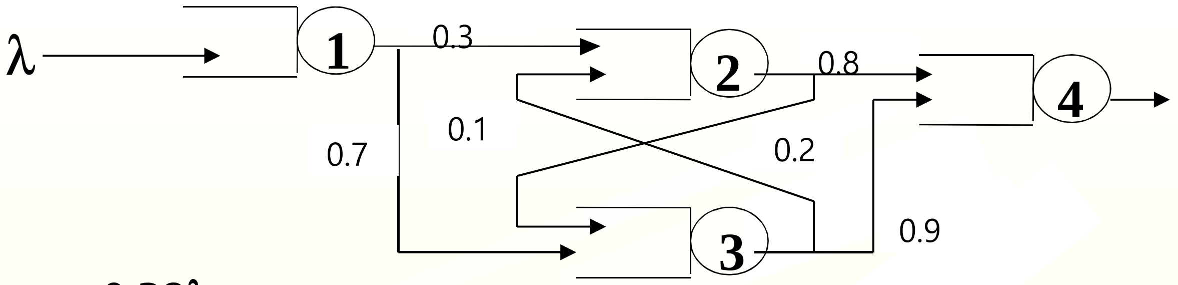
- FCFS queue with exp. dist. service

$$\mu_1 = \frac{1}{20\,\text{sec}}, \quad \mu_2 = \frac{1}{10\,\text{min}} = \frac{1}{600\,\text{sec}}$$

$$\mu_3 = \frac{1}{5\,\text{min}} = \frac{1}{300\,\text{sec}}, \quad \mu_4 = \frac{1}{1\,\text{min}} = \frac{1}{60\,\text{sec}}$$

Solving open networks

$$\lambda_1 = \lambda,\ \rho_1 = \frac{\lambda}{\mu_1} = 20\lambda$$



$$\left.\begin{array}{l}\lambda_2 = 0.3\lambda + 0.1\lambda_3 \\ \\ \lambda_3 = 0.7\lambda + 0.2\lambda_2\end{array}\right\} \begin{array}{l}\lambda_2 = 0.38\lambda,\ \rho_2 = \dfrac{0.38\lambda}{\frac{1}{600}} = 228\lambda \\ \\ \lambda_3 = 0.78\lambda,\ \rho_3 = \dfrac{0.78\lambda}{\frac{1}{300}} = 234\lambda\end{array}$$

$$\lambda_4 = 0.8\lambda_2 + 0.9\lambda_3 = \lambda,\ \rho_4 = \frac{\lambda}{\frac{1}{60}} = 60\lambda$$

Solving open networks

$$\rho_n < 1 \rightarrow \lambda < \frac{1}{234} < 15.38/\text{hour}$$

$$\pi_{n_1 n_2 n_3 n_4} = (1-\rho_1)\rho_1^{n_1}(1-\rho_2)\rho_2^{n_2}(1-\rho_3)\rho_3^{n_3}(1-\rho_4)\rho_4^{n_4}$$

$$E[n_1] = \frac{\rho_1}{1-\rho_1} = \frac{20\lambda}{1-20\lambda} \leq 0.093 \quad \text{for } \lambda \leq 0.00427/\text{sec}$$

$$E[n_2] = \frac{\rho_2}{1-\rho_2} = \frac{228\lambda}{1-228\lambda} \leq 36.82$$

Solving open networks

$$E[n_3] = \frac{\rho_3}{1-\rho_3} = \frac{234\lambda}{1-234\lambda} \leq \infty$$

$$E[n_4] = \frac{\rho_4}{1-\rho_4} = \frac{60\lambda}{1-60\lambda} \leq 0.34 \qquad N = \sum_{j=1}^{4} E[n_j]$$

- Avg. time to process,

$$T = \frac{N}{\lambda} \quad (= 23 \text{ min for } \lambda = 10 / \text{hour})$$

- Avg. time to visit queue 1,3, and then 4,

$$T_R = \frac{E[n_1]}{\lambda_1} + \frac{E[n_3]}{\lambda_3} + \frac{E[n_4]}{\lambda_4}$$

Solving closed networks

- Need utilization and normalization constant (sum of all combination of utilization)

- Use relative utilization
  - $r_m = u_m \mu_m = \alpha\lambda_m$

- Normalization constant (with N customers in the network)
  - $$G(N) = \sum_{\forall n, \sum_{i=1}^{M} n_i = N} \prod_{i=1}^{M} \frac{u_i^{n_i}}{\beta_i(n_i)} \qquad \beta_i(n_i) = \begin{cases} n_i! & n_i \le S_i \\ S_i!S_i^{(n_i - S_i)} & n_i > S_i \end{cases}$$

Solving closed networks

- Prob. that queue-m has n or more customers
  - $P[n_m \geq n] = u_m^n \frac{G(N-n)}{G(N)}$

    (*n* customers are already in queue-*m*, while the remaining (*N*-*n*) are ___________)
- Solution involves
  - Relative utility (u_m)
  - Relative throughput (r_m)
  - Normalization constant (G(N))

Solving closed networks

- $r_m = u_m \mu_m = \alpha \lambda_m$
  - α: scalar required to change relative to absolute
- $\rho_m = u_m \frac{G(N-1)}{G(N)}$ $\lambda_m = r_m \frac{G(N-1)}{G(N)} = \mu_m u_m \frac{G(N-1)}{G(N)}$

$$E[n] = \sum_{n=1}^{\infty} nP[n] = \sum_{n=1}^{\infty} \sum_{k=n}^{\infty} P[k] = \sum_{n=1}^{\infty} P[k \geq n]$$

$$E[n_m] = \sum_{n=1}^{N} u_m^n \frac{G(N-n)}{G(N)}$$

Generating the normalization constant

- **Generating the normalization constant – (Convolution method)**
  - By Buzen
  - Partial sum including the terms for n ≤ N customers in m ≤ M of the queues

$$g(n,m) = \sum_{(\sum_{i=1}^{m} n_i)=n} \prod_{i=1}^{m} u_i^{n_i}$$

$$= (u_1^n + u_1^{n-1}u_2 + \dots + u_1^{n-1}u_m + u_1^{n-2}u_2^2 + \dots + u_m^n)$$

$$g(n\text{-}1,m) = (u_1^{n-1} + u_1^{n-2}u_2 + \dots + u_1^{n-2}u_m + u_1^{n-3}u_2^2 + \dots + u_m^{n-1})$$

$$g(n,m\text{-}1) = (u_1^n + u_1^{n-1}u_2 + \dots + u_1^{n-1}u_{m-1} + u_1^{n-2}u_2^2 + \dots + u_{m-1}^{n-1})$$

$$g(n,m) = g(n,m\text{-}1) + u_m g(n\text{-}1,m)$$

$$g(n,0) = 0,\ g(0,m) = 1,\ g(n,1) = u_1^n,\ G(N) = g(N,M)$$

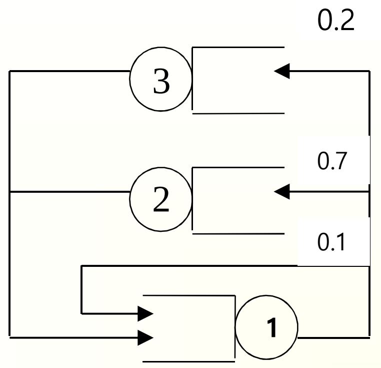
Generating the normalization constant

n\m	0	1	2				m				M
0	0	1	1	.	.	.	1	.	.	.	1
1	0	g(1,1)	g(1,2)	.	.	.	g(1,m)	.	.	.	G(1)
2	0	g(2,1)	g(2,2)	.	.	.	g(2,m)	.	.	.	G(2)
.											
.											
.											
n	0	g(n,1)	g(n,2)	.	.	.	g(n,m)	.	.	.	G(n)
.											
.											
.											
N	0	g(N,1)	g(N,2)	.	.	.	g(N,m)	.	.	.	G(N)

(ex) $g(2,2) = g(2,1) + u_2 g(1,2)$

Generating the normalization constant

- (ex)



Queue-1: processor	28msec	$\mu_1 = \frac{1}{0.028\,\text{sec}}$
Queue-2: 5.25" HD	40msec	$\mu_2 = \frac{1}{0.04\,\text{sec}}$
Queue-3: 8" HD	280msec	$\mu_3 = \frac{1}{0.28\,\text{sec}}$

$$\mu_2 u_2 = 0.7\mu_1 u_1 \qquad \mu_3 u_3 = 0.2\mu_1 u_1$$

$$\frac{1}{40} u_2 = 0.7\frac{1}{28} u_1 \qquad \frac{1}{280} u_3 = 0.\,2\frac{1}{28} u_1$$

$$u_2 = u_1 \qquad u_3 = 2u_1$$

Generating the normalization constant

- $g(n,m) = g(n,m-1) + u_m g(n-1,m)$

  $u_1 \to 1;\ u_2 = u_1 = 1;\ u_3 = 2u_1 = 2$

n\m	0	1	2	3
0	0	1	1	1=G(0)
1	0	1	2	4
2	0	1	3	11
3	0	1	4	26
4	0	1	5	57
5	0	1	6	120
6	0	1	7	247
7	0	1	8	502=G(7)

$g(1,1) = 0 + u_1 * 1 = u_1 = 1$
$g(1,2) = u_1 + u_2 * 1 = 2u_1 = 2$
$g(1,3) = u_1 + u_2 + u_3 * 1 = 4u_1 = 4$
$g(2,1) = 0 + u_1 * 1 = 1$
$g(2,2) = 1 + u_2 * 2 = 3$
$g(2,3) = 3 + u_3 * 4 = 11$
$g(3,1) = 0 + u_1 * 1 = 1$
$g(3,2) = 1 + u_2 * 3 = 4$
$g(3,3) = 4 + u_3 * 11 = 26$

Generating the normalization constant

N=7

$\rho_1 = u_1 \frac{G(6)}{G(7)} = 1 * \frac{247}{502} = 0.492,\ \rho_2 = \rho_1,\ \rho_3 = u_3 \frac{247}{502} = 0.984$

$\lambda_1 = \rho_1 \mu_1 = 0.492 * \frac{1}{0.028} = 17.57/\text{sec}$

$\lambda_2 = 12.3/\text{sec},\ \lambda_3 = 3.51/\text{sec}$

$$E[n_m] = \sum_{n=1}^{N} u_m^n \frac{G(N-n)}{G(N)},\quad u_1^n = 1$$

$$E[n_1] = E[n_2] = \frac{1+4+\ldots+247}{502} = 0.928$$

$$E[n_3] = \frac{2^7 \times 1 + 2^6 \times 4 + \bullet\bullet\bullet + 2^1 \times 247}{502} = 5.143$$

$T_1 = \frac{E[n_1]}{\lambda_1} = \frac{0.928}{17.57} = 0.0528$ sec, $T_2 = \frac{E[n_2]}{\lambda_2} = 0.0755$,

$T_3 = 1.465$

$\pi_{n1,n2,7-n1-n2} = (1/502)1^{n1}1^{n2}2^{7-n1-n2}$

Generating the normalization constant

- (Ex 7.3) What is the utilization of queue-1 with 4 customers?

  $\rho_1 = u_1(G(3)/G(\_)) = 1x\ 26/57 = 0.456$

- (Ex 7.4) What is the utilization of queue-3 with 4 customers?

  $\rho_3 = \_\_ (G(3)/G(4)) = 2x\ 26/57 = 0.912$

- (Ex 7.5) What is the normalization constant with 8 customers?

  7 1 8 502

  8 1 9 _________ =1013

- (Ex 7.6) What is the utilization of queue-1 with 8 customers?

  $\rho_1 = u_1(G(\_)/G(8)) = 1x\ 502/1013 = 0.496$

- What is the prob. for queue-1 to have six or more customers with 7 customers in the network?

$P[n_1 \geq 6] = \_\_\_ (G(1)/G(7)) = 4/502 = 0.00797$ or

$P[n_1 \geq 6] = \pi_{610} + \pi_{601} + \_\_\_\_ = (1^6 1^1 2^0 + 1^6 1^0 2^1 + 1^7 1^0 2^0)/502 = 4/502$

$(= 1^6 (1^0 1^1 2^0 + 1^0 1^0 2^1 + 1^1 1^0 2^0)/502$

$= u_1^6 (G(1)/G(7)))$

Mean Value Analysis (MVA)

- The normalization constant method may cause overflow or underflow when automated

  $\rho_m = u_m(G(N-1)/G(N))$; if $\rho_m > u_m$, G(N) gets very ______ ; if $\rho_m < u_m$, very large

- One solution is scaling the relative utilization

- MVA avoids the problem by computing averages.
  It also computes system parameters for all populations less than the target level as well

- But it cannot provide specific prob. dist. :

$$P[n_m \geq n] = u_m^n \frac{G(N-n)}{G(N)}$$

- Little’s law is applied to each queue and network as a whole.

The MV theorem

- The MV theorem (by Lavenberg and Reiser)

"A customer arriving at a queue sees the same dist. of customers as does an observer outside of the network if one less customer was circulating."

(since itself needs to be ________)

(Good for product form NWs, approximation for others)

MVA procedure

```
For n := 1 to N
(* Iterate through NW population to target, N *)

    For m := 1 to M
    (* Iterate through all queues *)
                (average ________ time)
        $T_m$ =  (* Application of  MV theorem *)
        $n_m(0) = 0$        (just ________)

        $\lambda = \frac{n}{\sum_{m=1}^{M} T_m V_m}$

        $\lambda_m = \lambda V_m$

        $n_m(n) = \lambda_m T_m$
```

MVA procedure

- (ex) same example as for the convolution method

  n=1;

  $n_m(0) = 0$, $T_1 = 0.028\{1+0\} = 0.028$ sec

  $T_2 = 0.04$, $T_3 = 0.28$

$$\lambda = \frac{n}{\sum_{m=1}^{M} T_m V_m} = \frac{1}{0.028 + 0.04 * 0.7 + 0.28 * 0.2} = \frac{1}{0.112} = 8.93/\mathrm{sec}$$

  $\lambda_m = \lambda V_m$; $\lambda_1 = 8.93$, $\lambda_2 = 6.25$, $\lambda_3 = 1.786$

  $n_m(n) = \lambda_m T_m$

  $n_1(1) = 0.25$, $n_2(1) = 0.25$, $n_3(1) = 0.5$

n\m	0	1	2	3
0	0	1	1	1=G(0)
1	0	1	2	4
2	0	1	3	11
3	0	1	4	26
4	0	1	5	57
5	0	1	6	120
6	0	1	7	247
7	0	1	8	502=G(7)

(From convolution, $\pi_{100} = (1/4)1^1 1^0 2^0 = 0.25$;$\pi_{010} = (1/4)1^0 1^1 2^0 = 0.25$;$\pi_{001} = (1/4)$______ $= 0.5$)

MVA procedure

$n=2;$

$T_1= 0.28\{1+0.25\} = 0.035$ sec

$T_2= 0.05,\ T_3=0.42$

$$\lambda = \frac{n}{\sum_{m=1}^{M} T_m V_m} = \frac{2}{0.035+0.035+0.084} = \frac{2}{0.154} = 12.99/\text{sec}$$

$\lambda_m = \lambda V_m;\ \lambda_1 = 12.99,\ \lambda_2 = 9.09,\ \lambda_3 = 2.60$

$n_1(2) = 0.455,\ n_2(2) = 0.455,\ n_3(2) = 1.092$

n\m	0	1	2	3
0	0	1	1	1=G(0)
1	0	1	2	4
2	0	1	3	11
3	0	1	4	26
4	0	1	5	57
5	0	1	6	120
6	0	1	7	247
7	0	1	8	502=G(7)

From convolution, $E[n_1]= u_1^1(G(1)/G(2)) +$ ______________ $= 4/11 + 1/11 = 0.455$

MVA procedure

n=3;

T_1= 0.028{1+0.455} = 0.041 sec

T_2= 0.058, T_3=0.586

$$\lambda = \frac{n}{\sum_{m=1}^{M} T_m V_m} = \frac{3}{0.041 + 0.041 + 0.117} = \frac{3}{0.199} = 15.1/\text{sec}$$

$\lambda_m = \lambda V_m$;

$\lambda_1 = 15.11$, $\lambda_2 = 10.6$, $\lambda_3 = 3.02$

$n_1(3) = 0.615$, $n_2(3) = 0.615$, $n_3(3) = 1.77$

Verification with convolution method

$$\rho_1 = \frac{\lambda_1}{\mu_1} = \frac{15.11}{\frac{1}{0.028}} = 0.423$$

$$\rho_1 = u_1 \frac{G(2)}{G(3)} = 1 * \frac{11}{26} = 0.423$$