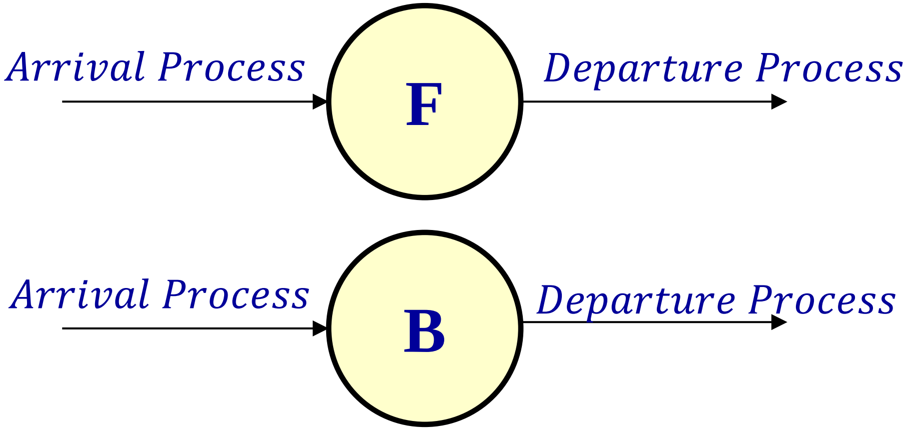
Queuing Networks

Wha Sook Jeon
Mobile Computing & Communications Lab.
Seoul National University

Time Reversibility (1)

- Time reversibility
  - Statistical characteristic of forward process is the same as that of backward process
  - The arrival process of the forward process is the arrival process of the backward process, which is the departure process of the forward process



arrival process $_F$ = arrival process $_B$ = departure process $_B$ = departure process $_F$

Time Reversibility (2)

- Forward process
  - Transition probability from state *i* to state *j*: P_{ij}
- Backward process
  - Transition probability from state *i* to state *j*: q_{ij}
- Time reversible: $q_{ij} = P_{ij}$
  - $q_{ij} = \Pr\{X_n = j | X_{n+1} = i\}$

$$= \frac{\Pr\{X_n = j, X_{n+1} = i\}}{\Pr\{X_{n+1} = i\}} = \frac{\Pr\{X_{n+1} = i | X_n = j\} \Pr\{X_n = j\}}{\Pr\{X_{n+1} = i\}}$$

$$= \frac{\pi_j P_{ji}}{\pi_i}$$

  - $\pi_i q_{ij} = \pi_j P_{ji}$
- When time reversibility is hold, $\pi_i q_{ij} = \pi_j P_{ji}$

Time Reversibility (3)

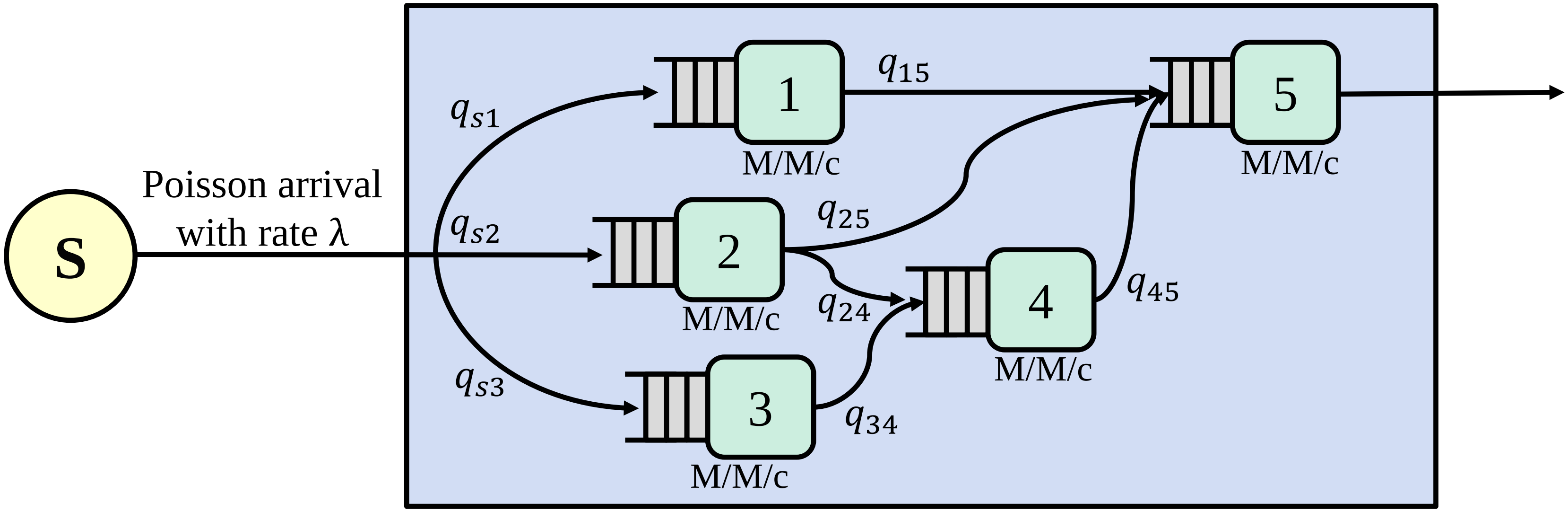
- Time reversible DTMC : $\pi_i q_{ij} = \pi_j P_{ji}$
- Time reversible CTMC : $\pi_i r_{ij} = \pi_j r_{ji}$
- Birth & death process is time reversible
  - Since M/M/c queueing system is a special case of birth & death process, M/M/c is time reversible
  - Arrival process of M/M/c queuing system is the same as its departure process. Thus, departure process of M/M/c is a Poisson process

Open Queueing Networks (1)

- Open Queueing networks with product form solution

  <Assumption>

  – Poisson arrivals from outside source
  – All servers have exponentially distributed service time
  – A job from device *i* joins device *j* with (routing) probability q_{ij}



Open Queueing Networks (2)

- System state: $(n_1, n_2, n_3, n_4, n_5)$
  - n_i: number of jobs in server *i*

- Jackson's decomposition theorem

$$P(n_1, n_2, n_3, n_4, n_5) = P_1(n_1)\, P_2(n_2) P_3(n_3)\, P_4(n_4) P_5(n_5)$$

  - $P(n_1, n_2, n_3, n_4, n_5)$: System state probability
  - $P_i(n_i)$: Probability of n_i jobs in server *i*

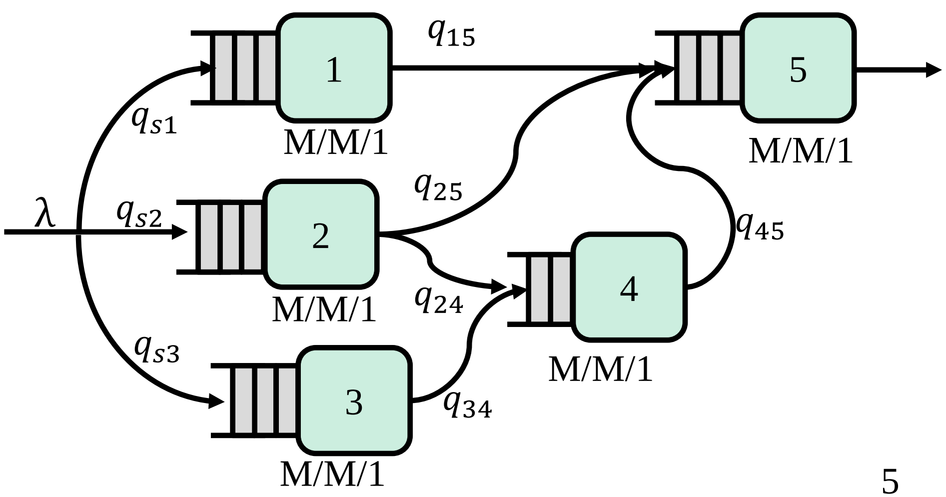
<example>

- When all devices are M/M/1

$P_i(n_i) = {\rho_i}^{n_i}(1 - \rho_i)$

$P(n_1, n_2, n_3, n_4, n_5) = \prod_{i=1}^{5} {\rho_i}^{n_i}(1 - \rho_i)$

  - $\rho_i = \frac{\lambda_i}{\mu_i}$
  - $\lambda_1 = \lambda q_{s1}, \quad \lambda_2 = \lambda q_{s2}\,, \quad \lambda_3 = \lambda q_{s3},$
  - $\lambda_4 = \lambda_3 + \lambda_2 q_{24}\,, \; \lambda_5 = \lambda_1 + \lambda_2 q_{25} + \lambda_4$



Open Queueing Networks (3)

- Performance measure

< Device i >

– Utilization of device i: $\rho_i = \frac{\lambda_i}{\mu_i}$

– Mean number of jobs in device i: $\bar{N}_i = \frac{\rho_i}{1-\rho_i}$

< System >

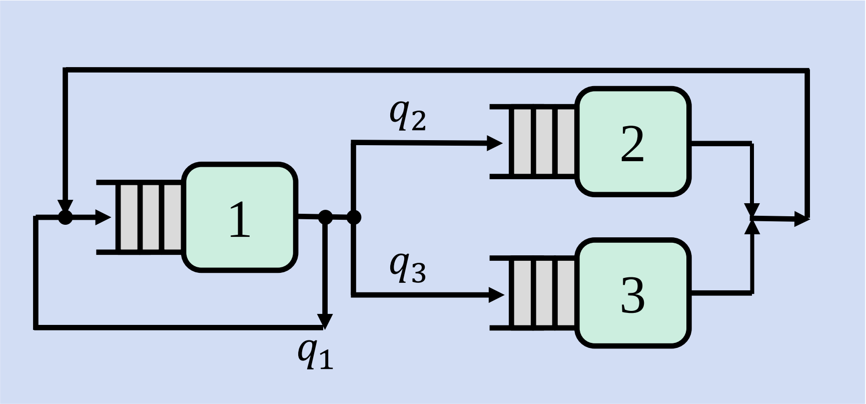
– Mean number of jobs: $\bar{N} = \sum_{i=1}^{M} \bar{N}_i$

  - M: the number of devices in the network

– Mean sojourn time of a job in the network: $\bar{T} = \frac{\bar{N}}{\lambda}$

Closed Queueing Networks (1)

- M: the number of devices in the network
- N: the total number of jobs in the network
  - **N is fixed** in the closed queueing networks



- System state: $(n_1, n_2, n_3, n_4, n_5)$
  - n_i: number of jobs in server *i*

Closed Queueing Networks (2)

- Assumptions for product form solution
  - The system is in steady state
  - All servers have exponentially distributed service time
  - Jobs are stochastically independent of each other
  - A job from device *i* joins device *j* with the (routing) probability q_{ij}
- Gordon and Newell's decomposition theorem

$$P(n_1, n_2, \ldots, n_M) = \frac{1}{G} F_1(n_1) F_2(n_2) \ldots F_M(n_M)$$

- $\sum_{\mathbb{N} \in S(M,N)} P(n_1, n_2, \ldots, n_M) = 1$
  - ✓ $\mathbb{N} = (n_1, n_2, \ldots, n_M)$
  - ✓ $S(M, N) = \{(n_1, n_2, \ldots, n_M) | n_1 + n_1 + \cdots + n_M = N\}$
- Normalization factor $G = \sum_{\mathbb{N} \in S(M,N)} \prod_{i=1}^{M} F_i(n_i)$

Closed Queueing Network (3)

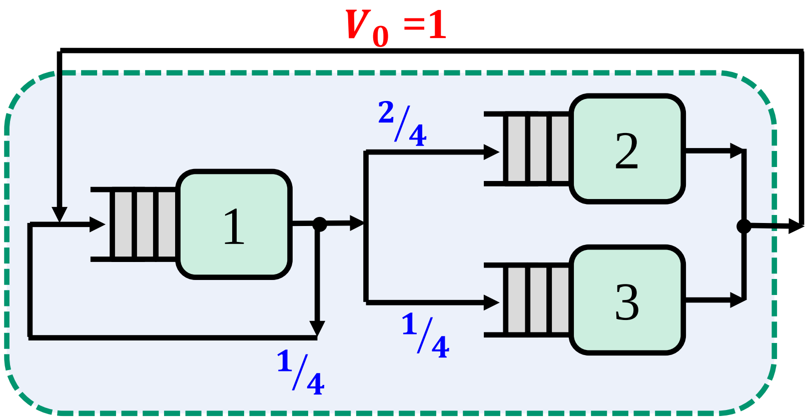
- $F_i(n_i) = \begin{cases} 1 & , \quad n_i = 0 \\ V_i S(n_i) F_i(n_i - 1) & , \quad n_i \geq 1 \end{cases}$
  - V_i : Visit ratio of device *i*
  - $S(n_i)$: the service time of device *i* when there are n_i jobs in device *i*
- Derivation of V_i,
  - For any appropriate link, V_0 =1.
  - Then, calculate other V_i values : $V_i = \sum_{j=0}^{M} V_j \, q_{ji}$

< Example >

$V_0 = 1$, $V_0 = V_2 + V_3$,

$V_1 = \frac{1}{4} V_1 + V_0$, $V_2 = \frac{2}{4} V_1$, $V_3 = \frac{1}{4} V_1$

$\Rightarrow \quad V_1 = \frac{4}{3}$, $V_2 = \frac{2}{3}$, $V_3 = \frac{1}{3}$



Closed Queueing Network (4)

- Buzen's Algorithm for calculating G
  - Let $g_m(n) \coloneqq \sum_{\mathbf{n}\in S(m,n)} \prod_{i=1}^{m} F_i(n_i)$

    where $\mathbf{n} = (n_1, n_2, \ldots, n_m)$, $S(M,N) = \{(n_1, n_2, \ldots, n_m) | n_1 + n_1 + \cdots + n_m = n\}$
  - $G = g_M(N)$
  - $g_1(n) = F_1(n)$
  - $g_m(0) = \prod_{i=1}^{m} F_i(0) = 1$
  - $g_m(n) = \sum_{k=0}^{n} F_m(k) \sum_{(n_1 \ldots n_{m-1})\in S(m-1,n-k)} \prod_{i=1}^{m-1} F_i(n_i),\quad (n > 0, m > 1)$

    $= \sum_{k=0}^{n} F_m(k) g_{m-1}(n-k)$

    $g_m(n)$ can be calculated in a recursive fashion

Closed Queueing Network (5)

Calculation of $g_m(n)$

	1	2		m-1	m		M	
0	1	1	…	1 $\times F_m(n)$	1	…	1	
1	$F_1(1)$	$g_2(1)$	…	+ $g_{m-1}(1) \times F_m(n)$				
⋮	⋮	⋮		⋮				
n-1	$F_1(n\text{-}1)$	$g_2(n\text{-}1)$	…	+ $g_{m-1}(n\text{-}1) \times F_m(1)$				
n	$F_1(n)$	$g_2(n)$	…	+ $g_{m-1}(n) \times F_m(0)$	$= g_m(n)$			
⋮	⋮	⋮					$g_M(N\text{-}1)$	
N	$F_1(N)$	$g_2(N)$					$g_M(N)$	$= \boldsymbol{G}$

Closed Queueing Network (6)

- When the service rate of each device is constant (a single server)
  - $S(n_i) = S_i \,, \forall n_i \geq 1 \quad \Rightarrow \quad F_m(k) = V_m S_m F_m(k-1)$
- $g_m(n) = F_m(0) g_{m-1}(n) + \sum_{k=1}^{n} F_m(k) g_{m-1}(n-k)$

$$= g_{m-1}(n) + V_m S_m \sum_{k=1}^{n} F_m(k-1) g_{m-1}(n-k)$$

$$= g_{m-1}(n) + V_m S_m g_{m-1}(n-1)$$

	1	2		m-1	m		M
0	1	1	…	1	1	…	1
1	$F_1(1)$	$g_2(1)$	…	$g_{m-1}(1)$	$g_m(1)$		
⋮	⋮	⋮		⋮			
n-1	$F_1(n\text{-}1)$	$g_2(n\text{-}1)$	…	$g_{m-1}(n\text{-}1)$	$g_m(n\text{-}1)$ × $V_m S_m$		
n	$F_1(n)$	$g_2(n)$	…	$g_{m-1}(n)$	+ = $g_m(n)$		
⋮	⋮	⋮					
N	$F_1(N)$	$g_2(N)$					

Closed Queueing Network (7)

- Performance measure
  - Throughput of device M: X_M
    - $X_M = \sum_{k=1}^{N} P_M(k) \frac{1}{S_M(k)}$
      - ✓ $P_M(k)$: Probability that there are k jobs in the device M
      - ✓ $P_M(k) = \sum_{(n_1,n_2,\dots,n_{M-1}) \in S(M-1,N-k)} \frac{1}{G} F_1(n_1) \dots F_{M-1}(n_{M-1}) F_M(k)$

$$= \frac{1}{G} F_M(k) g_{M-1}(N-k)$$

$$\Rightarrow \quad X_M = \sum_{k=1}^{N} \frac{1}{G} F_M(k) g_{M-1}(N-k) \frac{1}{S_M(k)}$$

$$= \sum_{k=1}^{N} \frac{1}{G} V_M S_M(k) F_M(k-1)\, g_{M-1}(N-k) \frac{1}{S_M(k)}$$

$$= \frac{1}{G} V_M \; g_M(N-1)$$

Closed Queueing Network (8)

Since $\frac{X_i}{X_j} = \frac{V_i}{V_j}$ for any device *i*, *j*

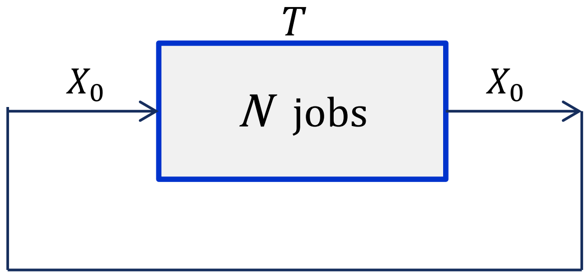
– System Throughput : X_0

$$X_0 = \frac{X_M}{V_M} = \frac{g_M(N-1)}{G}$$

– Throughput of arbitrary device *i*

$$X_i = V_i X_0$$

– System response time: T



By Little's Law, $T = \frac{N}{X_0}$