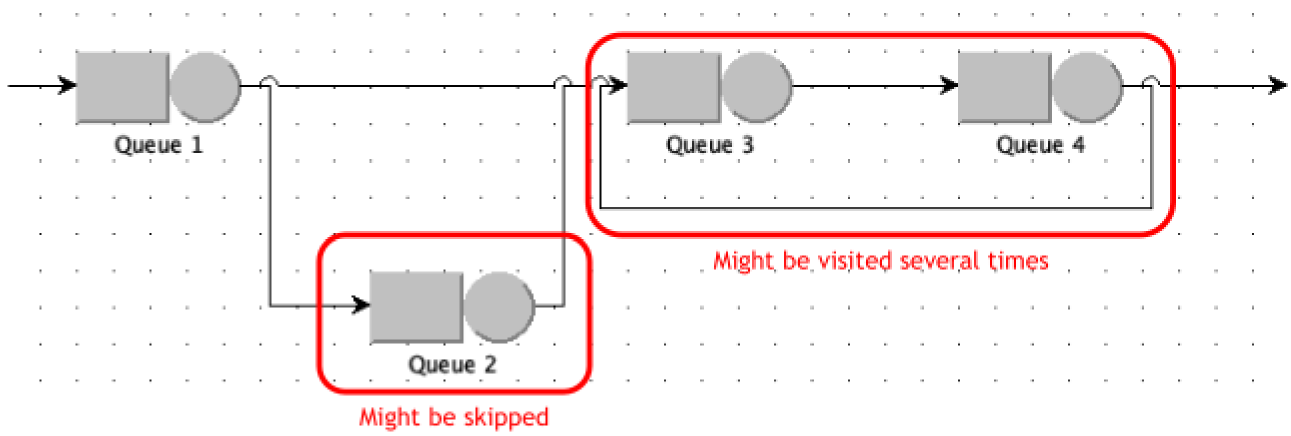
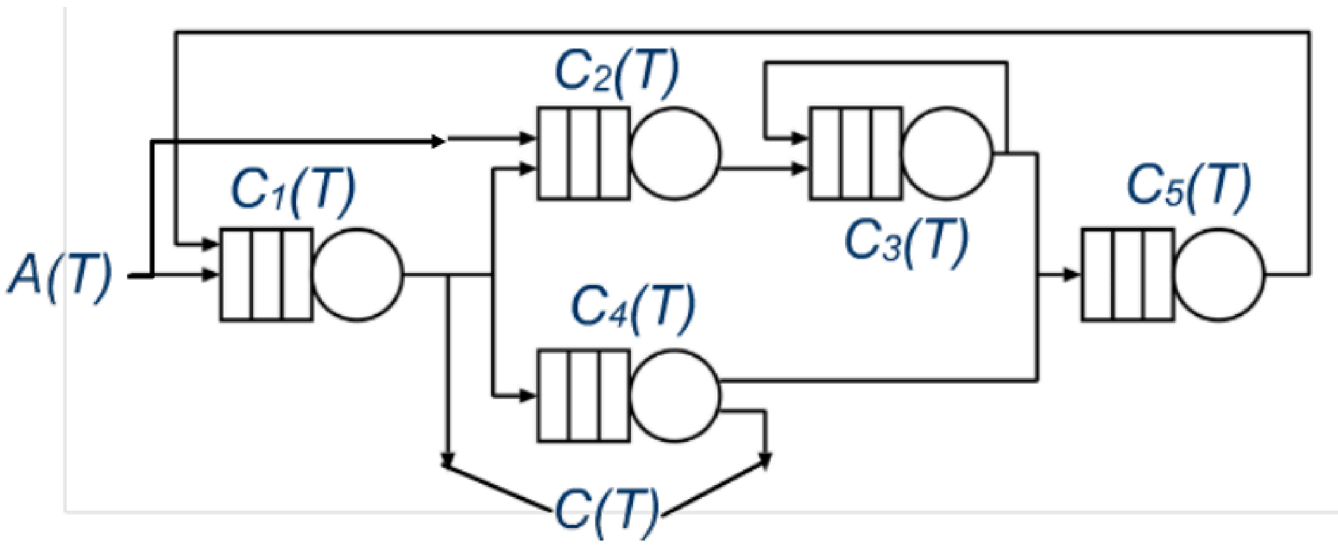
18.Open models

Visits and demands

Systems with multiple stations. Need to routing the jobs through the multiple stations.



Some stations are always visited and others can be skipped.
The arrival rate of a station that receives all the jobs is equal to the arrival rate of the initial station, meanwhile the arrival rate of a station where some jobs don't arrive the arrival rate would be less than the initial one.
We describe as visits the average number of jobs that enter a station of the system.
The visits v_k represents the average number of times a job pass through station k from the moment it enters the system to the time it leaves.
Let us call $C_k(T)$ the number of completions of jobs at station k, and C(T) the number of completions of jobs in the system (i.e. number of jobs leaving the network).



Visits vk can be defined as:

$$v_k = \lim_{T\to\infty} \frac{C_k(T)}{C(T)}$$

- Can be >1 when a customer pass through the station more than one time in the same loop
- Can be =1 if every job will pass in each station only once. Can also be 1 even if the station is skipped/looped more than one(average need to be one)
- Can be <1 if the station is not visited when a job enters the system

Service Demand

The Service Demand D_k of a station k, accounts for the average time spent by a job in the considered service center during all its visits, taking into account the fact that such resource might also be skipped. If we call $B_k(T)$ the busy time of station k after a time T, C(T) the completion of the system, it can be defined as:

$$D_k = \lim_{T\to\infty} \frac{B_k(T)}{C(T)} = \lim_{T\to\infty} \frac{C_k(T)}{C(T)} \frac{B_k(T)}{C_k(T)} = v_k S_k$$

Service time and demand are identical if we have a single station
Note that:

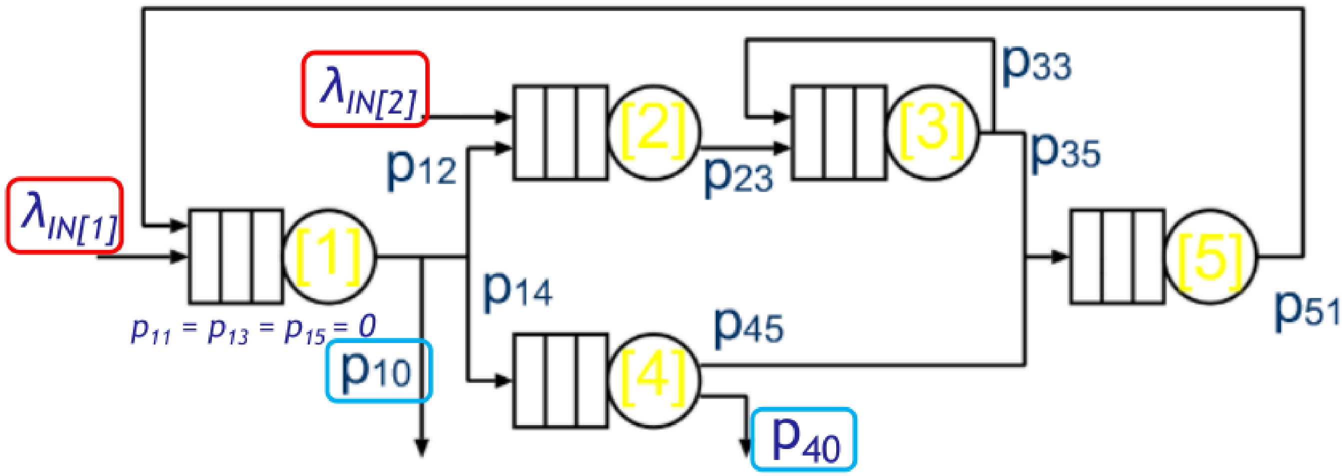
- Average service time S_k accounts for the average time that a job spends during a service in station k when IT IS SERVED.
- Average service demand D_k accounts for the average time a job spends during service in station k for a complete stay in the system – including the fact that it might be visited several times, or that it might not be visited at all.
  As seen for the visits, depending on the way in which the jobs move in the system, the demand can be less than, greater than or equal to the average service time of station k.
  In particular, one of the most important relations that can be given with the definition of visits is the Forced Flow law. It relates the throughput of one station, with the one of the system.

  $$\text{The Forced Flow Law: } X_k = V_k X$$

  Since:

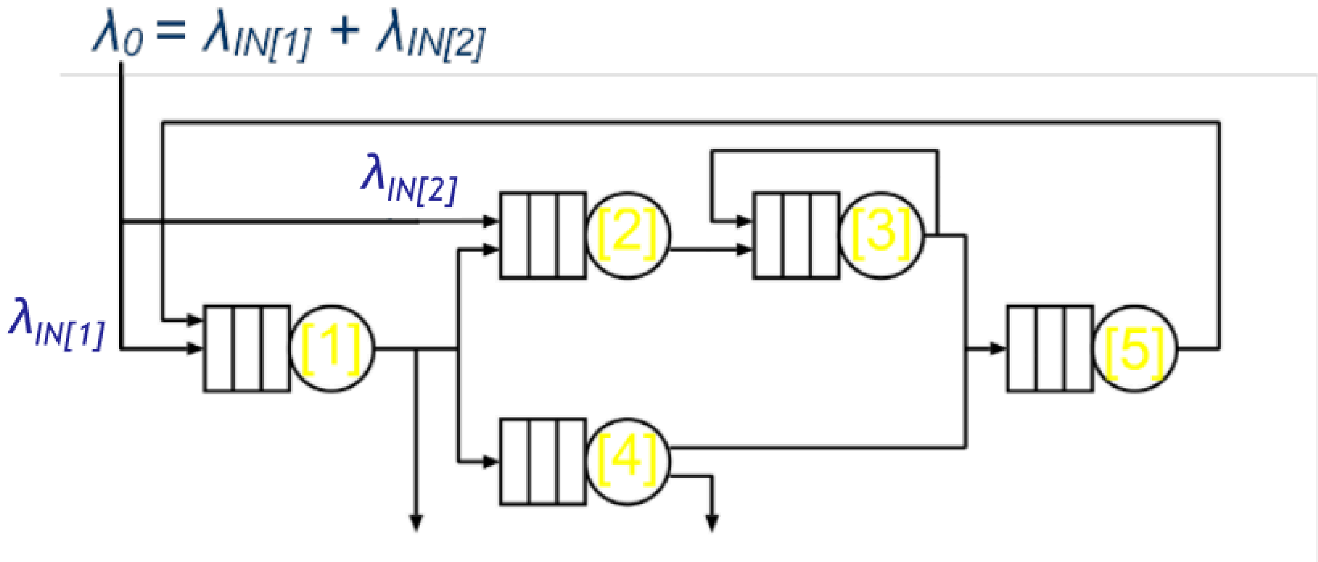
  $$X_k = \lim_{T\to\infty} \frac{C_k(T)}{T} = \lim_{T\to\infty} \frac{C_k(T)}{C(T)} \frac{C_k(T)}{T} = v_k X$$

  For probabilistic routing, it is possible to computed the visits to reach station starting from the routing probabilities. Let us call p_{ij} the probability that a job, which finishes its service at node i, choses node j as its next destination. If the considered route is not possible, we set p_{ij} = 0.
  The way in which visits are computed is different in open and closed models. Let's start focusing on open models. We call p_{i0} the probability that a job leaves the system after finishing its service at station i. Let us also call $\lambda_{IN[k]}$ the arrival rate at station k.



We sum up all the arrivals to the system in a single source with global speed λ_0:

$$\lambda_0 = \sum_{k=1}^{K} \lambda_{IN[k]} = X$$



The jobs that enter the systems need to be equal to the jobs that exit the system. Otherwise we are loosing something
Recalling the Forced Flow Law and the definition of throughput and arrival rate we can define:

$$v_k = \frac{X_k}{X} = \frac{\lambda_k}{\lambda_0}$$

We can then determine the arrival rate λ_k to all the stations k by solving the following linear system of equations:

$$\begin{cases} nbsp; \quad nbsp; \qquad nbsp; \lambda_k = \lambda_{IN[k]} + \sum_{i=1}^{K} X_i p_{ik} \\ nbsp; \quad nbsp; \ldots \end{cases}$$

The term λ_k on the left hand side of the equations accounts for the jobs that arrive to a station k.
It is equal to the sum of the jobs that arrives from outside $\lambda_{IN[k]}$ plus the jobs that exits from

every other station $i(X_i)$ that are routed to the considered station k (p_{ik}). Note that the summation includes also index k to allow self-loops.
Since the system is stable the throughput is equal to the arrival rate
If we are not interesting in the arrival rates (or throughputs) of the single stations, we can simplify the equation dividing each term by λ_0 and obtain visits directly:

$$\begin{cases} \dfrac{\lambda_k}{\lambda_o} = \dfrac{\lambda_{IN[k]}}{\lambda_o} + \displaystyle\sum_{i=1}^{K} \dfrac{\lambda_i}{\lambda_o} \cdot p_{ik} \\ \dots \end{cases}$$

$$v_k = \frac{\lambda_k}{\lambda_0}$$

$$\begin{cases} v_k = \dfrac{\lambda_{IN[k]}}{\lambda_o} + \displaystyle\sum_{i=1}^{K} v_i \cdot p_{ik} \\ \dots \end{cases}$$

Let us call $l_k = \lambda_{IN[k]}/\lambda_0$. We then have:

$$\begin{cases} v_k = l_k + \displaystyle\sum_{i=1}^{K} v_i \cdot p_{ik} \\ \dots \end{cases}$$

$$\sum_{i}^{K} l_i = 1$$

Note also that, in most cases $\lambda_{IN[k]}$ is different from zero only for one station k, which means that in such cases we have $\lambda_{IN[k]}/\lambda_0 = l_k = 1$ and $\lambda_{IN[j]}/\lambda_0 = l_j = 0$ for all j different from k.

$$\left\{ v_k = l_k + \sum_{i=1}^{K} v_i \cdot p_{ik} \right. \qquad \left\{ \begin{array}{ll} v_k = 1 + \sum_{i=1}^{K} v_i \cdot p_{ik} & \\ \dots & \\ v_j = \sum_{i=1}^{K} v_i \cdot p_{ik} & j \neq k \\ \dots & \end{array} \right.$$

Valid only when we have a single input to the system, which makes all the jobs enter into station k.

Moreover, to simplify the computation in mathematical packages, the previous equation can be written in matrix form:

$$\boldsymbol{l} = |\cdots l_k \cdots| \quad \boldsymbol{v} = |\cdots v_k \cdots| \quad \boldsymbol{P} = |\cdots p_{ij} \cdots|$$

$$\boldsymbol{v} = \boldsymbol{l} + \boldsymbol{v} \cdot \boldsymbol{P} \qquad \boldsymbol{v} = \boldsymbol{l} \cdot (\boldsymbol{I} - \boldsymbol{P})^{-1}$$

Response and Residence times

The response time of one station is the average time spent in a system in a station when a job is in a queue.
Residence time is the total time a job remain in a system. The residence time is equal to the visit per the response time
The utilization law and the Little's law become:

$$Uk = XD_k = Xv_kSk = X_kS_k$$

(Demand correspond to throughput of the system, the service time correspond to the throughput of the station, same thing for the number of the of jobs)

$$N_k = XR_k = Xv_k\phi_k = X_k\phi_k$$

Performance of an open model

For separable models we can compute the models performance indexes in a simple way, but it is difficult to separate the system. Inter-arrival times are exponentially distributed, infinite capacity, FCFS and infinite servers.
If we have a delay station the waiting time will be zero and the response time will be equal to the response time
For the exponential distribution the small value of w are distributed exactly as the initial distribution
For delay stations we have $R_k = D_k$
The average number of jobs in a node can be computed by having the utilization of the system.
For the single station we have the same formula of M/M/1

Network performance indices

A system is stable only if all its stations are stable, ii its arrival rate is less than the inverse of the demand of each station

Total average population of the system is the sum of the population of all the stations

The system response time is the sum of the residence time of each stations!!!

Little's law continue to be valid system wide but for each station we need to calculate the average jobs in the system accordingly to the situation