Lecture 7

Burke's Theorem and Networks of Queues

Eytan Modiano
Massachusetts Institute of Technology

Burke's Theorem

- An interesting property of an M/M/1 queue, which greatly simplifies combining these queues into a network, is the surprising fact that the output of an M/M/1 queue with arrival rate λ is a Poisson process of rate λ
  – This is part of Burke's theorem, which follows from reversibility

- A Markov chain has the property that
  – P[future | present, past] = P[future | present]

    Conditional on the present state, future states and past states are independent

    P[past | present, future] = P[past | present]

    $\Rightarrow P[X_n=j \mid X_{n+1}=i, X_{n+2}=i_2,\ldots] = P[X_n=j \mid X_{n+1}=i] = P^*_{ij}$

Burke's Theorem (continued)

- The state sequence, run backward in time, in steady state, is a Markov chain again and it can be easily shown that

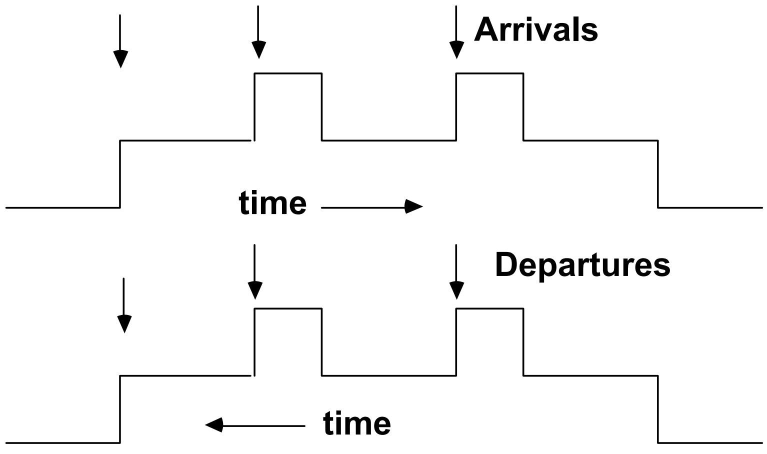
$$p_i P^*_{ij} = p_j P_{ji} \qquad \text{(e.g., M/M/1} \quad (p_n)\lambda=(p_{n+1})\mu)$$

- A Markov chain is <u>reversible</u> if P*ij = Pij
  – Forward transition probabilities are the same as the backward probabilities
  – If reversible, a sequence of states run backwards in time is statistically indistinguishable from a sequence run forward

- A chain is reversible iff $p_i P_{ij} = p_j P_{ji}$

- All birth/death processes are reversible
  – Detailed balance equations must be satisfied

Implications of Burke's Theorem

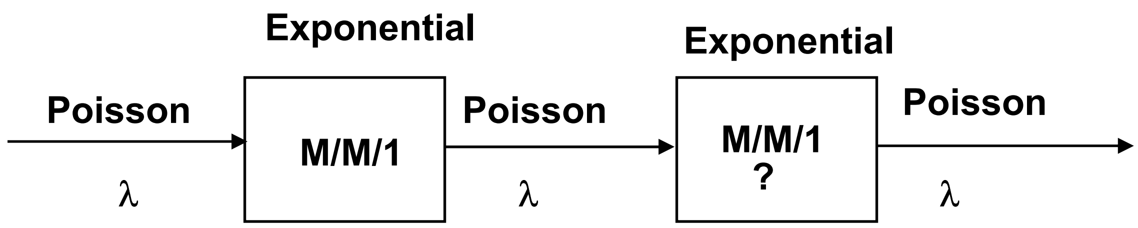


- **Since the arrivals in forward time form a Poisson process, the departures in backward time form a Poisson process**

- **Since the backward process is statistically the same as the forward process, the (forward) departure process is Poisson**

- **By the same type of argument, the state (packets in system) left by a (forward) departure is independent of the past departures**
  - **In backward process the state is independent of future arrivals**

NETWORKS OF QUEUES

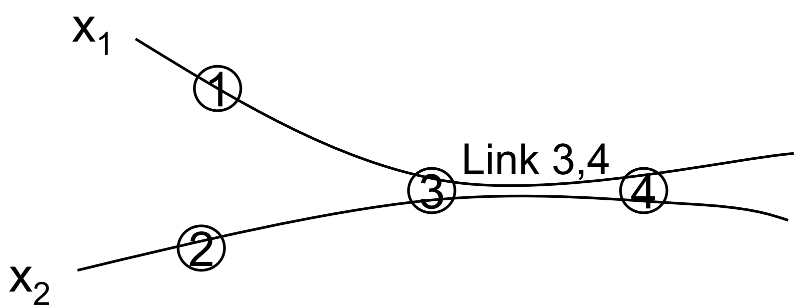


- The output process from an M/M/1 queue is a Poisson process of the same rate λ as the input

- Is the second queue M/M/1?

Independence Approximation (Kleinrock)

- **Assume that service times are independent from queue to queue**
  - **Not a realistic assumption: the service time of a packet is determined by its length, which doesn't change from queue to queue**



- **X_p = arrival rate of packets along path p**

- **Let λ_{ij} = arrival rate of packets to link (i,j)**

$$\lambda_{ij} = \sum_{\text{P traverses link (i, j)}} X_p$$

- **μ_{ij} = service rate on link (i,j)**

Kleinrock approximation

- **Assume all queues behave as independent M/M/1 queues**

$$N_{ij} = \frac{\lambda_{ij}}{\mu_{ij} - \lambda_{ij}}$$

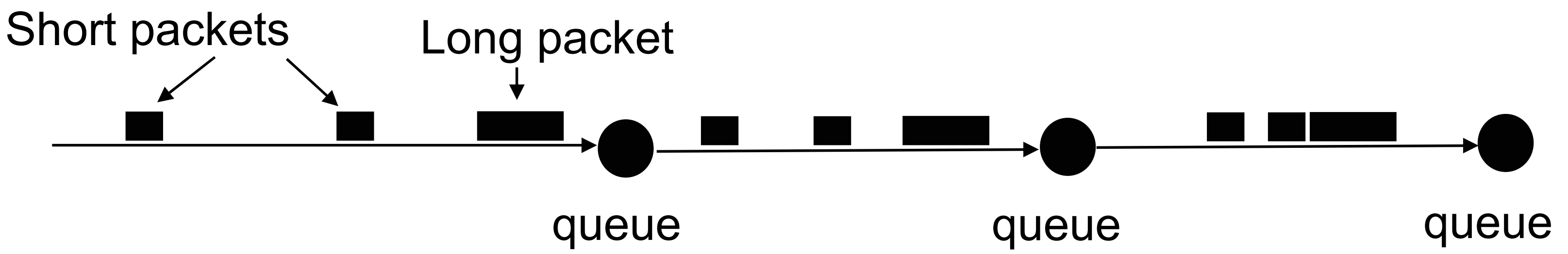
- **N = Ave. packets in network, T = Ave. packet delay in network**

$$N = \sum_{i,j} N_{ij} = \frac{\lambda_{ij}}{\mu_{ij} - \lambda_{ij}}, \qquad T = \frac{N}{\lambda}$$

$$\lambda = \sum_{all\ paths\ p} X_P = \text{ total external arrival rate}$$

- **Approximation is not always good, but is useful when accuracy of prediction is not critical**
  - **Relative performance but not actual performance matters**
  - **E.g., topology design**

Slow truck effect



- **Example of bunching from slow truck effect**
  - **long packets require long service at each node**
  - **Shorter packets catch up with the long packets**

- **Similar to phenomenon that we experience on the roads**
  - **Slow car is followed by many faster cars because they catch up with it**

Jackson Networks

- **Independent external Poisson arrivals**
- **Independent Exponential service times**
  - **Same job has independent service time at different queues**
- **Independent routing of packets**
  - **When a packet leaves node i it goes to node j with probability P_{ij}**
  - **Packet leaves system with probability** $1 - \sum_j P_{ij}$
  - **Packets can loop inside network**

- **Arrival rate at node i,**

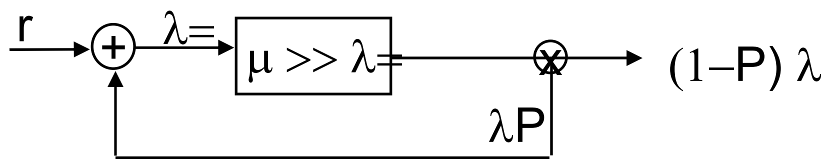
$$\lambda_i = r_i + \sum_k \lambda_k P_{ki}$$

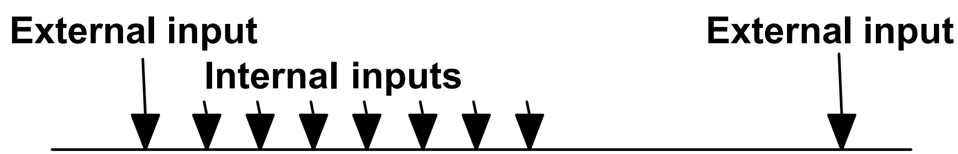
External arrivals (r_i)

Internal arrivals from Other nodes ($\sum_k \lambda_k P_{ki}$)

  - **Set of equations can be solve to obtain unique λ_i's**
  - **Service rate at node i = μ_i**

Jackson Network (continued)





- **Customers are processed fast** $(\mu >> \lambda)$
- **Customers exit with probability (1-P)**
  - **Customers return to queue with probability P**
  - $\lambda = r + P\lambda \Rightarrow \lambda = r/(1-P)$
- **When P is large, each external arrival is followed by a burst of internal arrivals**
  - **Arrivals to queues are not Poisson**

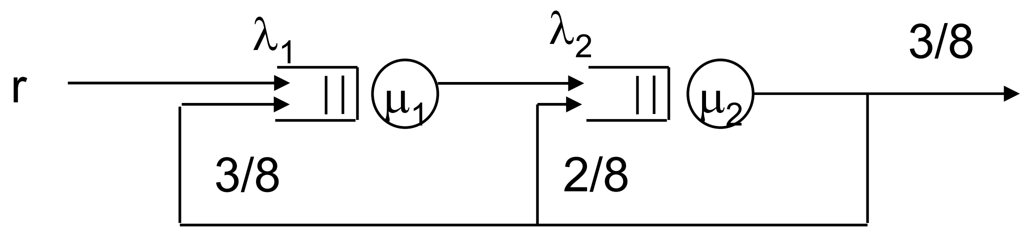
Jackson's Theorem

- **We define the state of the system to be** $\vec{n} = (n_1, n_2 \ldots n_k)$
  where n_i **is the number of customers at node i**
- **Jackson's theorem:**

$$P(\vec{n}) = \prod_{i=1}^{i=k} P_i(n_i) = \prod_{i=1}^{i=k} \rho_i^{n_i}(1-\rho_i), \qquad where\ \rho_i = \frac{\lambda_i}{\mu_i}$$

- **That is, in steady state the state of node i (n_i) is independent of the states of all other nodes (at a given time)**
  - **Independent M/M/1 queues**
  - **Surprising result given that arrivals to each queue are neither Poisson nor independent**
  - **Similar to Kleinrock's independence approximation**
  - **Reversibility**
    - **Exogenous outputs are independent and Poisson**
    - **The state of the entire system is independent of past exogenous departures**

Example



$\lambda_1 = ?$
$\lambda_2 = ?$

$P(n_1,n_2) = ?$