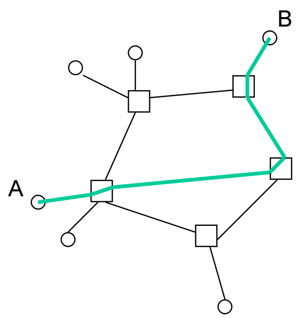
10. Network models

Contents

- Circuit switched network modelled as a loss network
- Packet switched network modelled as a queueing network

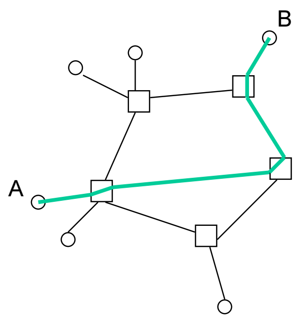
Teletraffic model of a circuit switched network (1)

- Consider a circuit switched network
  - e.g. a telephone network
- Traffic:
  - telephone calls
  - each (carried) call occupies one channel on each link among its route
- System:
  - telephone machines (terminals)
  - exchanges (network nodes)
  - access links (from terminals to exchanges)
  - trunks (between exchanges)



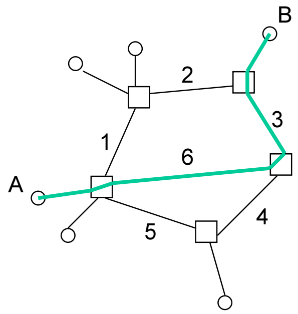
Teletraffic model of a circuit switched network (2)

- Quality of service:
  - described by the **end-to-end call blocking probability** (prob. that a desired connection cannot be set up due to congestion along the route of the connection)
- In our model we assume that
  - the network nodes and the whole access network are non-blocking
- Thus, a call is blocked
  - if and only if all channels are occupied in any trunk network link along the route of that call



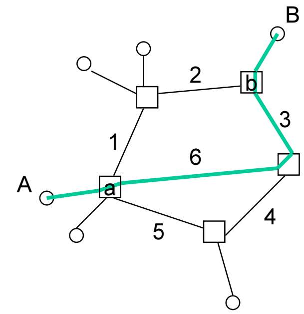
Links j = 1,…,J

- In our model,
  - all links are **two-way** (why?)
- We index the links in the trunk network by
  - $j = 1,\dots,J$
  - example on the right: $J = 6$
- Let n_j denote the number of channels in link j (that is: the link capacity)
  - $\mathbf{n} = (n_1,\dots,n_J)$
- Each link is modelled as a
  - **pure loss system**



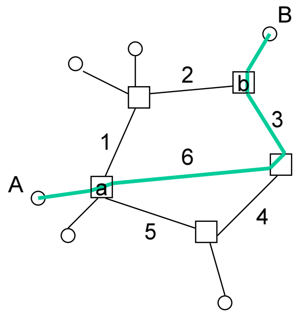
Routes r = 1,…,R

- We define a **route** as a
  - set of consecutive (two-way) links connecting two network nodes
- We index the routes by
  - $r = 1,\dots,R$
- In the example on the right:
  - $R = 12 + 10 + 7 + 3 = 32$
  - there are three routes **between** nodes a and b: {1,2}, {6,3}, {5,4,3}
- Let $d_{jr} = 1$ if link j belongs to route r (otherwise $d_{jr} = 0$)
  - $\mathbf{D} = (d_{jr} \mid j = 1,\dots,J;\, r = 1,\dots,R)$



Traffic classes

- Note:
  - End-to-end call blocking prob. is equal for all the connections following the same route
- Thus the **traffic class** of a connection is determined by the route r the connection follows
  - Example on the right: connection between A and B belongs to class using route {6,3}
- Let x_r denote the number of active connections following route r
  - $\mathbf{x} = (x_1,\dots,x_R)$
- Vector $\mathbf{x}$ is called the **state** of the system



State space

- The number of active connections x_r for any traffic class r is limited by the link capacities n_j along the corresponding route r :

$$\sum_{r=1}^{R} d_{jr} x_r \le n_j \quad \text{for all } j$$

- The same in vector form:

$$\mathbf{D} \cdot \mathbf{x} \le \mathbf{n}$$

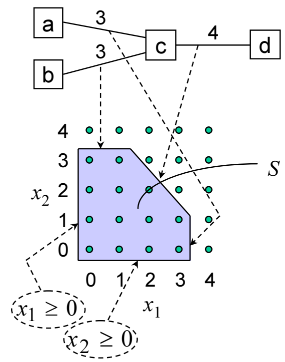
- Thus, the **state space** S (that is: the set of **admissible** states) is

$$S = \{\mathbf{x} \ge 0 \mid \mathbf{D} \cdot \mathbf{x} \le \mathbf{n}\}$$

  - Note that, due to finite link capacities, set S is **finite**

Example

- 3 links with capacities:
  - link a-c: 3 channels
  - link b-c: 3 channels
  - link c-d: 4 channels
- 2 routes:
  - route a-c-d
  - route b-c-d
  - The other 4 routes (which?) are ignored in this model
- State space:
  - S = {(0,0),(0,1),(0,2),(0,3), (1,0),(1,1),(1,2),(1,3), (2,0),(2,1),(2,2), (3,0),(3,1)}



Set S_r of non-blocking states for class r

- Consider
  - an arriving call belonging to class r (that is: following route r)
- It will **not** be blocked by link j belonging to route r
  - if there is at least one free channel on link j:

$$\sum_{r'=1}^{R} d_{jr'} x_{r'} \leq n_j - 1 \quad \text{for all } j \in r$$

- The same in vector form ($\mathbf{e}_r$ being here the unit vector in direction r):

$$\mathbf{D} \cdot (\mathbf{x} + \mathbf{e}_r) \leq \mathbf{n}$$

- The set S_r of **non-blocking** states for class r is thus

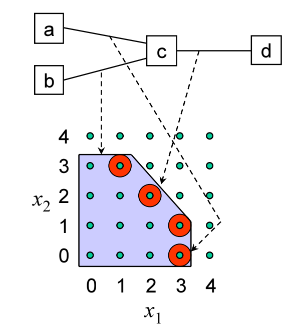
$$S_r = \{\mathbf{x} \geq 0 \mid \mathbf{D} \cdot (\mathbf{x} + \mathbf{e}_r) \leq \mathbf{n}\}$$

Set S_r^B of blocking states for class r

- The set S_r^B of **blocking** states for class r is clearly:

$$S_r^B = S \setminus S_r$$

- Summary:
  - an arriving call of class r is blocked (and lost) if and only if the state x of the system belongs to set S_r^B
- Example (continued):
  - The blocking states S_1^B for connections of class 1 (using route a-c-d) are circulated in the figure
  - S_1^B = { (1,3),(2,2),(3,0),(3,1)}



Loss network

- Assume that
  - new connection requests belonging to traffic class r arrive (independently) according to a Poisson process with intensity λ_r
  - call holding times independently and identically distributed with mean h
- Denote
  - $a_r = \lambda_r h$ (traffic intensity for class r)

Equilibrium distribution (1)

- Then it is possible to show that
  - the stationary state probability $\pi(\mathbf{x})$ for any state $\mathbf{x} \in S$ is as follows:

$$\pi(\mathbf{x}) = G^{-1} \cdot \prod_{r=1}^{R} f_r(x_r)$$

where G is a normalizing constant:

$$G = \sum_{\mathbf{x} \in S} \prod_{r=1}^{R} f_r(x_r)$$

and the functions $f_r(x_r)$ are defined as follows:

$$f_r(x_r) = \frac{a_r^{x_r}}{x_r!}$$

Equilibrium distribution (2)

- Probability $\pi(\mathbf{x})$ is said to be of **product-form**
  - However, the number of active connections of different classes are **not** independent (since the normalizing constant G depends on each x_r)
  - Only if all the links had infinite capacities, all the traffic classes would be independent of each other
  - Thus, it is the limited resources shared by the traffic classes that makes them dependent on each other

PASTA

- Consider, for a while,
  - any simple teletraffic model with Poisson arrivals
- According to so called **PASTA** (Poisson Arrivals See Time Averages) property,
  - arriving calls (obeying a Poisson process) see the system in equilibrium
- This is an important observation
  - applicable in many problems
- For example,
  - it allows us to calculate the end-to-end blocking probabilities in our circuit switched network model (since we assumed that new calls arrive according to a Poisson process)

End-to-end blocking: exact formula

- The probability that the system is in a state such that it cannot accept any more connections of type r is clearly given by the sum

$$\sum_{\mathbf{x} \in S_r^B} \pi(\mathbf{x})$$

  - Call this the end-to-end **time blocking** probability for class r
- Due to the PASTA property,
  - the end-to-end **call blocking** probability B_r equals this:

$$B_r = \sum_{\mathbf{x} \in S_r^B} \pi(\mathbf{x})$$

- Since there is no difference between time and call blocking in this case, we may briefly call it **end-to-end blocking**.

Example

- Consider the example presented in slide 9 (and continued in slide 11)
- The end-to-end blocking probability B_1 for class 1 will be

$$B_1 = \pi(1,3) + \pi(2,2) + \pi(3,0) + \pi(3,1) = \frac{\frac{a_1^1 a_2^3}{1!3!} + \frac{a_1^2 a_2^2}{2!2!} + \frac{a_1^3}{3!}\left(1 + \frac{a_2^1}{1!}\right)}{\left(1 + \frac{a_2^1}{1!} + \frac{a_2^2}{2!} + \frac{a_2^3}{3!}\right) + \frac{a_1^1}{1!}\left(1 + \frac{a_2^1}{1!} + \frac{a_2^2}{2!} + \frac{a_2^3}{3!}\right) + \frac{a_1^2}{2!}\left(1 + \frac{a_2^1}{1!} + \frac{a_2^2}{2!}\right) + \frac{a_1^3}{3!}\left(1 + \frac{a_2^1}{1!}\right)}$$

Approximative methods

- In practice,
  - it is extremely hard (even impossible) to apply the exact formula
  - This is due to the so called **state space explosion**:
    there are as many **dimensions** in the state spaces as
    there are routes in our model
    $\Rightarrow$ exponential growth of the state space
- Thus, **approximative** methods are needed
  - Below we will present (the simplest) one of them: product bound
- **Product Bound** method
  - estimate first blocking probabilities in each separate link
    (common to all traffic classes)
  - calculate then the end-to-end blocking probabilities for each class
    based on the hypothesis that "blocking occurs independently in each link"

Product Bound (1)

- Consider first the blocking probability $B(j)$ in an arbitrary link j
  - Let $R(j)$ denote the set of routes that use link j
- If the capacities of all the other links (but j) were infinite,
  - link j could be modelled as a loss system where new calls arrive according to a Poisson process with intensity $\lambda(j)$,

$$\lambda(j) = \sum_{r \in R(j)} \lambda_r$$

  - In this case, the blocking probability could be calculated from formula

$$B(j) \approx \mathrm{Erl}(n_j, \sum_{r \in R(j)} a_r)$$

  - Note that this is really an approximation, since the traffic offered to link j is smaller due to blockings in other links (and not even of Poisson type).

Product Bound (2)

- Consider then the **end-to-end blocking** probability B_r for class r
  - Let $J(r)$ denote the set of the links that belong to route r
  - Note that an arriving call of class r will not be blocked,
    if it is not blocked in any link $j \in J(r)$
- If blocking occured independently in each link,
  - an arriving call of class r would be blocked with probability

$$B_r \approx 1 - \prod_{j \in J(r)} (1 - B(j))$$

  - Note that for small values of $B(j)$'s, we can use the following approximation:

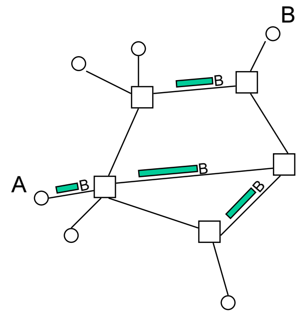
$$B_r \approx \sum_{j \in J(r)} B(j)$$

Contents

- Circuit switched network modelled as a loss network
- Packet switched network modelled as a queueing network

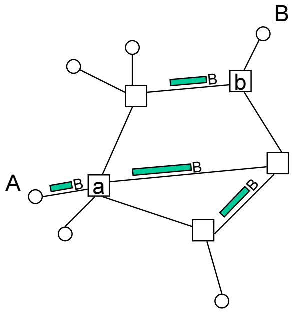
Teletraffic model of a packet switched network (1)

- Consider a connectionless packet switched network **at packet level**
  - e.g. an Internet subnetwork
- Traffic:
  - data packets
  - identified by their source (A) and destination (B)
- System:
  - workstations & servers (terminals)
  - routers (network nodes)
  - access links (from terminals to routers)
  - trunks (between routers)



Teletraffic model of a packet switched network (2)

- Quality of service:
  - described by the **average end-to-end packet delay** (the mean time for a packet to get from the source (A) to the destination (B))
- However, in our model
  - we restrict ourselves to the **average trunk network delay** (the mean time for a packet to get from the source router (a) to the destination router (b))
  - implicitly, we assume that the delay due to access network is negligible (or, at least, almost deterministic)

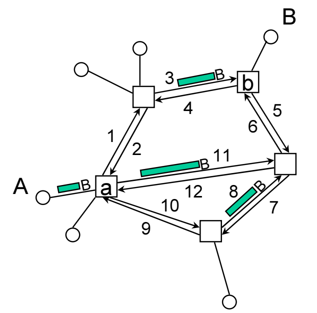


End-to-end delay components

- Trunk network delay consists of
  - **propagation delays** (in links)
  - **transmission delays** (in links)
  - **processing delays** (in nodes)
  - **queueing delays** (before transmission and before processing)
- Note that
  - propagation and transmission delays are deterministic,
  - processing delays might be random, and
  - queueing delays are surely random
- In our model,
  - we will take into account the transmission and the related queueing delays
  - but we will ignore the propagation delays in links and the delays in nodes (the processing and the related queueing delays)

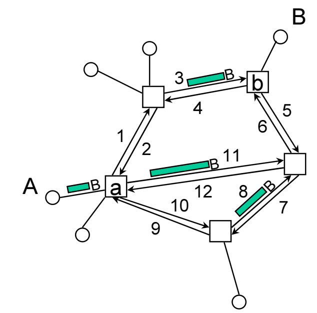
Links j = 1,…,J

- In this case we separate the directions so that
  - all links are **one-way** (why?)
- We index the links in the trunk network by
  - $j = 1,\ldots,J$
  - example on the right: $J = 12$
- Let C_j denote the capacity of link j (in bps)



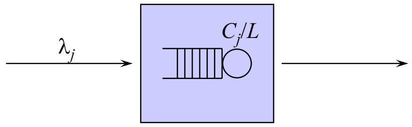
Routes r = 1,…,R

- We define here a **route** as an
  - ordered set of consecutive (one-way) links connecting two network nodes (called origin and destination)
- We index the routes by
  - $r = 1,\ldots,R$
- In the example on the right:
  - $R = 2*(12+10+7+3) = 64$
  - there are three routes **from** node a **to** node b: (1,3), (11,6), (10,8,6)
  - for these routes, node a is the origin and node b is the destination



Individual link model

- Each link is modelled as a
  - **pure waiting system** (with a single server and an infinite buffer)
- Let
  - λ_j = arrival rate of packets to be transmitted on link j (in packets/s)
  - L = mean packet length (in bits)
  - $1/\mu_j = L/C_j$ = average packet transmission time on link j (in seconds)
- Stability requirement: $\lambda_j < \mu_j$



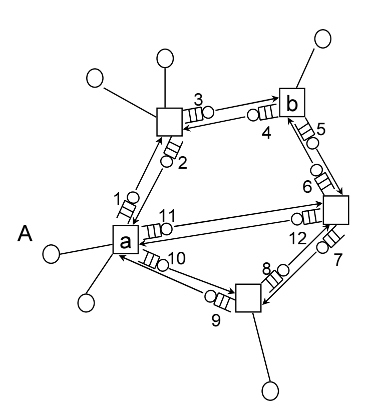
Packet arrival rates in links

- Let
  - $\lambda(r)$ = arrival rate of packets following route r
  - $R(j)$ = the set of routes that use link j
    - can be deduced from the routing tables
- It follows that the arrival rate for link j is as follows:

$$\lambda_j = \sum_{r \in R(j)} \lambda(r)$$

Traffic classes

- Note:
  - Average end-to-end delay is equal for all the packets following the same route
- Thus,
  - the **traffic class** of a packet is determined by the route r that the connection follows



State space

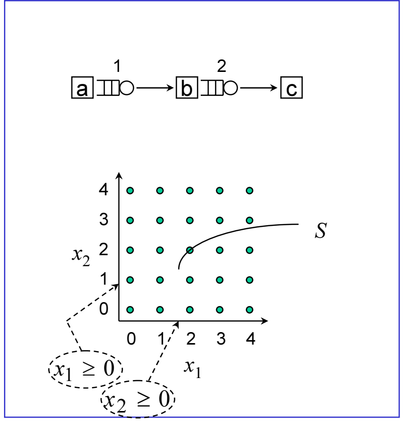
- Let x_j = denote the number of packets in queue j (including the packet being transmitted (if any))
  - $\mathbf{x} = (x_1,\ldots,x_J)$
- Vector $\mathbf{x}$ is called the **state** of the system
  - A more detailed state description (including the position and traffic class of each packet in the whole system) is not needed under the assumptions that we will make later!
- In this case, x_j can have any non-negative value
- Thus, the **state space** S is

$$S = \{\mathbf{x} \geq 0\}$$

  - Note that, set S is now **infinite**

Example

- 2 links:
  - link a-b
  - link b-c
- 3 routes:
  - route a-b
  - route b-c
  - route a-b-c
- State space:
  - S = {(0,0),
    (1,0),(0,1),
    (2,0),(1,1),(0,2),
    (3,0),(2,1),(1,2),(0,3),
    ...}



Queueing network

- Assume that
  - new packets following route r arrive (independently) according to a Poisson process with intensity $\lambda(r)$
  - packet lengths are independently and exponentially distributed with mean L
- It follows that
  - new packets to be transmitted on link j arrive (independently) according to a Poisson process with intensity λ_j, where

$$\lambda_j = \sum_{r \in R(j)} \lambda(r)$$

  - packet transmission times are independently and exponentially distributed with mean $1/\mu_j = L/C_j$

Equilibrium distribution (1)

- Assume further that
  - the system is **stable**: $\lambda_j < \mu_j$ for all j
  - packet length is independently redrawn (from the same distribution) every time the packet moves from one link to another
    - This is so called **Kleinrock's independence assumption**
- Under these assumptions, it is possible to show that
  - the stationary state probability $\pi(\mathbf{x})$ for any state $\mathbf{x} \in S$ is as follows:

$$\pi(\mathbf{x}) = \prod_{j=1}^{J} (1-\rho_j)\rho_j^{x_j}$$

  - where ρ_j denotes the traffic load of link j:

$$\rho_j = \frac{\lambda_j}{\mu_j} = \frac{\lambda_j L}{C_j} < 1$$

Equilibrium distribution (2)

- Probability $\pi(\mathbf{x})$ is again said to be of **product-form**
  - Now, the number of packets in different queues are **independent** (why?)
- Each individual queue j behaves as an M/M/1 queue
  - Number of packets in queue j follows a geometric distribution with mean

$$\overline{X}_j = \frac{\rho_j}{1-\rho_j}$$

Mean end-to-end delay

- Consider then the mean end-to-end delay for class r
  - Let $J(r)$ denote the set of the links that belong to route r
- In our model, the mean end-to-end delay will be
  - the sum of mean delays experienced in the links along the route (including **both** the transmission delay **and** the queueing delay)
- By Little's formula, the mean link delay is

$$\overline{T}_j = \frac{\overline{X}_j}{\lambda_j} = \frac{1}{\lambda_j} \cdot \frac{\rho_j}{1-\rho_j} = \frac{1}{\mu_j} \cdot \frac{1}{1-\rho_j} = \frac{1}{\mu_j - \lambda_j}$$

- Thus, the mean end-to-end delay for class r is

$$\overline{T}(r) = \sum_{j \in J(r)} \overline{T}_j = \sum_{j \in J(r)} \frac{1}{\mu_j(1-\rho_j)} = \sum_{j \in J(r)} \frac{1}{\mu_j - \lambda_j}$$