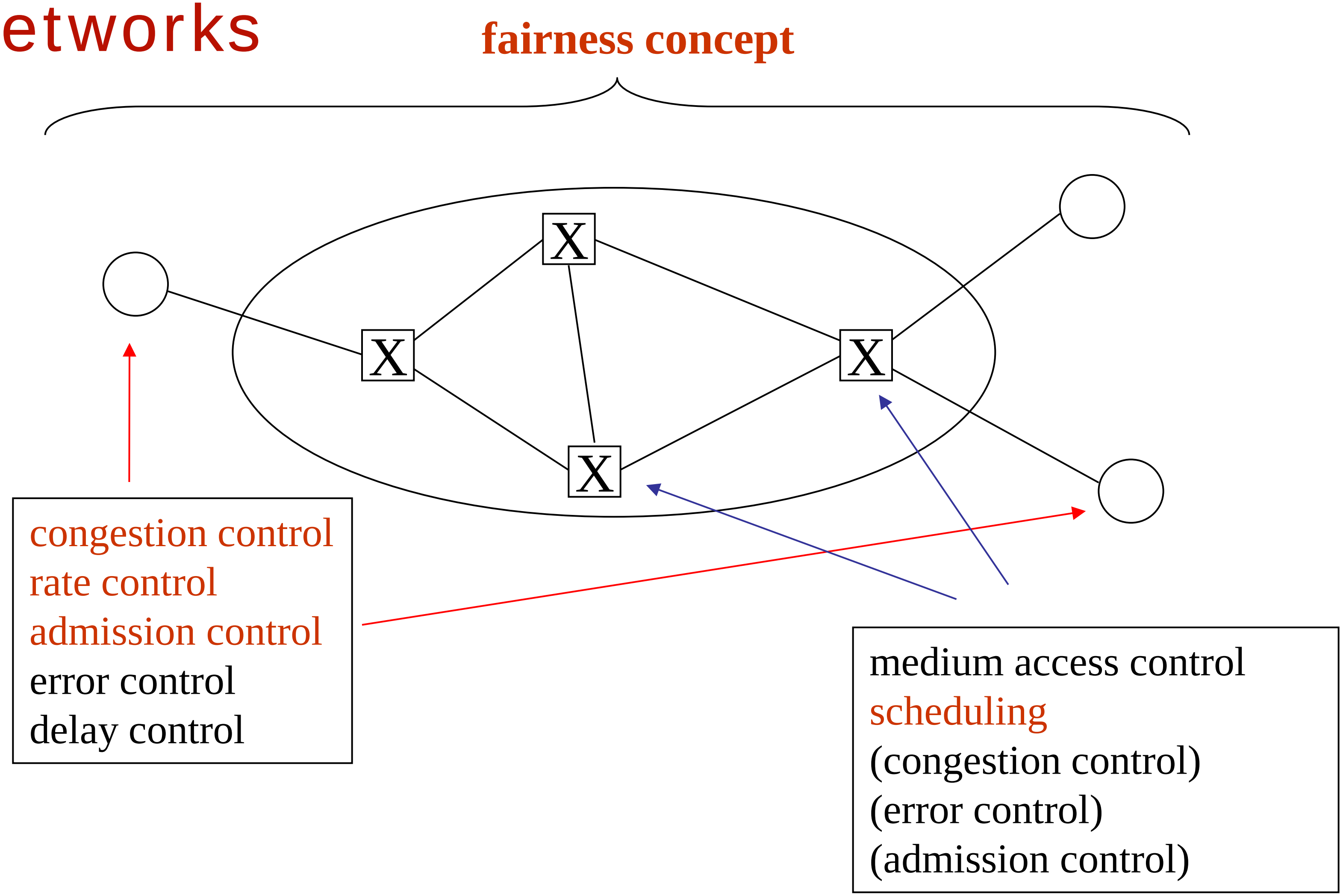
EP2210 Fairness

- Lecture material:
  - Bertsekas, Gallager, *Data networks*, 6.5
  - L. Massoulie, J. Roberts, "Bandwidth sharing: objectives and algorithms,“ IEEE Infocom 2000, Sec. II.B.1, III.C.3.
  - J-Y Le Boudec, “Rate adaptation, congestion control and fairness: a tutorial,” Nov. 2005, 1.2.1, 1.4.
  - MIT OpenCourseWare, 6.829
- Reading for next lecture:
  - L. Massoulie, J. Roberts, "Bandwidth sharing: objectives and algorithms,“ IEEE Infocom 2000.

Control functions in communication networks

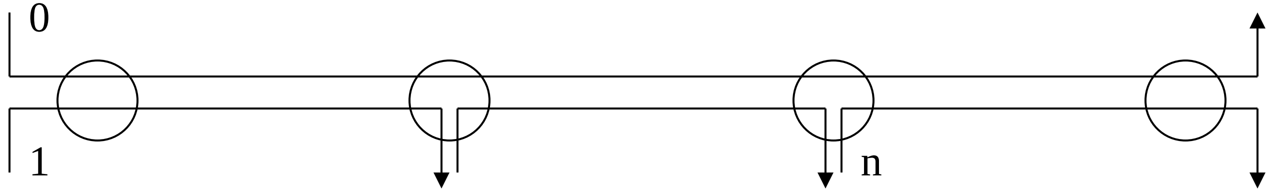


Fairness

- Scheduling: means to achieve fairness on a single link
  - E.g., GPS provides max-min fairness
- Networks?
  - How to define fairness
  - How to achieve fairness

Fairness - objectives

- How to share the network resources among the competing flows? ("parking lot scenario")



Equal rate:

$$r_i = \frac{1}{2}, \quad i = 0..n$$

$$Th = \sum_{i=0}^{n} r_i = \frac{n+1}{2}$$

Maximum network throughput (Th=n would be nice):

$$r_0 = 0$$

$$r_i = 1, \quad i = 1..n$$

$$Th = \sum_{i=0}^{n} r_i = n$$

Equal network resource:
$l_0 * r_0 = l_i * r_i$, l_i is the path length

$$r_0 = \frac{1}{n+1}$$

$$r_i = \frac{n}{n+1}$$

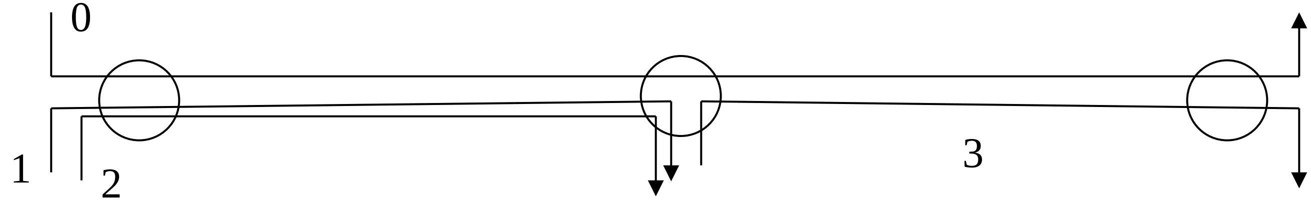
$$Th = \frac{n^2+1}{n+1}$$

Fairness - objectives and algorithms

- Step 1: what is the “optimal” share?
  - What is optimal – a design decision
  - Fairness definitions
  - Centralized algorithms to calculate fair shares
- Step 2: how to ensure fair shares?
  - Traffic control at the network edges (congestion or rate control)
  - Scheduling at the network nodes
- This lecture:
  - max-min fairness definition and allocation algorithm
  - proportional fairness, other fairness definitions
- Student presentation:
  - distributed control for fairness

Max-Min Fairness

- Simplest case:
  - without requirements on minimum or maximum rate
  - constraints are the link bandwidths
- Definition: Maximize the allocation for the most poorly treated sessions, i.e., *maximize the minimum*.
- Equivalent definition: allocation is max-min fair if no rates can be increased without decreasing an already smaller rate



$$r_0 = r_1 = r_2 = \frac{1}{3}, \quad r_3 = \frac{2}{3}$$

Max-Min Fairness

- Formal description:
  - allocated rate for session p: r_p, $\mathbf{r}=\{r_p\}$
    (maximum and minimum rate requirements not considered)
  - allocated flow on link a: $F_a=\sum_{p\in a} r_p$
  - capacity of link a: C_a

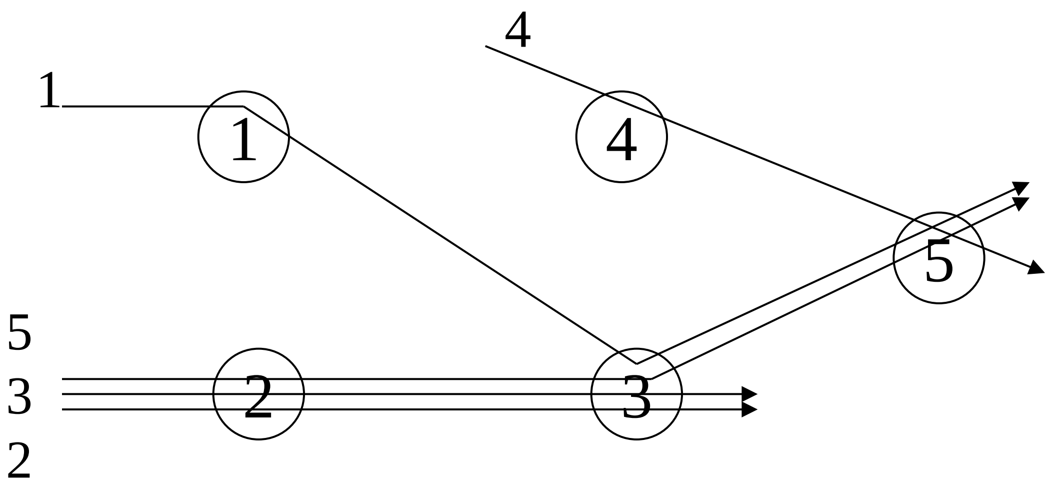
  Feasible allocation r: $r_p \geq 0$, $F_a \leq C_a$

  Max-min fair allocation r:
  - consider **r** max-min fair allocation and **r*** any feasible allocation
  - for any feasible $\mathbf{r^*} \neq \mathbf{r}$ for which $r^*_p > r_p$
    *(if in **r*** there is a session that gets higher rate)*
  - there is a p' with $r_{p'} \leq r_p$ and $r^*_{p'} < r_{p'}$
    *(then there is a session that has minimum rate in **r** and has even smaller rate in **r***.)*

Max-Min Fairness

- Simple algorithm to compute max-min fair rate vector r
  - Idea: filling procedure
    1. increase rates for all sessions until one link gets saturated (the link with highest number of sessions if there are no max. rates)
    2. consider only sessions not crossing saturated links, go back to 1
  - Formal algorithm in B-G p.527
  - Note, it is a centralized algorithm, it requires information about all sessions.

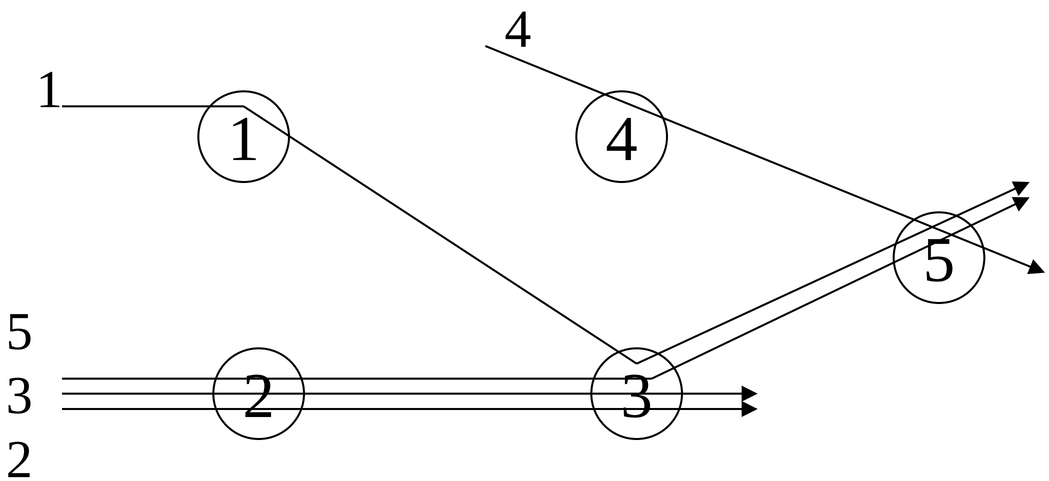
Max-Min Fairness



Filling procedure:

1. increase rates for all sessions until one link gets saturated
   (the link with highest number of sessions if there are no max. rates)
2. consider only sessions not crossing saturated links, go back to 1

Max-Min Fairness

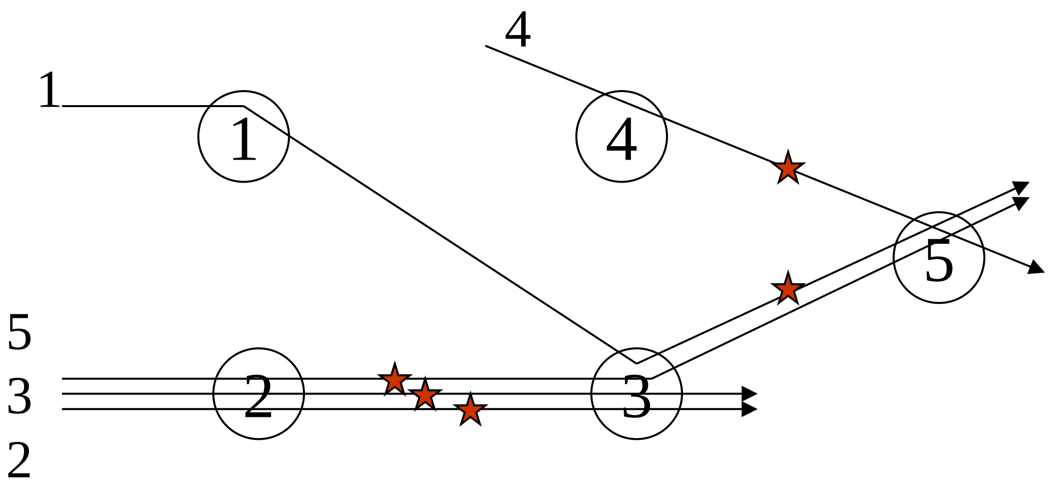


1. All sessions get rate of 1/3, link(2,3) saturated, r2=r3=r5=1/3
2. Sessions 1 and 4 get rate increment of 1/3, link(3,5) saturated, r1=2/3
3. Session 4 gets rate increment of 1/3, link(4,5) saturated, r4=1

What happens with the rates if session 2 leaves?

Max-Min Fairness

- Can we evaluate whether an allocation is max-min fair?
- Proposition: Allocation is max-min fair if and only if each session has a bottleneck link
- Def: *a* is a bottleneck link for p if $F_a = C_a$ and $r_p \geq r_{p'}$ for all $p' \neq p$
- Find the bottleneck links for p1,p2,p3,p4,p5.



r2=r3=r5=1/3, r1=2/3, r4=1

* : bottleneck link

Max-Min Fairness

- Proposition: Allocation is max-min fair *if and only if* each session has a bottleneck link

1. *If **r** is max-min fair then each session has a bottleneck link*
2. *If each session has a bottleneck link then **r** is max-min fair*

- Why do we like this proposition: given allocation **r** it is easy to check if a session has a bottleneck link or not, and this way we can see if **r** is max-min fair or not.

Max-Min Fairness

Proof:

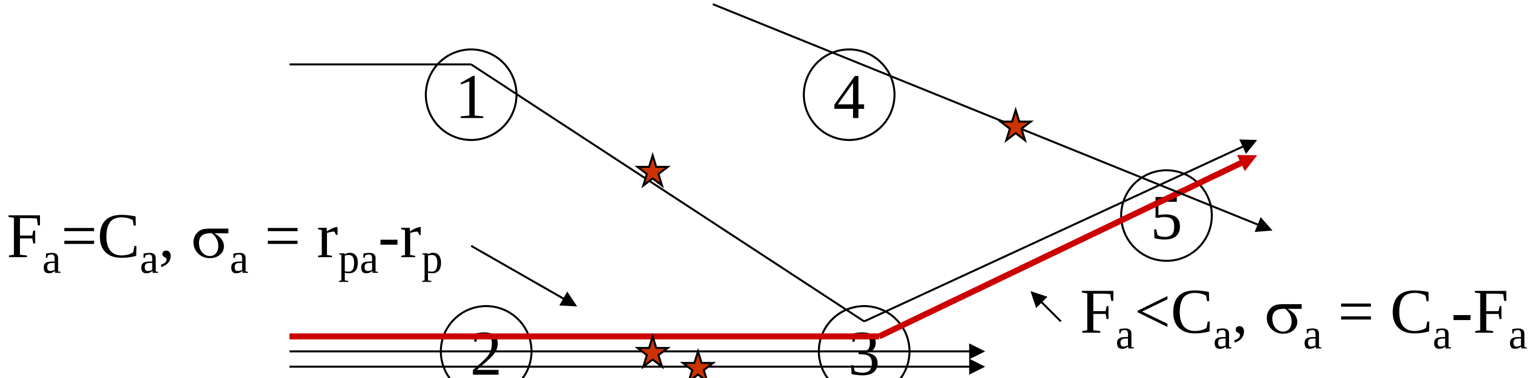
1. *If **r** is max-min fair then each session has a bottleneck link*

Def: *a* is a bottleneck link for p if $F_a = C_a$ and $r_p \geq r_{p'}$ for all $p' \neq p$

Proof with contradiction: assume max-min, but p does not have bottleneck link.

- For all link *a* on the path, define σ_a:
- if $F_a = C_a$, then there is at least one session with rate r_{pa} higher than r_p, and let $\sigma_a = r_{pa} - r_p$ and
- if $F_a < C_a$, then the link is not saturated, and let $\sigma_a = C_a - F_a$.

Possible to increase r_p with $\min(\sigma_a)$ without decreasing rates lower than r_p. This contradicts the max-min fairness definition.



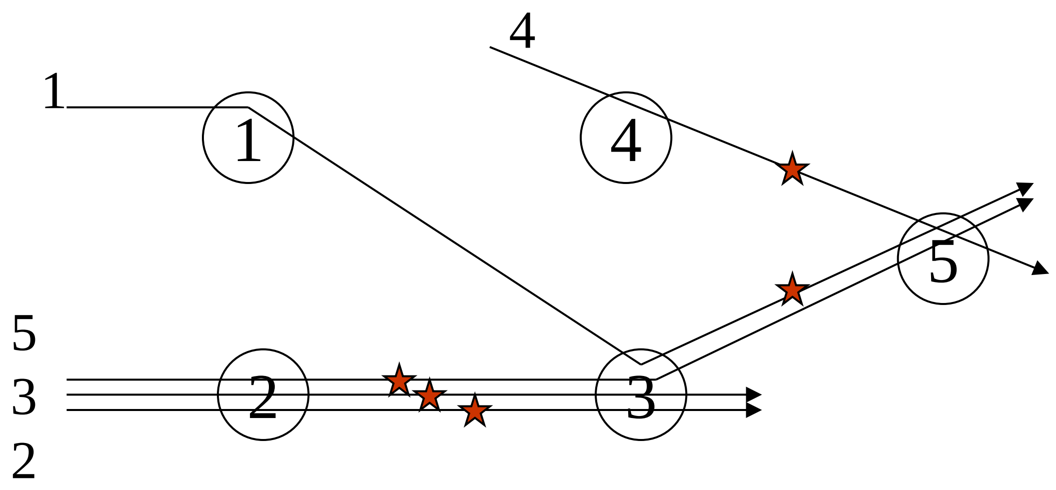
Max-Min Fairness

Proof:

2. *If each session has a bottleneck link then* ***r*** *is max-min fair*

Proof: consider the following for each session.

- Consider session p with bottleneck link *a* ($F_a = C_a$)
- Due to the definition of bottleneck link $r_{pa} \leq r_p$ and consequently r_p can not be increased without decreasing a session with lower rate.
- This is true for all sessions, thus the allocation is max-min fair.



Other fairness definitions - Utility function

- Utility function: to describe the value of a resource.
- E.g.,
  - Application requires fixed rate: *r**
  - Allocated rate: r
  - Utility of allocated rate:
    $u(r)=0$ if $r<r^*$
    $u(r)=1$ if $r>=r^*$
- Typical utility functions:
  - Linear $u(r)=r$
  - Logarithmic $u(r)=\log r$
  - Step function – as above

Rate-proportional fairness

- Name: rate proportional or proportional fairness
- Note! Change in notation! Rate: λ, flow: r, set of flows: R
- Def1: Allocation $\Lambda = \{\lambda_r\}$ is proportionally fair if for any $\Lambda' = \{\lambda'_r\}$:

$$\sum_R \frac{\lambda_r' - \lambda_r}{\lambda_r} \leq 0$$

- thus, for all other allocation the sum of *proportional rate changes* with respect to Λ are negative.
- Def2: The proportionally far allocation maximizes $\sum_R \log\lambda_r$ – maximizes the overall utility of rate allocations with a *logarithmic utility function*.

Rate-proportional fairness

- Example: parking lot sceneario
- L links, R_0 crosses all links, others only one link

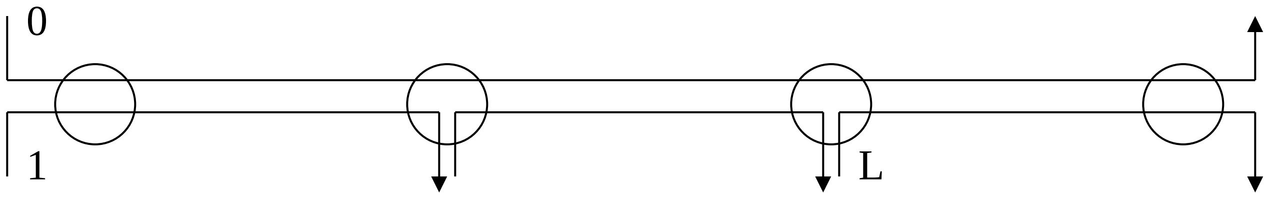
Maximize $\sum_{i=0}^{L} \log \lambda_i$

$$\sum_{i=0}^{L} \log \lambda_i = \log \lambda_0 + \sum_{i=1}^{L} \log \lambda_i = \log \lambda_0 + L \log(1 - \lambda_0)$$

$$\frac{\partial}{\partial \lambda_0} (\log \lambda_0 + L \log(1 - \lambda_0)) = 0$$

$$\Rightarrow \quad \frac{1}{\lambda_0} - \frac{L}{1-\lambda_0} = 0$$

$$\lambda_0 = \frac{1}{1+L}, \quad \lambda_i = \frac{L}{1+L}$$



Rate-proportional fairness

Maximize $\sum_{i=0}^{L} \log \lambda_i$

$$\sum_{i=0}^{L} \log \lambda_i = \log \lambda_0 + \sum_{i=1}^{L} \log \lambda_i = \log \lambda_0 + L \log(1 - \lambda_0)$$

$$\frac{\partial}{\partial \lambda_0} \left(\log \lambda_0 + L \log(1 - \lambda_0)\right) = 0$$

$$\Rightarrow \quad \frac{1}{\lambda_0} - \frac{L}{1-\lambda_0} = 0$$

$$\lambda_0 = \frac{1}{1+L}, \quad \lambda_i = \frac{L}{1+L}$$

- Long routes are penalized
- The same as the “equal resources” scenario on the first slides.

Rate-proportional fairness – equivalence of definitions

- Let $\{\lambda_i{}^*\}$ be the optimal rate allocation and an other $\{\lambda_i'\}$ allocation.

Let $\lambda_i' = \lambda_i^* + \Delta_i$

$$\begin{aligned}
\sum_{i=0}^{L} \log \lambda_i' &= \sum_{i=0}^{L} \log\left(\lambda_i^* + \Delta_i\right) \\
&= \sum_{i=0}^{L} \log \lambda_i^* + \sum_{i=0}^{L} \frac{\Delta_i}{\lambda_i^*} + o(\Delta^2) \\
\sum_{i=0}^{L} \log \lambda_i' &\approx \sum_{i=0}^{L} \log \lambda_i^* + \sum_{i=0}^{L} \frac{\Delta_i}{\lambda_i^*} \quad \Rightarrow \quad \sum_{i=0}^{L} \frac{\Delta_i}{\lambda_i^*} \leq 0 \\
&\Leftrightarrow \sum_{i=0}^{L} \frac{\lambda_i' - \lambda_i^*}{\lambda_i^*} \leq 0.
\end{aligned}$$

Other bandwidth sharing objectives

- L. Massoulie, J. Roberts, "Bandwidth sharing: objectives and algorithms,“ IEEE Infocom 2000, sections I and II.

- *Max-min*
- *Proportional*
- Potential delay minimization

- Weighted shares for various fairness definitions

Potential delay minimization

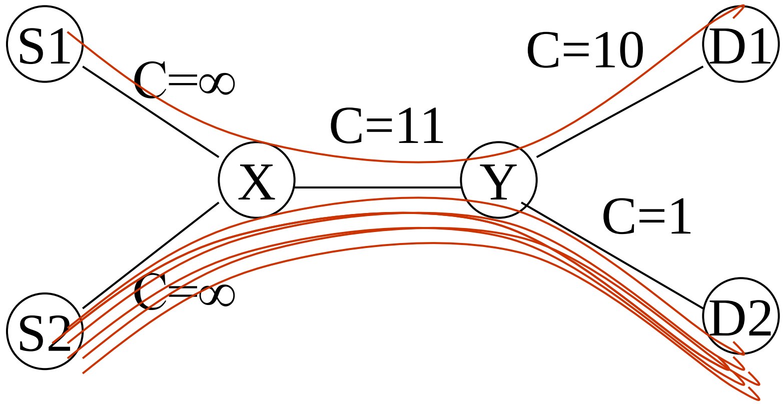
- Bandwidth sharing objective: minimize the delay of all transfers (elastic flows)
- File transfer time: inversly proportional to rate λ
- Objective: $min \sum 1/\lambda_r$

Fairness – distributed control

- We have seen a number of fairness definitions and bandwidth sharing objectives
- Fair allocation for a given set of flows can be calculated (filling, or solving the related optimization problem).
- How can fair allocation be provided in a distributed way?

Traffic control for max-min fairness

- GPS provides max-min fairness for a single node.
- What happens in networks with GPS nodes but without any end-to-end control? Is max-min fairness achieved?
- Multiple node example:
  - 1 flow from S1 to D1
  - 10 flows from S2 to D2

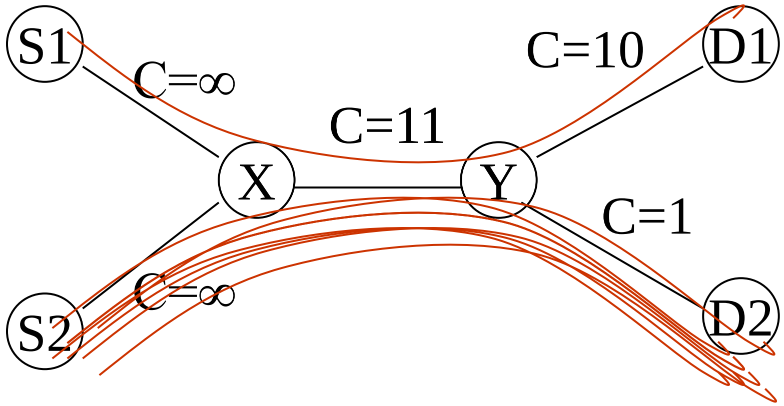


- Calculate the max-min fair rates for the entire network.

- Flow to D1: 10
- Flows to D2: 0.1

Traffic control for max-min fairness

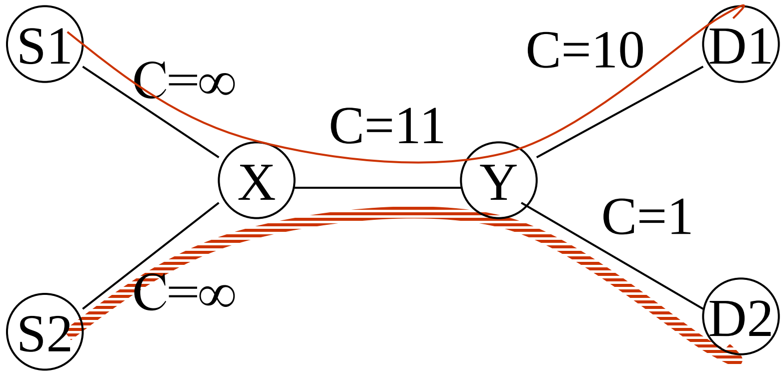
- GPS provides max-min fairness for a single node.
- What happens in networks with GPS nodes but without any rate control? Is max-min fairness achieved?
- Multiple node example:
  - 1 flow from S1 to D1
  - 10 flows from S2 to D2



- Calculate the per flow rates on the links when node X and Y provides GPS, independently from each other.
  (X considers the traffic that arrives to it from S1 and S2, Y considers the traffic arriving from X.)

Traffic control for max-min fairness

- GPS provides max-min fairness for a single node.
- What happens in networks with GPS nodes but without any rate control? Is max-min fairness achieved?
- Multiple node example:
  - 1 flow from S1 to D1
  - 10 flows from S2 to D2



- Without rate control:
  - X: rate 1 to all flows
  - Y: rate 0.1 to flows to D2
  - Result:
    - Flow to D1: 1
    - Flows to D2: 0.1
- Fair rates would be:
  - Flows to D1: 10
  - Flows to D2: 0.1

- Thus, max-min fairness is not achieved without end-to-end control.

Traffic control for fairness

- Student presentation on how to achieve fairness with distributed control –

Traffic control for fairness

- How to achieve fairness with distributed control – other results from Massoulie and Roberts
- With fixed window size:
  - FIFO achieves proportional fairness
  - longest queue first achieves maximum throughput
  - service proportional to the square root of the buffer content achieves minimum potential delay
- With dynamic window:
  - additive increase multiplicative decrease achieves proportional fair allocation (case of TCP)
  - logarithmic increase multiplicative decrease achieves minimum potential delay
  - max-min fair rate can not be achieved with increase-decrease algorithms

Fairness - objectives and algorithms - summary

- Step 1: what is the “optimal” share?
  - What is optimal – a design decision
  - Fairness definitions: max-min, proportional fair, etc.
  - Centralized algorithms to calculate fair shares
- Step 2: how to ensure fair shares?
  - Traffic control at the network edges (congestion or rate control)
  - Scheduling at the network nodes
  - E.g:
    - fixed window based congestion control + GPS: max-min
    - AIMD + FIFO: proportional fair

Processor sharing queue

- The performance of GPS (single link or single resource) under stochastic request arrival.
- Recall: for FIFO service, Poisson arrivals, Exp service time distributions we have M/M/1 queue.
- Question: how can we model the GPS service?

Processor sharing queue

- The performance of GPS (single link or single resource) under stochastic request arrival. Fluid model.

- Single server (single link, transmission medium or resource)
- The capacity of the server equally shared by the requests
  - if there are n requests, each receives service at a rate C/n
  - customers do not have to wait at all, service starts as the customer arrives (there is no queue…)
- M/M/1-PS
  - Poisson customer arrival process (λ)
  - Service demand (job size) is exponential in the sense, that if the customer got all the service capacity, then the service time would be Exp(μ) (models e.g., exponential file size)
  - Note: if the number of requests is higher, a request stays in the server for a longer time.

Processor sharing queue

- M/M/1-PS
  - Poisson customer arrival process (λ)
  - service demand (job size) is exponential in the sense, that if the customer got all the service capacity, then the service time would be Exp(μ)
- Draw the Markov chain
- Explain why is it the same as for the M/M/1-FIFO queue.

- Consequently, E[N] and E[T] is the same as M/M/1-FIFO

$$E[N] = \frac{\lambda/\mu}{1-\lambda/\mu}, \quad E[T] = \frac{E[N]}{\lambda} = \frac{1/\mu}{1-\lambda/\mu}$$