EE 597

Class Lecture notes 4/24/12

Congestion Control in Wireless Networks

- The need for Explicit Congestion Notification to deal with the problem of loss differentiation.

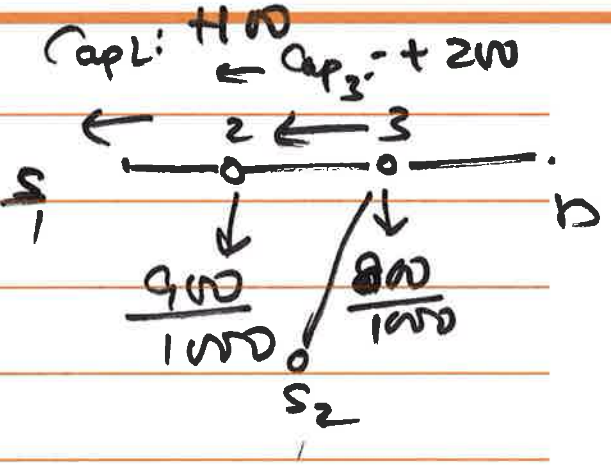
- The need for Congestion sharing to alleviate fairness problem arising due to shared medium.

Both of these solutions require greater intelligence ~~to~~ be embedded at the core of the network, i.e. in the routers, violating the end to end architecture of traditional Internet.

Another new idea in congestion control is "explicit & precise rate control"

- contrast w/ AIMD approach

for general (i.e. wired) networks:
RCP by Dukkipati & McKeown



Also a router-centric approach (hence not adopted earlier)

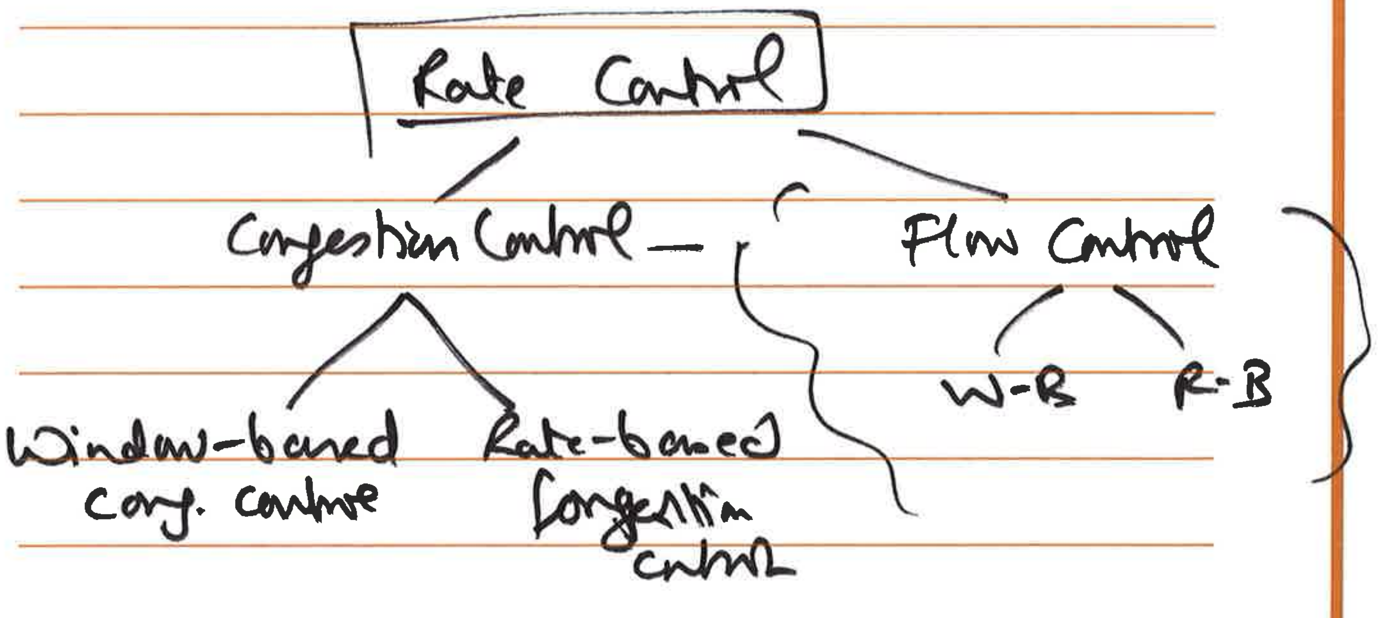
Big Industry development:
OpenFlow Routers —
Software defined routers

This is likely to speed adoption of router-centric protocols for wired networks

Examples in Wireless networks of E&P rate control:

- WRCP for sensor networks
- WCP-Cap for Mesh networks

Congestion Control with Backpressure



Observation: can solve some of the congestion-control problems such as fairness, at the MAC layer.

—

Backpressure Scheduling

Recall max weight algorithm: schedule independ set of links that max $\sum_{i,j,c} w_{ij}^{c}$

where $w_{ij}^{c} = (Q_i^{c} - Q_j^{c}) \cdot R_{ij}$

—

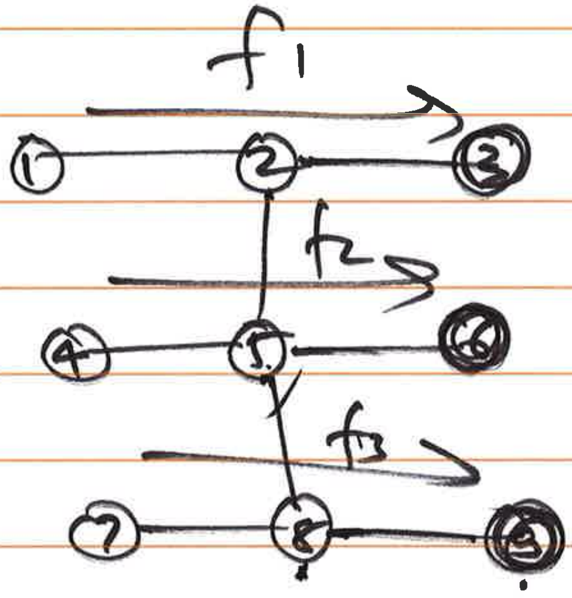
BCP was a distributed impl. at the routing layer (mostly)

DiffQ : an dist. implementation of Backpressure at the MAC layer

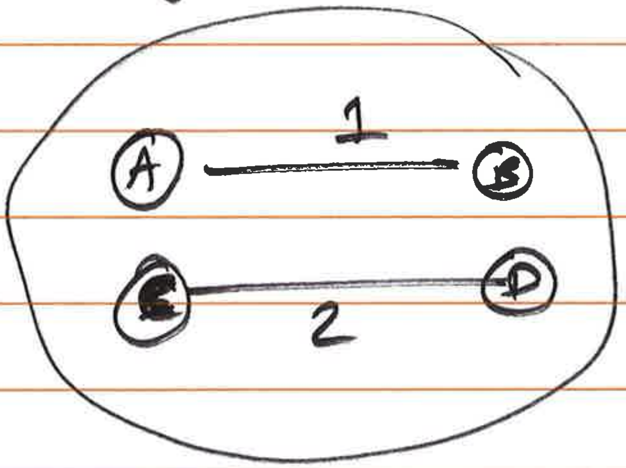
Basic idea of DiffQ:
choose the backoff interval based on ΔQ_{ij} (i.e. w_{ij}).

if we want to prefer transmissions on link w/ higher weight, want to make the max Backoff window smaller.

DiffQ: took 802.11e which specifies different B.O. window sizes for different QoS classes, & map the w_{ij} to classes.

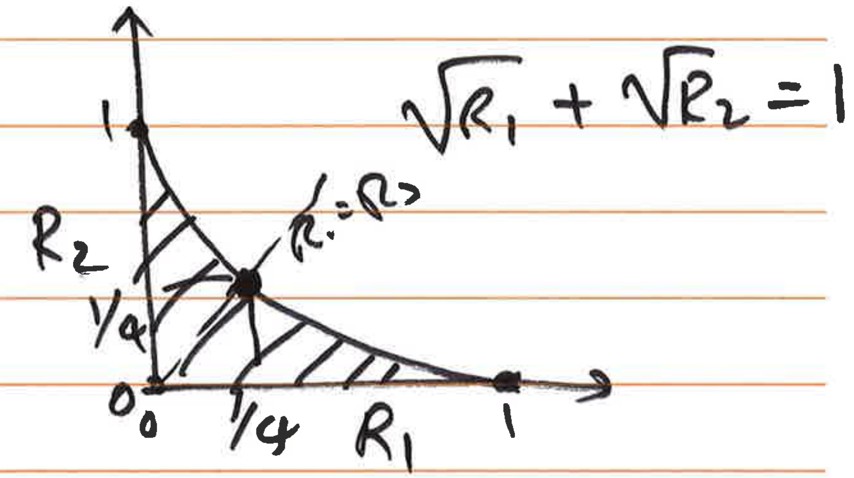


Utility-based rate control.



MAC layer:
slotted
Aloha.

Slotted Aloha Saturated Throughput Region :



Throughput efficient solution: (1, 0)
or (0, 1)

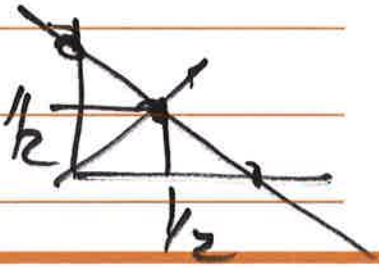
Sum rate = 1.

Clearly least fair solution!

Most Fair solution : (1/4, 1/4)

least sum rate : 1/2

w/ TDMA, can get both.



Kelly '98

Utility-based rate allocation

$$\max \sum_i g_i(r_i)$$

$$\text{s.t. } \vec{r} \in \Delta$$

rate region of the network.

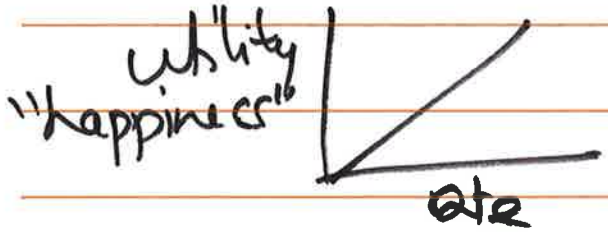
$i \leftarrow i^{th}$ flow.

$r_i \leftarrow$ rate allocated to flow i

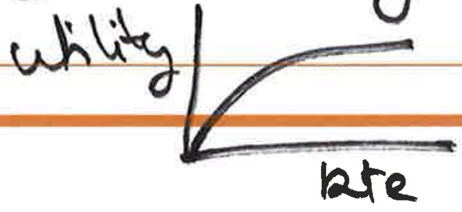
$g_i()$ is a utility function. that is concave (second derivative negative or zero). & monotonic

Special cases of $g_i(r_i)$.

Linear utility : $g_i(r_i) = c_i \cdot r_i$



Log utility : $g_i(r_i) = c_i \log(1+r_i)$



w/ slotted Aloha example

$c_i = 1 \ \forall i$

Linear utility

$$\max \ R_1 + R_2$$
$$\text{s.t. } \sqrt{R_1} + \sqrt{R_2} \le 1$$

$$\Longleftrightarrow \quad \max \ R_1 + R_2$$
$$\text{s.t. } \sqrt{R_1} + \sqrt{R_2} = 1$$

Solns: either $R_1 = 1 \quad R_2 = 0$

OR $\quad R_2 = 1 \quad R_1 = 0.$

Log utility - Proportional fair utility

$$\max \ \log R_1 + \log R_2$$
$$\text{s.t. } \sqrt{R_1} + \sqrt{R_2} = 1$$

$$\sqrt{R_2} = 1 - \sqrt{R_1}$$

$$R_2 = (1 - \sqrt{R_1})^2$$

$$\max \ \log R_1 + \log (1 - \sqrt{R_1})^2$$

$\Leftrightarrow$

$$\max \left(\log R_1 + 2\log(1-\sqrt{R_1})\right)$$

$$\frac{\partial \cdot}{\partial R_1} = 0 \Rightarrow \frac{1}{R_1} + \frac{\cancel{2}\cdot(-1)}{(1-\sqrt{R_1})\sqrt{R_1}\,\cancel{2}} = 0$$

$$(1-\sqrt{R_1})(\sqrt{R_1}) = R_1$$

$$\sqrt{R_1} - R_1 = R_1$$

$$\sqrt{R_1} = 2R_1$$

$$\sqrt{R_1} = \frac{1}{2}$$

$$\boxed{R_1 = 1/4}$$

$$\Rightarrow R_2 = 1/4$$

soln : $(1/4, 1/4)$

more
Generally : α-fair [illegible] utility function

Prof. Mike Neely's Dissertation at MIT :
utility optimization in a Backpressure framework

How to maximize utility (i.e. pick a desired rate allocation across the flows / commodities) while ensuring stable operation.

pb:

$$\max \sum g_i(r_i)$$
$$\text{s.t.} \quad r_i \in \Lambda$$

stability region of the network

Solution:

Part 1: rate allocation

each source picks rate r_i

to maximize:

$$\left(V \cdot g_i(R_i) - Q_i \cdot R_i \right)$$

only needs local queue state!!!!

Part 2: use maxWeight scheduling algorithm as before.

Part 1 rate control

$$V g'(R_i) - Q_i = 0.$$

pick R_i : $g'(R_i) = \frac{Q_i}{V}$

$$\underline{\underline{R_i}} = g'^{-1}\left(\frac{Q_i}{V}\right)$$

Ⓥ parameter tradeoff between utility optimization & average queue size (hence delay).

can be hard to find the best value of V for a given network

sensitive to the topology & traffic