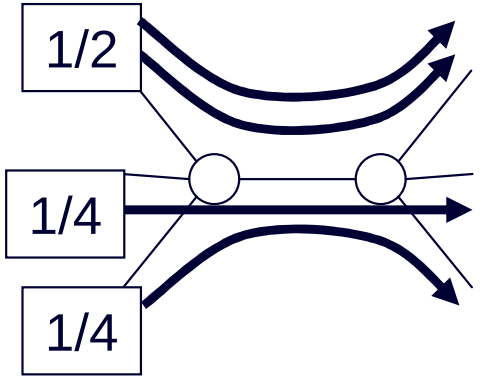
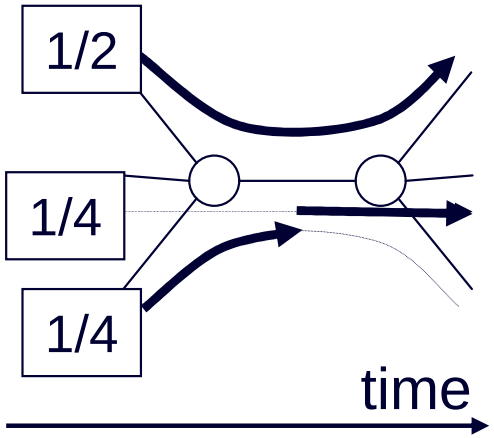
flow rate fairness dismantling a religion

Bob Briscoe
Chief Researcher, BT Group
IRTF E2ERG Feb 2007

today's shares are just the result of a brawl

- flow rate fairness is not even wrong
  - it doesn't even answer the right questions
  - it doesn't allocate the right thing
  - it doesn't allocate between the right entities
- how do you answer these questions?
  1) how many flows is it fair for an app to create?
  2) how fast should a brief flow go compared to a longer lasting one?





why the destructive approach?

destruction ...breeds creation

- resource allocation/accountability
  - ‘needs fixing’ status since early Internet
- will never get past ‘needs fixing’
  - unless we discard an idea that predated the Internet
- fairness between flow rates (used in TCP fairness, WFQ)
  - proven bogus 9yrs ago, but (I think) widely misunderstood / ignored
  - so we have no fairness at all
  - fairness between flow rates still the overwhelmingly dominant ideology
  - obscured by this idea, we wouldn’t know a bad fix from a good one
- this is important
  - probable cause of DPI middleboxes

- now ‘being fixed’
  - e.g. Re-ECN: Adding Accountability for Causing Congestion to TCP/IP

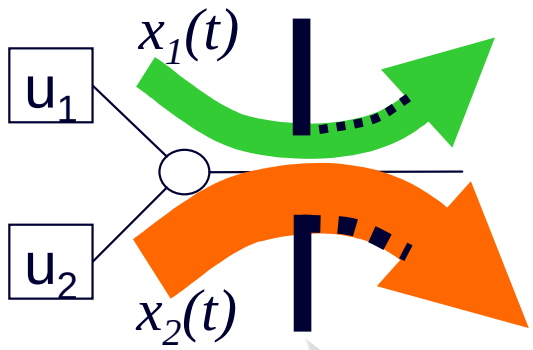
    <draft-briscoe-tsvwg-re-ecn-tcp-03.txt>
- this talk is *not* about re-ECN
  - but about why we need something like it
- nonetheless, to reassure you...
  - don’t need to throw away everything we’ve already engineered
  - despite being based on congestion pricing theory, don’t need to throw away traditional flat retail pricing

You got to be careful if you don't know where you're going, because you might not get there [Yogi Berra]

fair allocation... of what? among what?

☑ of 'cost' among bits

- cost of one user's behaviour on other users
  - congestion volume ≡ instantaneous congestion p...
  - ...shared proportionately over each user's bit rate, x_i
  - ...over (any) time
  - $v_i \equiv \int p(t)x_i(t)\,dt$
- volume of dropped/marked data each user sent
  - integrates simply and correctly over time and over flows

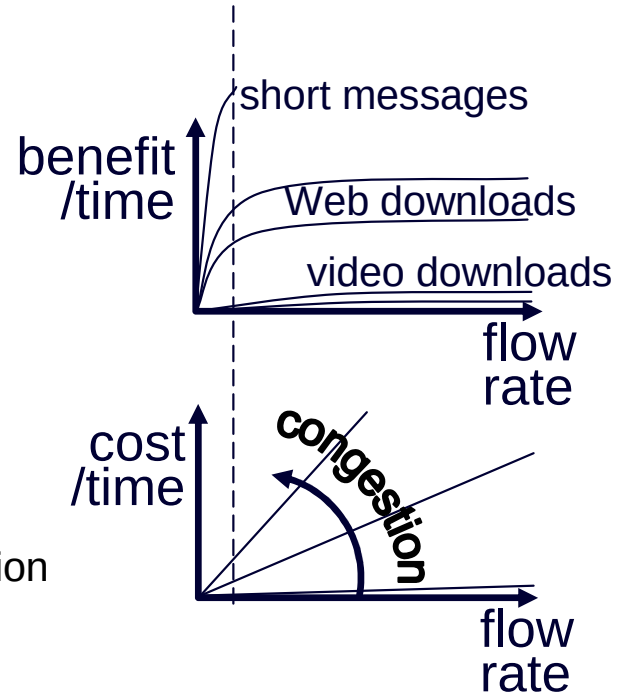


$$p(t) \equiv \frac{excess\ load}{offered\ load}$$

fair allocation... of what?

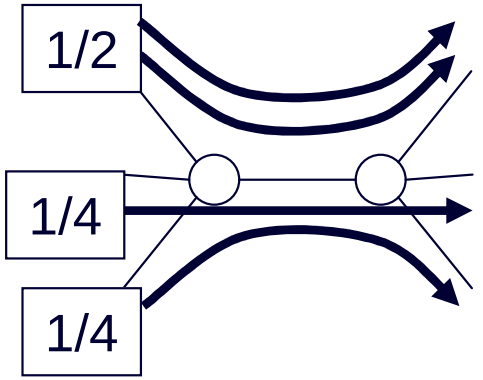
☒ not rate

- what discipline deals with fairness?
  - political economy (supported by philosophy)
- fairness concerns shares of
  - benefits (utility), costs or both
- benefit ≠ flow rate
  - users derive v different benefit per bit from each app
- cost ≠ flow rate
  - cost of building network covered by subscriptions
  - cost to other users depends on congestion
  - no cost to other users (or network) if no congestion
  - very different costs for same flow rate with diff congestion
- "equal flow rates are fair"?
  - no intellectual basis: random dogma
- even if aim were equal benefits / costs
  - equal flow rates would come nowhere near achieving it



fair allocation... among what?

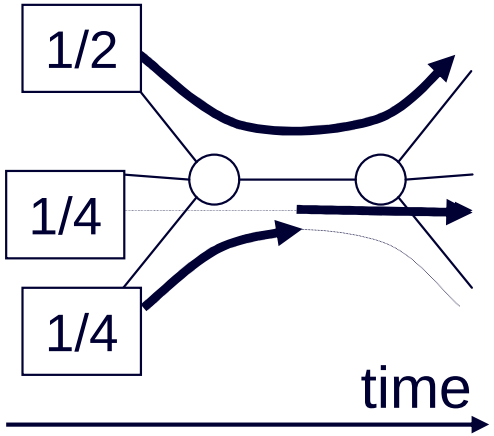
☒ not flows



- we expect to be fair to people, institutions, companies
  - ‘principals’ in security terms
- why should we be fair to transfers between apps?
  - where did this weird argument come from?
  - like claiming food rations are fair if the boxes are all the same size
    - irrespective of how many boxes each person gets
    - or how often they get them

fair allocation...

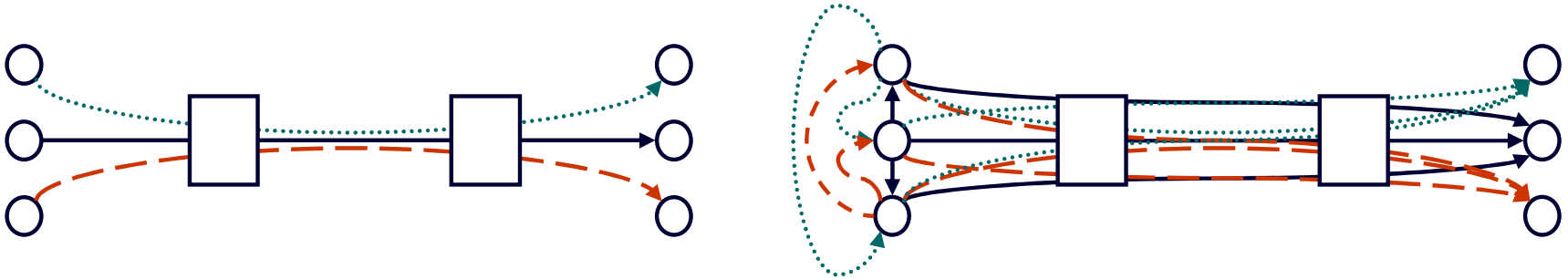
☑ among users, over time



- users A & B congest each other
  - then A & C cause similar congestion, then A & D...
  - is it fair for A to get equal shares to each of B, C & D each time?
- in life fairness is not just instantaneous
  - even if Internet doesn't always work this way, it must *be able* to
  - efficiency and stability might be instantaneous problems, but not fairness
- need somewhere to integrate cost over time (and over flows)
  - the sender's transport and/or network edge are the natural place(s)
- places big question mark over router-based fairness (e.g. XCP)
  - at most routers data from any user might appear
    - each router would need per-user state
    - and co-ordination with every other router

enforcement of fairness

- if it's easy to 'cheat', it's hardly a useful fairness mechanism
  - whether intentionally or by innocent experimentation
- if every flow gets equal rate
  - the more flows you split your flow into, the more capacity you get
  - fairness per source-destination pair is no better
    - Web/e-mail hosting under one IP addr
    - stepping stone routing (cf bitTorrent)



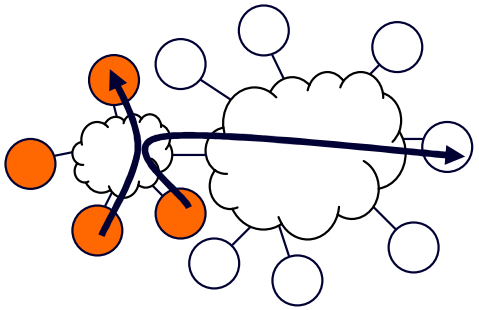
- by design, cost alloc'n among *bits* is immune to identifier cheats

missing the point
due to flow rate obsession

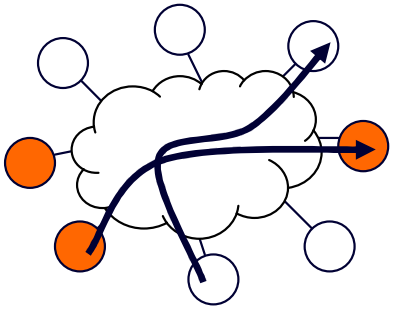
- max-min-, proportional-, TCP- fairness of flow rates
  - not even in same set as weighted proportional fairness
    - "flow A can go w times as fast as B"
    - hardly a useful definition of fairness if A can freely choose w*
    - interesting part is what regulates A's choice of w
- flow rates & their weights: outcome of a deeper level of fairness
  - congestion cost fairly allocated among bits (RED algorithm): cost fairness
  - if users (economic entities) accountable for cost of their bits
    - they will arrange their flow rates to be weighted by their (private) utility
  - the measure of fairness is *not* the resulting relative flow rates because w is private*
    - making users account for congestion costs is in itself sufficient fairness
- Kelly proved cost fairness maximises global benefits
  - any other allocation would reduce benefit
  - also, costs can easily be re-allocated to bring about other forms of fairness...

* original XCP paper, for example, makes this common mistake

fairness between fairnesses



- to isolate a subgroup who want their own fairness regime between them
  - must accept that network between them also carries flows to & from other users
- in life, local fairnesses interact through global trade
  - e.g. University assigns equal shares to each student
    - but whole Universities buy network capacity from the market
  - further examples: governments with social objectives, NATO etc



- cost fairness sufficient to support allocation on global market
  - then subgroups can reallocate the right to cause costs within their subgroup
    - around the edges (higher layer)
  - naturally supports current regime as one (big) subgroup
    - incremental deployment
- different fairness regimes will grow, shrink or die
  - determined by market, governments, regulators, society – around the edges
  - all over congestion marking at the IP layer – neck of the hourglass

conclusions

- we have nothing to lose but an outdated dogma
  - we can keep everything we've engineered, and traditional pricing
  - but no-one should ever again claim fairness based on flow rates
  - unless someone can give a rebuttal
    using a respected notion of fairness from social science

- this is important
  - conflicts between real people / businesses
  - probable cause of DPI middleboxes
- TCP, WFQ etc are insufficient to control fairness
  - we have freedom *without any form of fairness at all*
  - × rate is absolutely nothing like a measure of fairness
  - × being fair to flows is as weird as talking to vegetables
  - × not considering fairness over time is a huge oversight

$$\sum_{\forall i} v_i \equiv \sum_{\forall i} \int p(t) x_i(t)\, dt$$

- cost fairness requires users to be accountable for congestion costs
  - based on sound economics, justified by maximising global benefit
- sub-groups can assert different fairness regimes at higher layers

---

- re-ECN aims to make this underlying 'cost fairness' practical
  - networks can regulate congestion with engineering, rather than Kelly's pricing
  - plan to explain from scratch in Bar BoF at Prague IETF
    - also bar mitzvahs, weddings, after-dinner speeches, ...

flow rate fairness: dismantling a religion

<draft-briscoe-tsvarea-fair-00.pdf>
<www.sigcomm.org/ccr/drupal/?q=node/172>

spare slides:

- is this important?
- definition of congestion notification
- capturing (un)fairness during dynamics
- specific problems with rate fairness:
  - TFRC
  - max-min
- why cost fairness, not benefit fairness
- calibrating ‘cost to other users’
- next steps, incl. re-ECN
  <draft-briscoe-tsvwg-re-ecn-tcp-03>

Q&A

UCL

BT

exec summary

fair allocation... of what?	among what?
✗ rate	✗ flows
✓ congestion	✓ bits, sent by users’

is this important?

- working with packets depersonalises it
  - it's about conflicts between real people
  - it's about conflicts between real businesses
- 1st order fairness – average over time
  - 24x7 file-sharing vs interactive usage
- 2nd order fairness – instantaneous shares
  - unresponsive video streaming vs TCP
  - fair burden of preventing congestion collapse
- not some theoretical debate about tiny differences
  - huge differences in congestion caused by users on same contract
  - hugely different from the shares a `fairness god' or market would allocate
  - yes, there's a lot of slack capacity, but not that much in the backhaul and not for ever
- allocations badly off what a market would allocate
  - eventually lead to serious underinvestment in capacity
- 'do nothing' will not keep the Internet pure
  - without an architectural solution, we get more and more middlebox kludges

definition of congestion notification

from the outside looking in

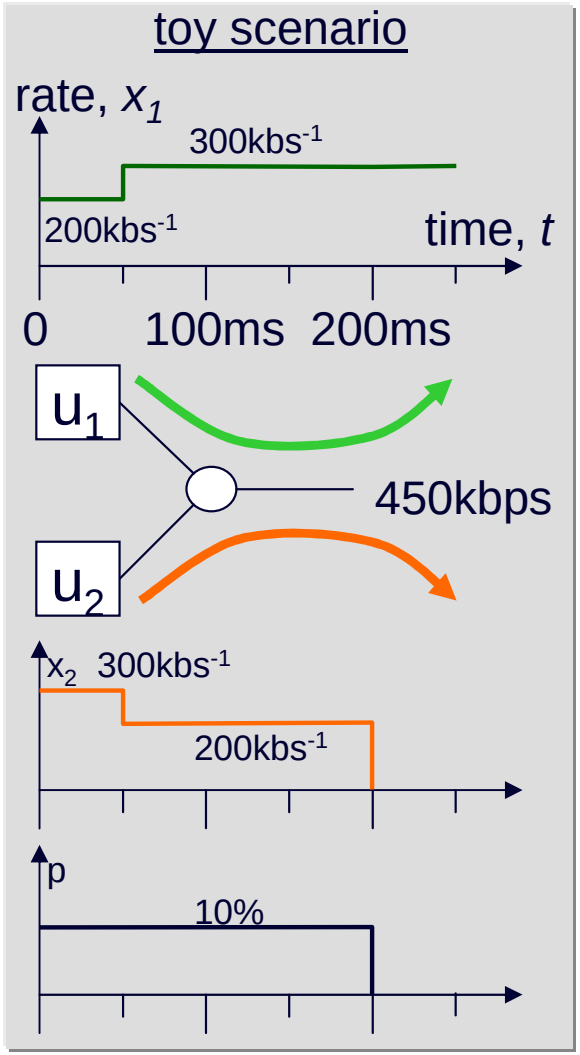
- instantaneous resource congestion, $p(t) \equiv \frac{excess\_load(t)^+}{offered\_load(t)}$
- divisor is significant
  - resource 'calculates' $\boldsymbol{p}$ in bulk and communicates it to each load
  - each load knows its own contribution to load – its own rate, $\boldsymbol{x_i}$
    - so each load can know its own contribution to excess load, $\boldsymbol{px_i}$
- equivalent to
  - probability of loss
  - probability of ECN marking (by redefining 'excess' load)
- probability of loss/marking along path
  - combinatorial probability of loss/marking at each resource along path

$$\boldsymbol{p \equiv 1 - (1 - p_1)(1 - p_2)}$$
$$\boldsymbol{\cong p_1 + p_2} \qquad \boldsymbol{\forall i,\ p_i << 1}$$

fair allocation... of what? among what?

of 'cost' among bits

- cost of one user's behaviour on other users
  - congestion volume ≡ instantaneous congestion p...
  - ...shared proportionately over each user's bit rate, x_i
  - ...over (any) time
  - $v_i \equiv \int p(t)x_i(t)\,dt$
- volume of dropped/marked data each user sent
  - integrates simply and correctly over time and over flows



- example

v_1	= 10% x 200kbs^{-1} x 50ms	+ 10% x 300kbs^{-1} x 150ms		
	= 1kb	+ 4.5kb	=	5.5kb
v_2	= 10% x 300kbs^{-1} x 50ms	+ 10% x 200kbs^{-1} x 150ms		
	= 1.5kb	+ 3kb	=	4.5kb

toy scenario for illustration only; strictly...
- a super-linear marking algorithms to determine p is preferable for control stability
- the scenario assumes we're starting with full buffers

fair allocation... of what?

why cost fairness, not benefit fairness?

- two electricity users
  - one uses a unit of electricity for a hot shower
  - next door the other uses a unit for her toast
- the one who showered enjoyed it more than the toast
  - should she pay more?
- in life, we expect to pay only the cost of commodities
  - a competitive market drives the price to cost (plus ‘reasonable’ profit)
  - if one provider tries to charge above cost, another will undercut
- cost metric is all that is needed technically anyway
  - if operator does charge by value (benefit), they’re selling snake-oil anyway
  - don’t need a snake-oil header field

congestion volume

captures (un)fairness during dynamics

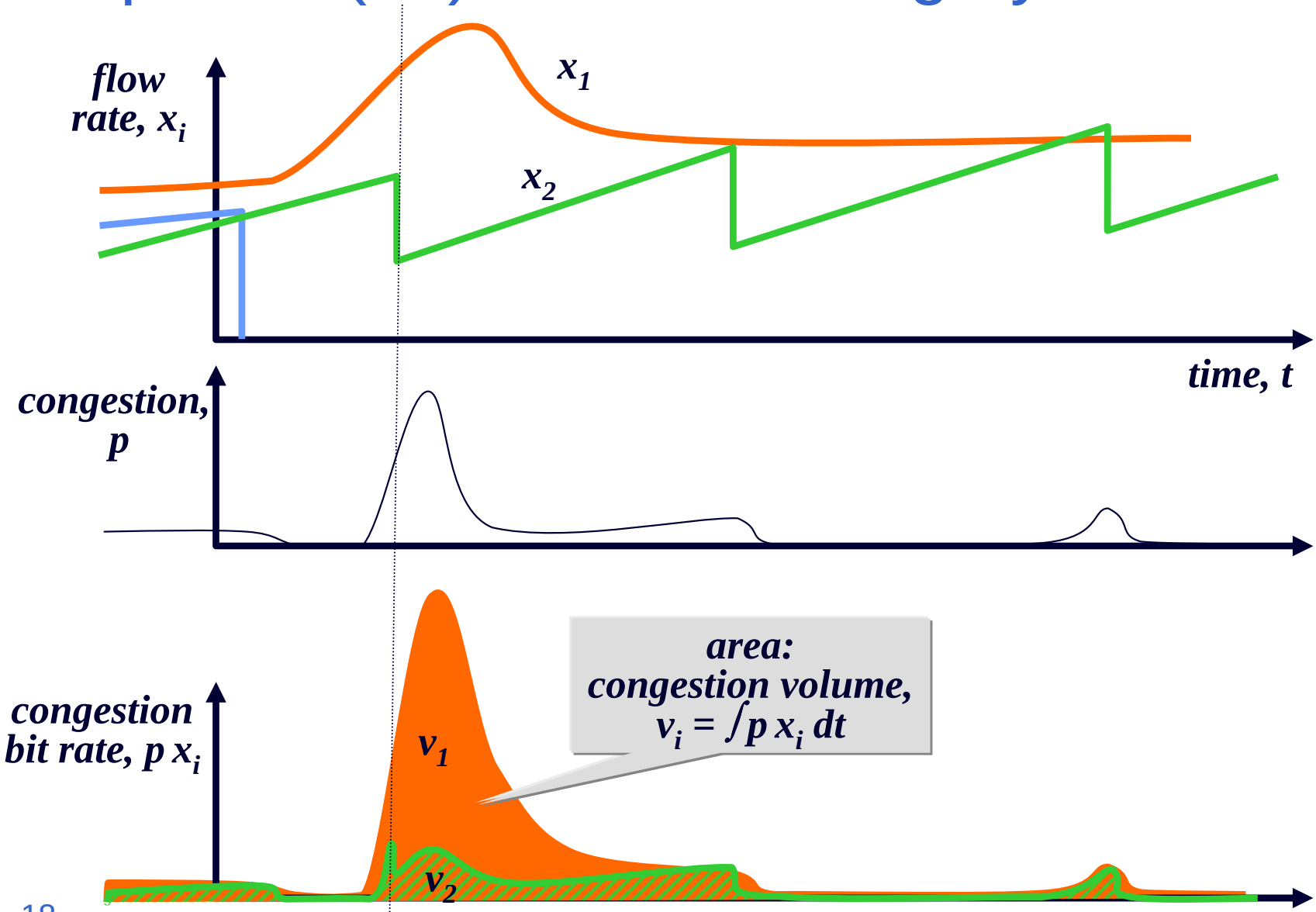
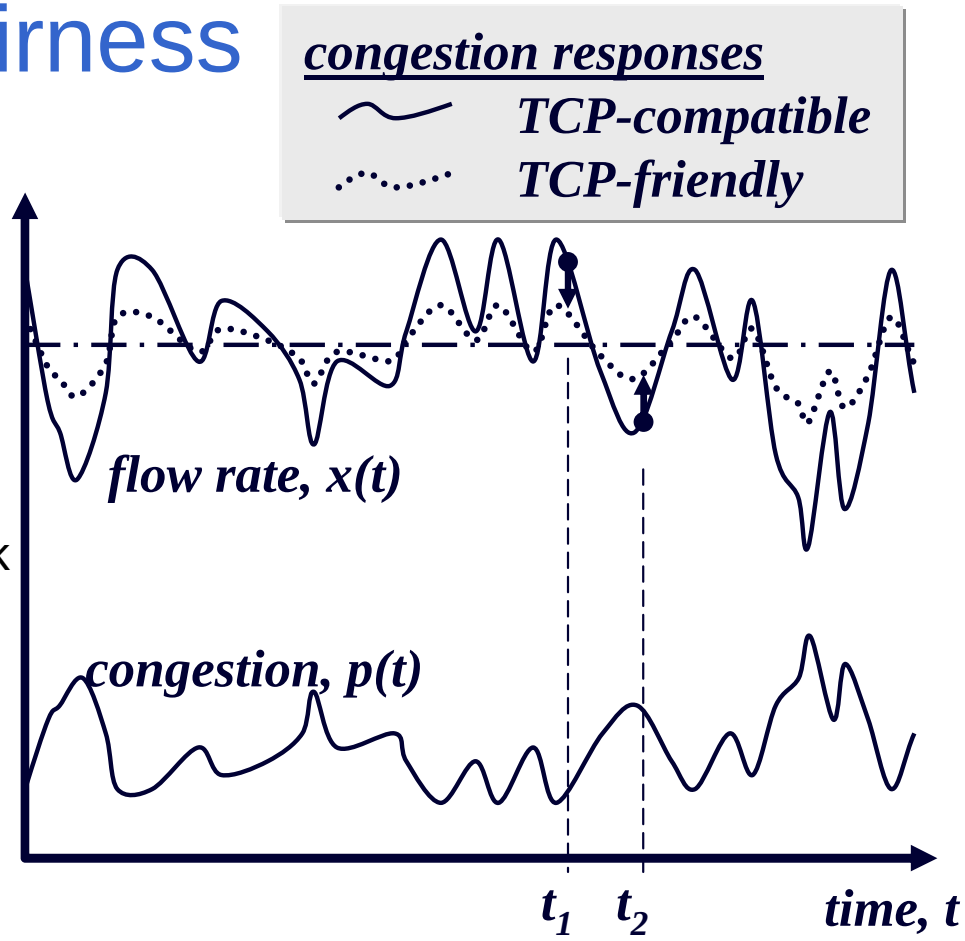


illustration: TCP-friendly rate control (TFRC)

problems with rate fairness

- TCP-friendly
  - same ave rate as TCP
  - congestion response can be more sluggish
- compared to TCP-compatible
  - higher b/w during high congestion
  - lower b/w during low congestion
- giving more during times of plenty doesn't compensate for taking it back during times of scarcity



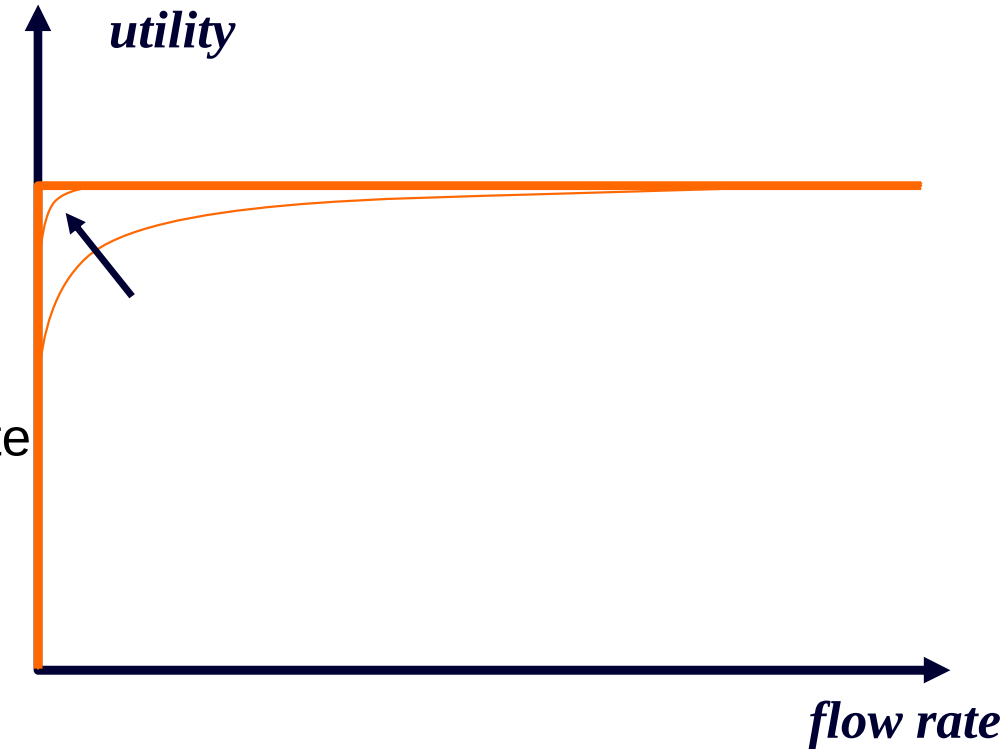
- TCP-friendly flow causes more congestion volume than TCP
- need lower rate if trying to cause same congestion cost

- TFRC vs TCP is a minor unfairness
  - compared to the broken per flow notion common to both

illustration: max-min rate fairness

problems with rate fairness

- max-min rate fairness
  - maximise the minimum share
  - then the next minimum & so on
- if users take account of the congestion they cause to others
- max-min rate fairness would result if all users' valuation of rate were like the sharpest of the set of utility curves shown [Kelly97]
  - they all value high rate exactly the same as each other
  - they all value very low rate just a smidgen less
  - ie, they are virtually indifferent to rate

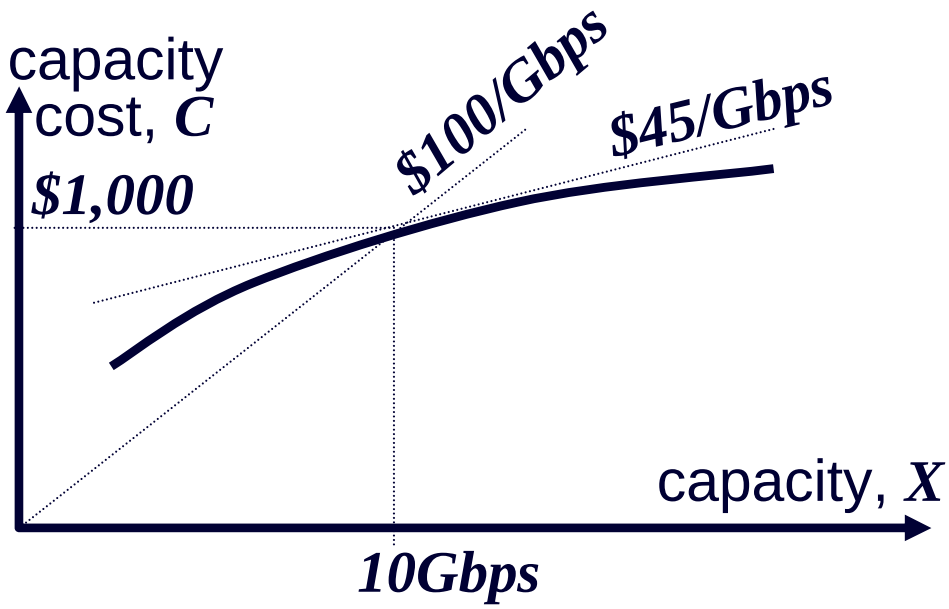


- users aren't that weird

∴ max-min is seriously unrealistic

calibrating ‘cost to other users’

- congestion volume
  1. both a measure of ‘cost to other users’
  2. and a measure of traffic not served
- a monetary value can be put on ‘traffic not served’
  - the marginal cost $\partial C/\partial X$ of upgrading the network equipment
    - so that it wouldn’t have dropped (or marked) the volume it did
- cost of 2. tends to 1.
  - in a competitive market
  - or some other welfare maximising ‘invisible hand’

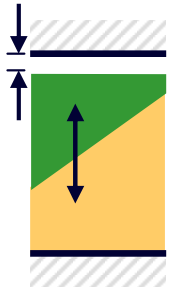


- example of one interface card
  - ***variable*** usage cost = $ 45/Gbps
  - balance of capacity = $ 55/Gbps
  - fixed capacity cost = $100/Gbps
  - fixed operational costs + whatever

next steps

who should decide what fairness to have?

- certainly not the IETF
- fairness nothing to do with functioning of network
  - there will always be *an* allocation
  - any allocation ‘works’
  - can alter fairness independently of utilisation
    - XCP, opening multiple TCPs



- a socio-economic requirement on engineering
- candidates
  - governments
  - network owner (e.g. military, university, private, commercial)
  - market
- should be able to do all the above
  - IETF skill should be to ‘design for tussle’ [Clark, 2002]
  - basis of the design of re-ECN

- currently the IETF does decide
  - based on an unsubstantiated notion of fairness between flow rates
    - which has no basis in real life, social science, philosophy or anything
  - this view isn’t even complete enough to be a form of fairness

next steps
aim, fire, ready

2. need *to be able* to make senders accountable' for congestion caused
   - accountable to whom?
     - the network(s) in which they are causing congestion
     - in practice: structure accountability through attached neighbours?
     - networks need to see reliable congestion information
   - 'accountable' doesn't mean 'pay for'
     - it can mean 'limit cost within the flat rate already paid'
     - it can also mean 'with a lot of give and take'
3. need weighting parameter added to transport APIs (cf MulTCP)
1. transition from what we have now?
   - we have absolutely no fairness, so there's nothing to transition from
   - but there is a danger of getting it *more* wrong than we have already
   - therefore MUST do step 2 before 3
   - hi-speed congestion ctrl in progress should be designed *as if* we have 2
     - voluntary cost fairness (cf. voluntary TCP fairness)

re-ECN

next step towards architectural change

- re-ECN: a change to IP <draft-briscoe-tsvwg-re-ecn-tcp-03>
  - evolutionary pressure on transports
  - **IP sender** has to mark at least as much congestion as emerges at the receiver
  - **networks** can use these markings to gradually tighten fairness controls
    - spectrum from tight to none
  - weighted **sender transports** evolve
  - **receiver transports** evolve that can negotiate weighting with sender
- propose to use last reserved bit in IPv4 header
- in return re-ECN enables
  - fairness
  - choice of fairness regimes
  - robustness against cheating
  - incremental deployment with strong deployment incentives
  - a natural mitigation of DDoS flooding
  - differentiated QoS
  - safe / fair evolution of new cc algs
    - DCCP, hi-speed cc etc.
  - policing TCP's congestion response for those hooked on per flow fairness

re-ECN IETF internet draft roadmap

Re-ECN: Adding Accountability for Causing Congestion to TCP/IP
draft-briscoe-tsvwg-re-ecn-tcp-03
intent
§3: overview in TCP/IP
§4: in TCP & other transports } *stds*
§5: in IP (v4 & v6)
§6: accountability apps *inform'l*

Emulating Border Flow Policing using Re-ECN on Bulk Data
draft-briscoe-tsvwg-re-ecn-border-cheat-02
intent: informational

RSVP Extensions
for Admission Control over Diffserv
using Pre-congestion Notification
draft-lefaucheur-rsvp-ecn-01
intent
adds congestion f/b to RSVP *stds*