policing congestion response in an internetwork using re-feedback

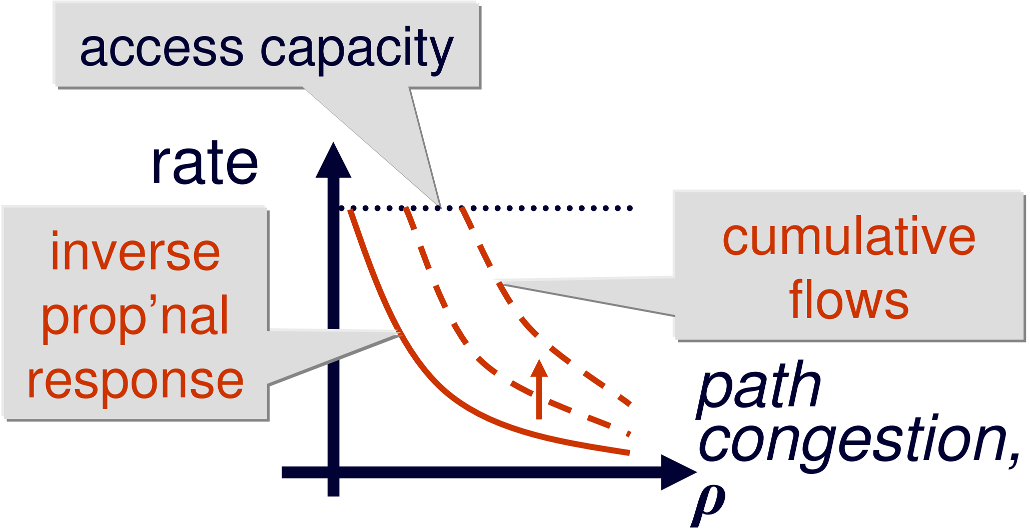
Bob Briscoe[1,2]

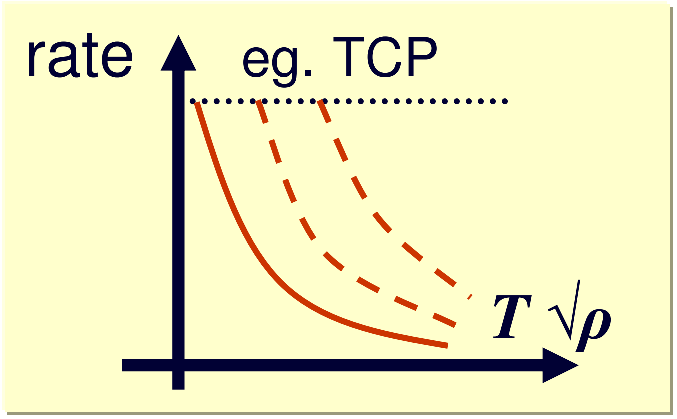
Arnaud Jacquet[1], Carla Di Cairano-Gilfedder[1], Alessandro Salvatori[1,3], Andrea Soppera[1] & Martin Koyabe[1]

[1]BT Research, [2]UCL, [3]Eurécom

the problem: policing congestion response

- host response to congestion: voluntary
- short and long term congestion
  - short: policing TCP-friendliness (or any agreed response)
  - long: policing file-sharing (selfish), zombie hosts (malicious/careless)





- network policing users' congestion response: voluntary
  - a network doesn't care if users cause congestion in other networks

very serious problem

- a few unresponsive (UDP) flows wasn't a problem
- converged IP network
  - initially ~30-50% of bits inelastic (mostly voice), for BT
  - internetwork similar
- can't police required response to *path* congestion, if you don't know it
  - each element only sees *local* congestion
  - network can't reliably see e2e feedback (IPsec encryption, lying, route asymmetry)
- can't hope inelastic apps *ask* to be unresponsive (Diffserv/signalling)
  - because those that don't ask can free-ride anyway
  - due to lack of evidence of their 'crime'
- capacity investment risk unacceptable if can't prevent free-riding
- uncontrollable demand dynamics *and* suppressed incentive to supply
  - risk of repeated congestion collapse (alarmist?)

previous work

- detect high *absolute* rate [commercial boxes]
- sampled rate response to *local* congestion [RED + sin bin]
- transport control *embedded in* network [ATM]
- *honest* senders police feedback from rcvrs [ECN nonce]

wouldn't it be nice if...

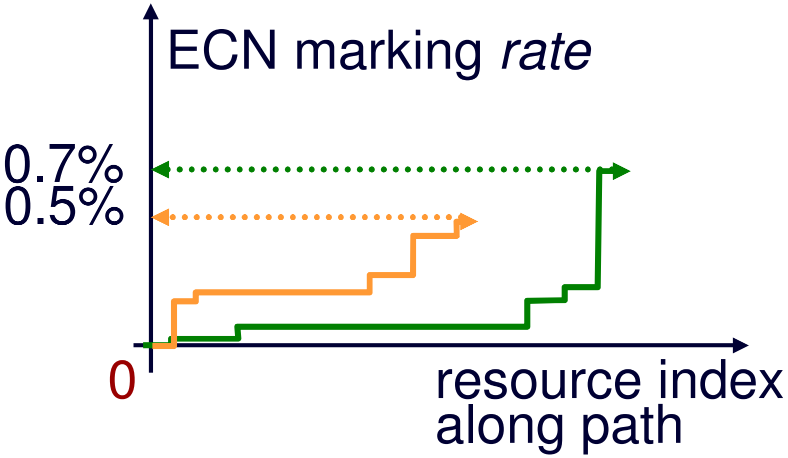
- source declared downstream path characteristics to network
- everyone was truthful:
  - endpoints and networks
- deployment could be incremental
- we could solve more general Internet Architecture problems
  - capacity allocn & accountability [NewArch]

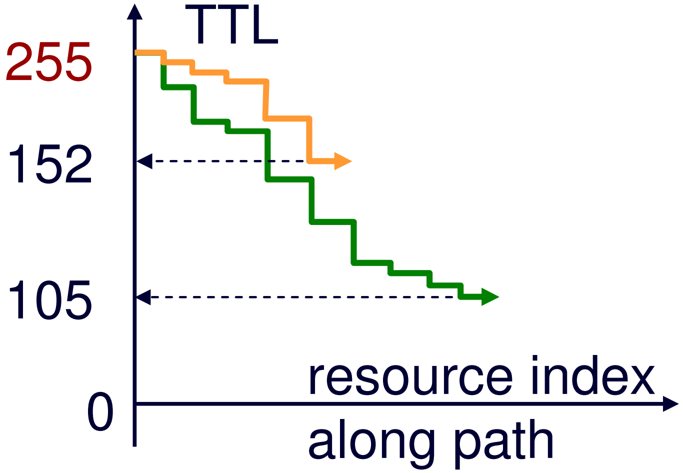
...we can: our approach

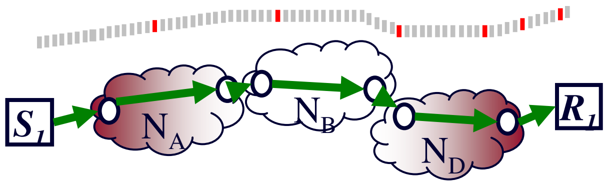
- the big idea #1
  - then 2 sub-ideas based on...
- network economics & incentives
  - rational networks (not users)
  - no fiddling with user pricing
  - challenge: break and improve
- incremental deployment idea #4
  - around unmodified IP routers
  - BUT limited header bits slows attack detection *considerably*
- generalisations
  - QoS
  - DoS mitigation
  - flow start incentives
  - inter-domain traffic engineering
  - non-IP internetworks

path characterisation via data headers

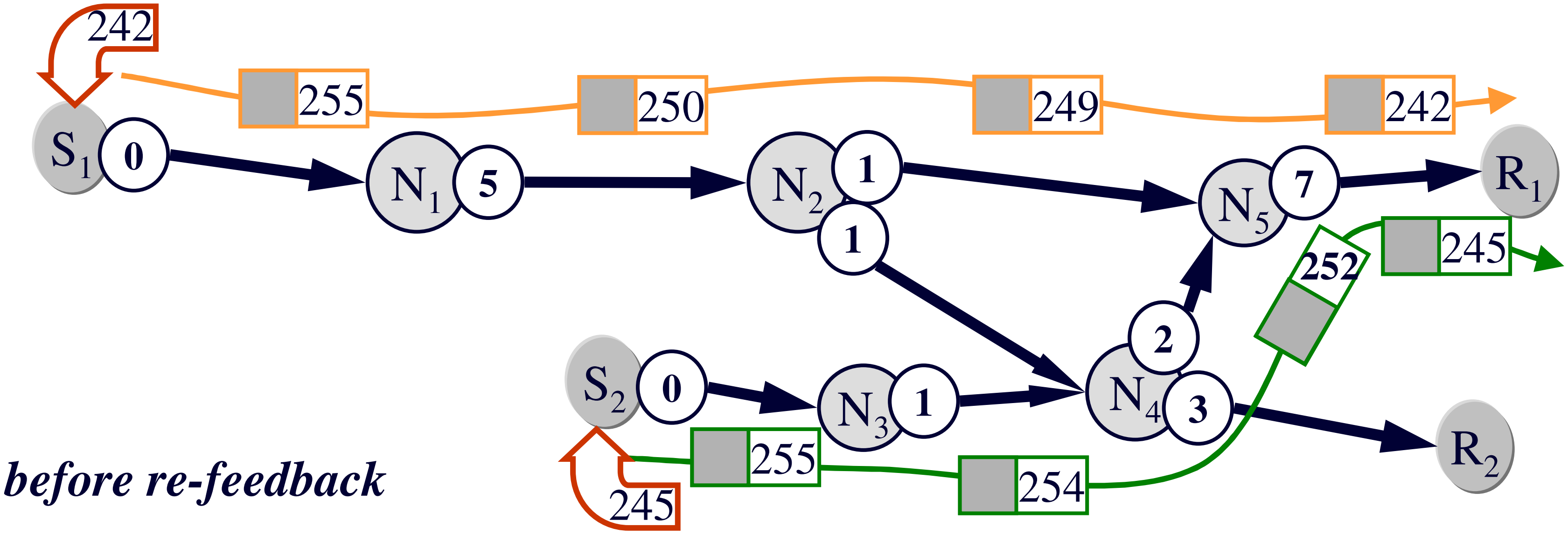
state of the art



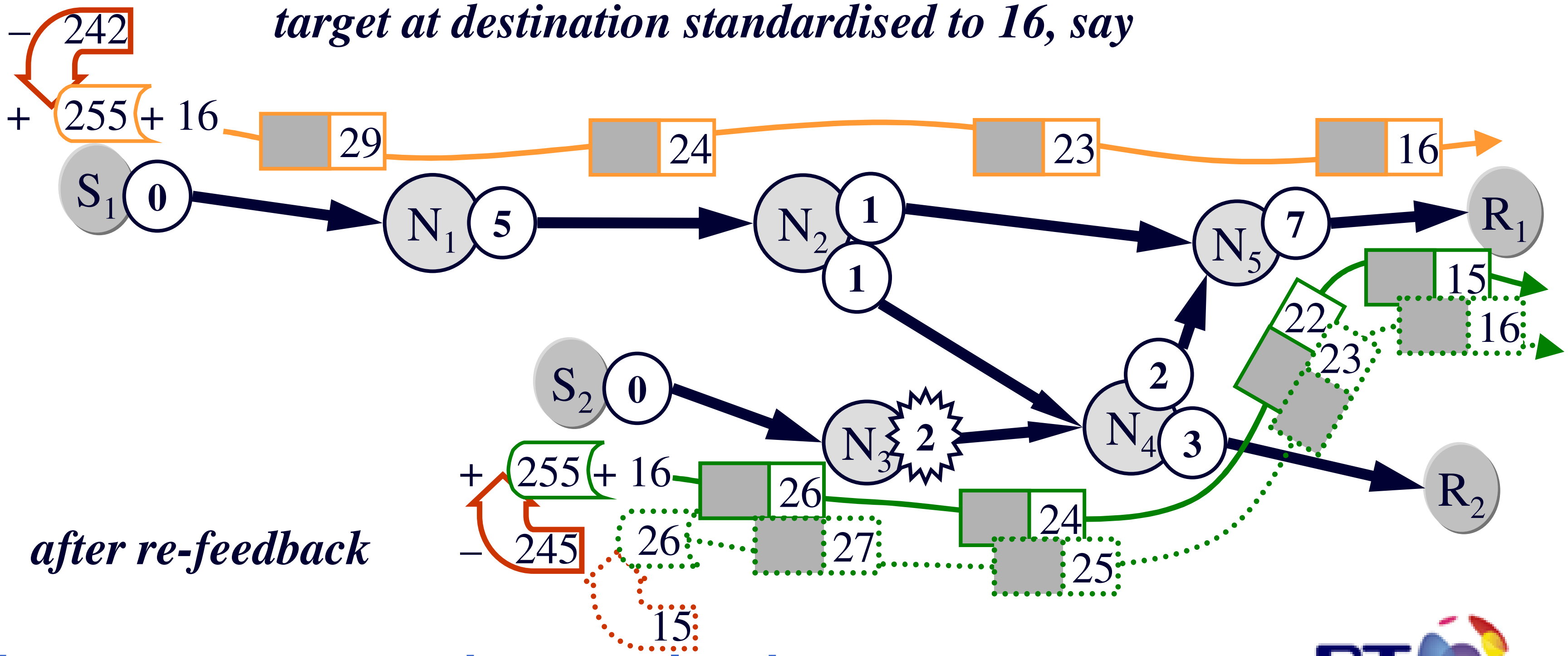




downstream knowledge upstream



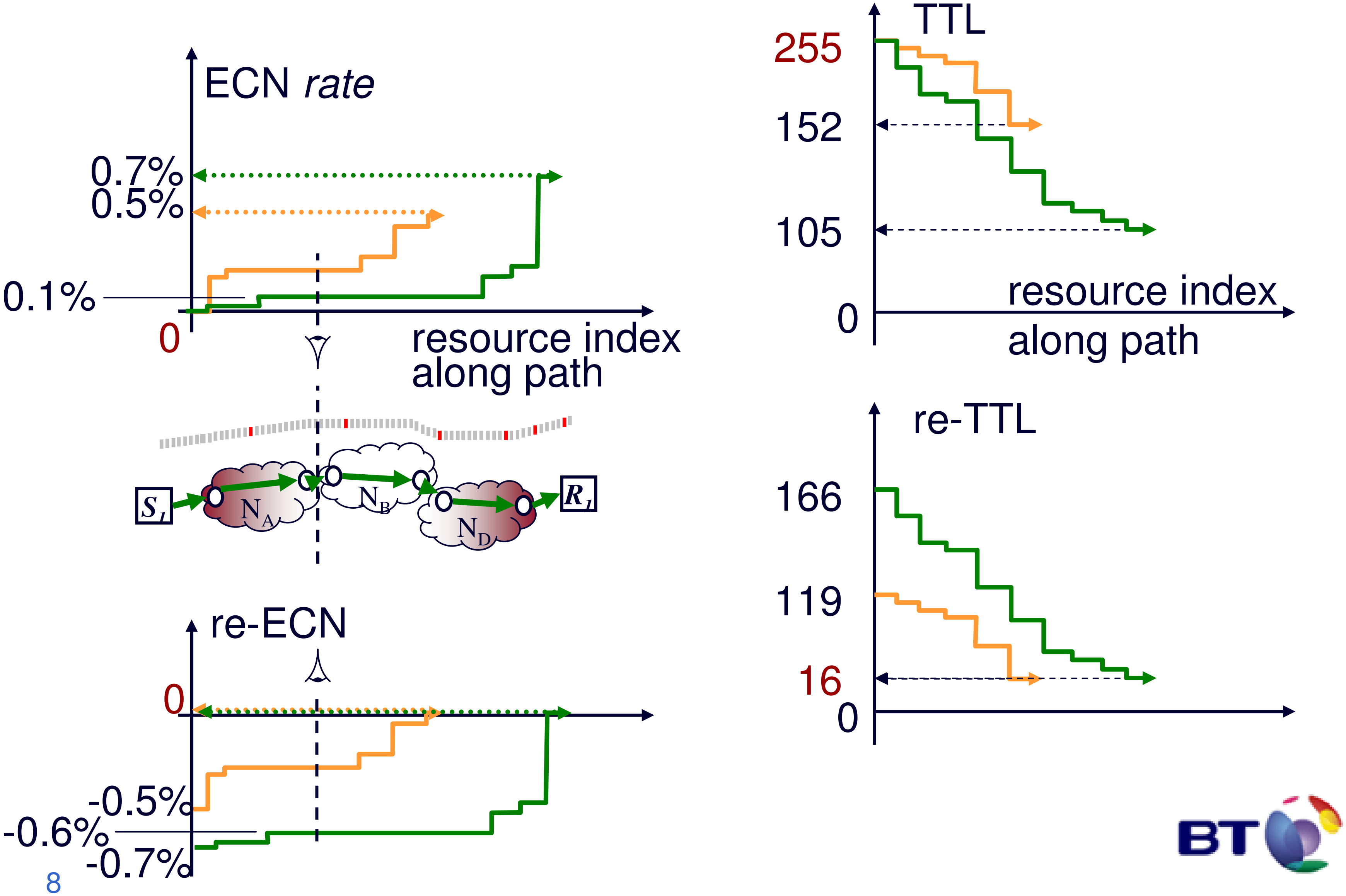
before re-feedback



target at destination standardised to 16, say

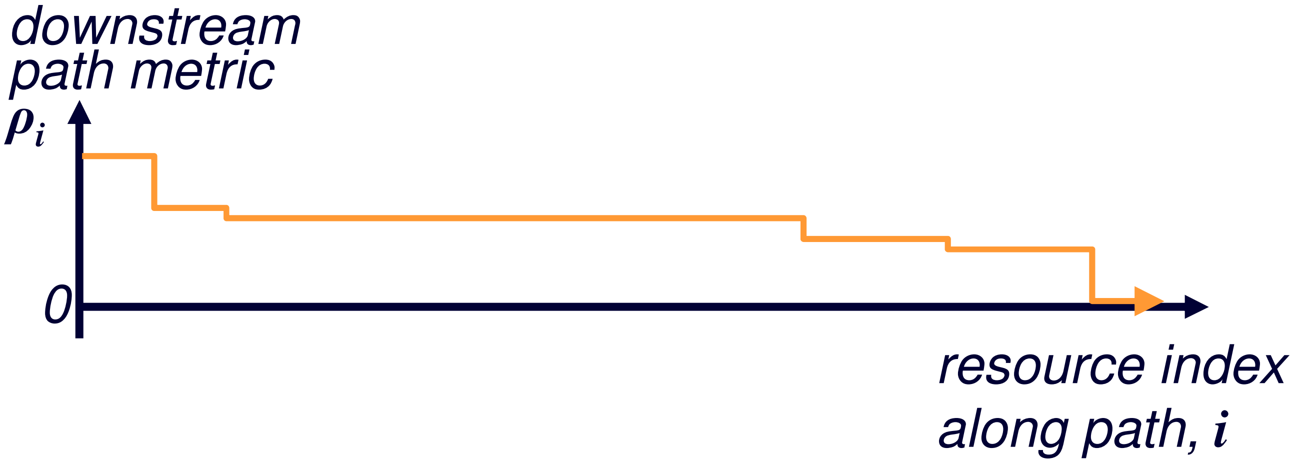
after re-feedback

downstream path characterisation

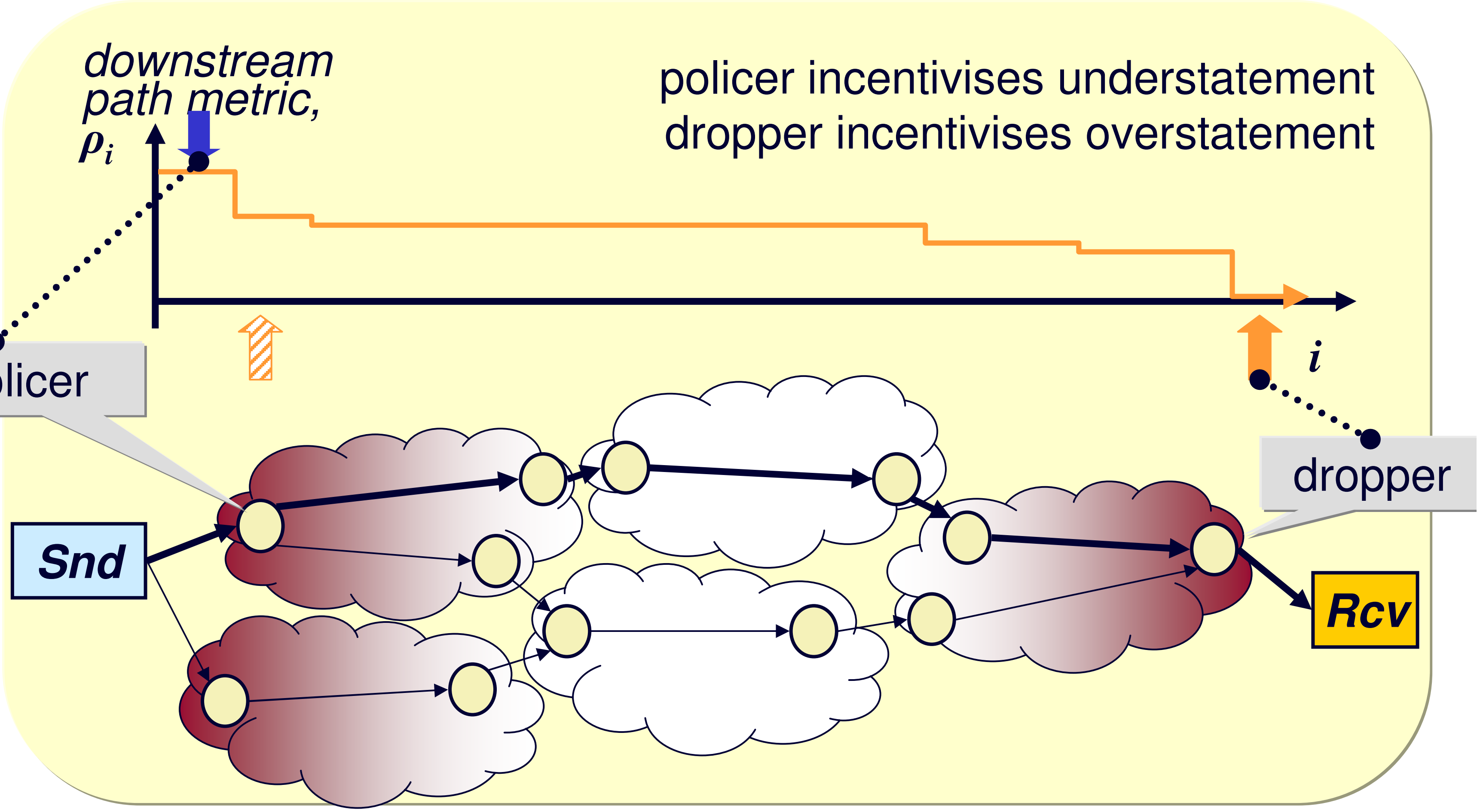


incentives: preamble

- so far, policing relies on self-incrimination?...
- focus initially on congestion
  - header processing not just additive/subtractive
  - generalises to monotonic functions (eg combinatorial probability of ECN marking)
  - downstream unloaded delay (~TTL/2) has identical incentive properties
- to aid understanding
  - solely graphical visualisation (see paper for maths)
  - imagine that header carries a real number
  - normalise: monotonically decreasing to target at zero

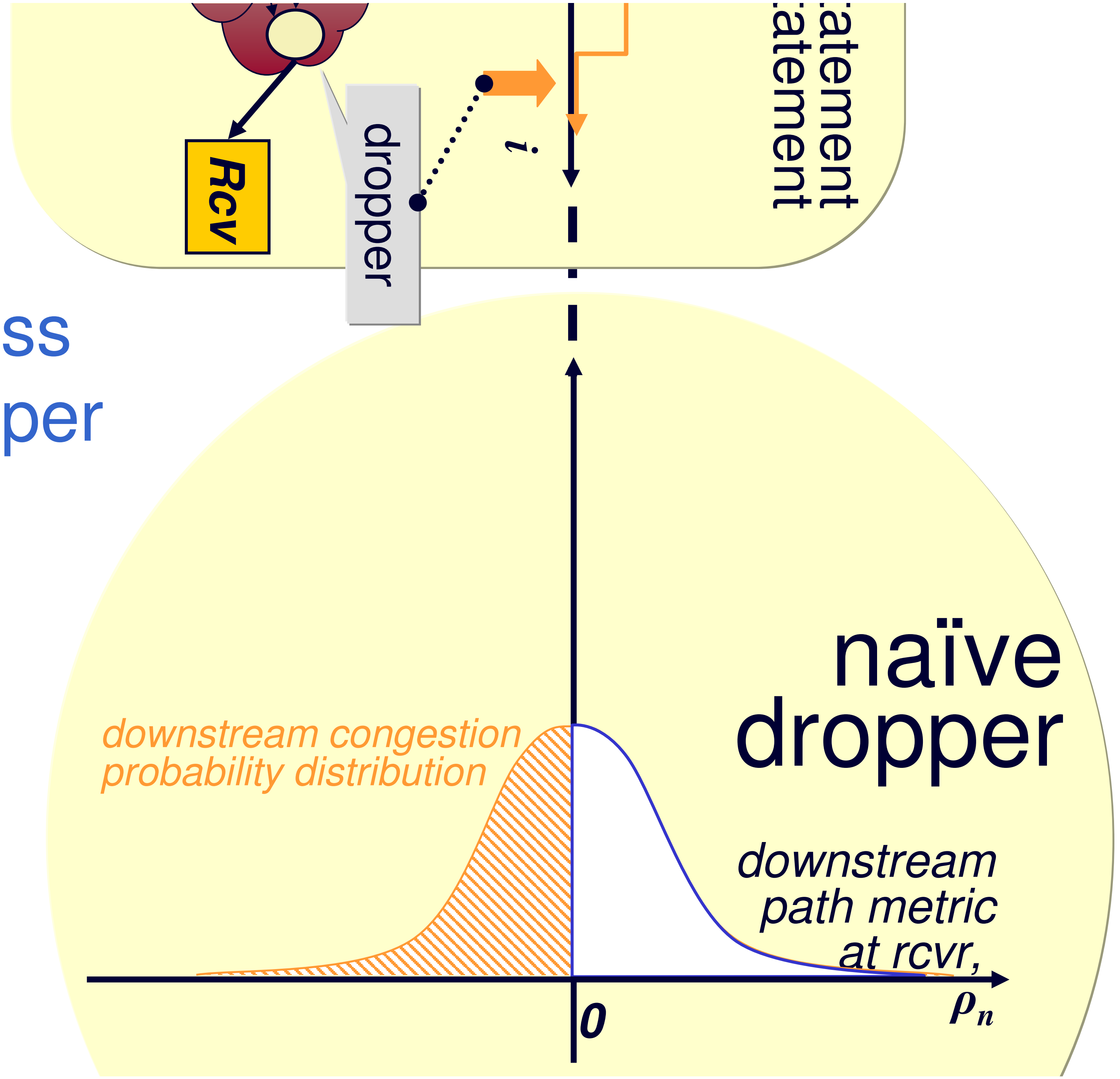


incentive framework: user-network



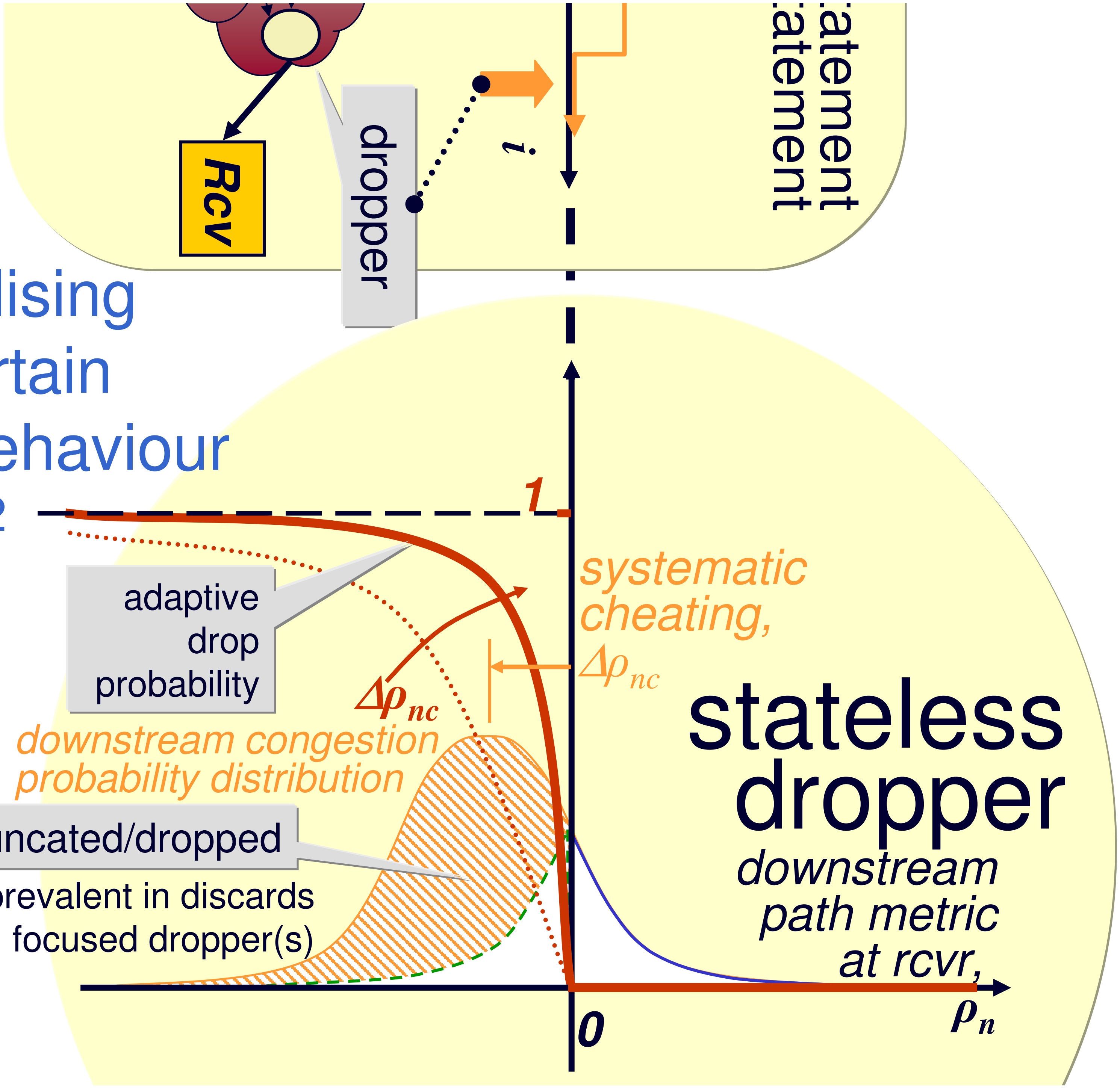
policer incentivises understatement
dropper incentivises overstatement

egress dropper


Rcv
dropper
i
atement
atement
downstream congestion probability distribution
naïve dropper
downstream path metric at rcvr,
0
ρ_n

penalising uncertain misbehaviour

idea #2



adaptive drop probability

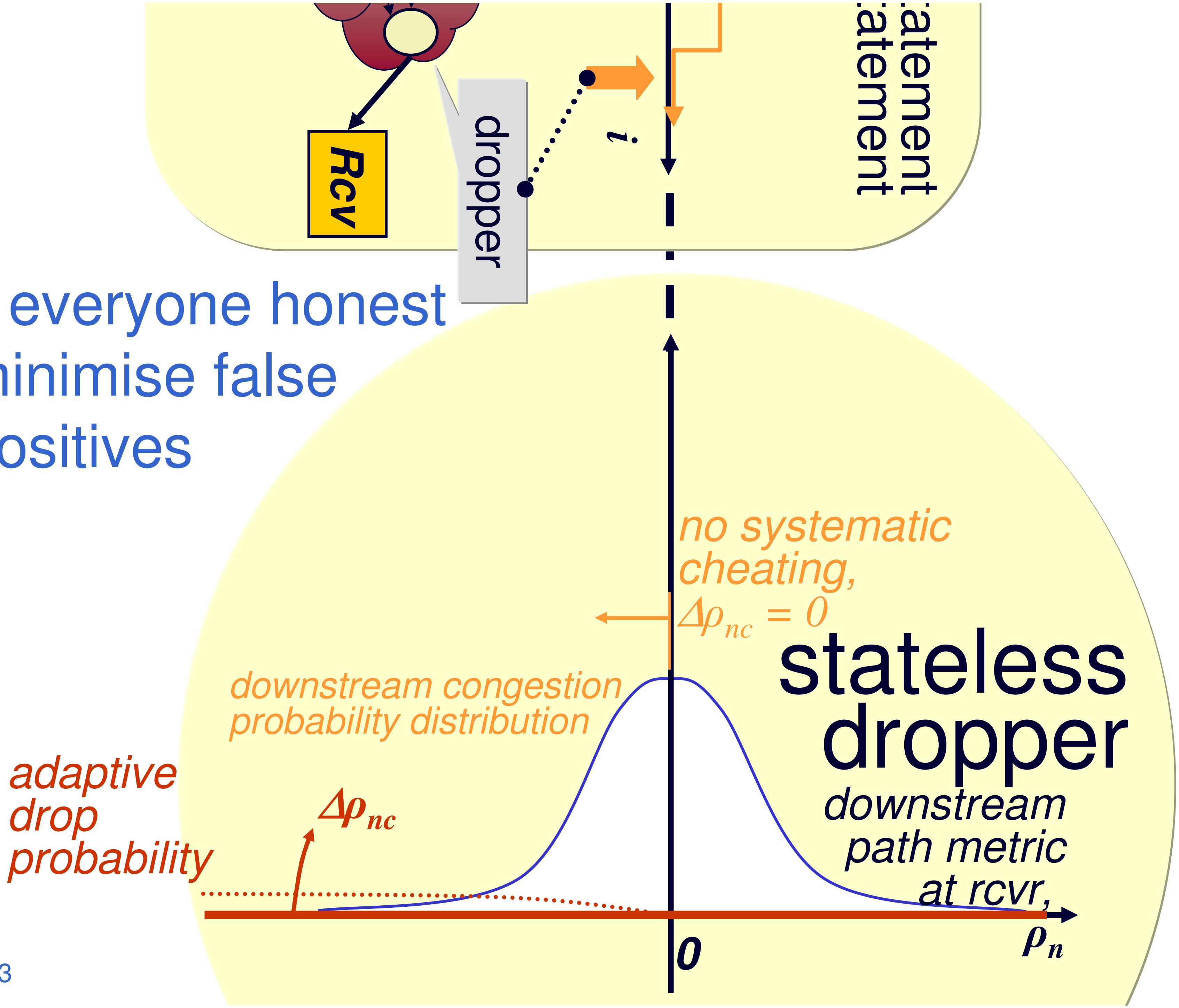
truncated/dropped

if signature prevalent in discards
spawn focused dropper(s)

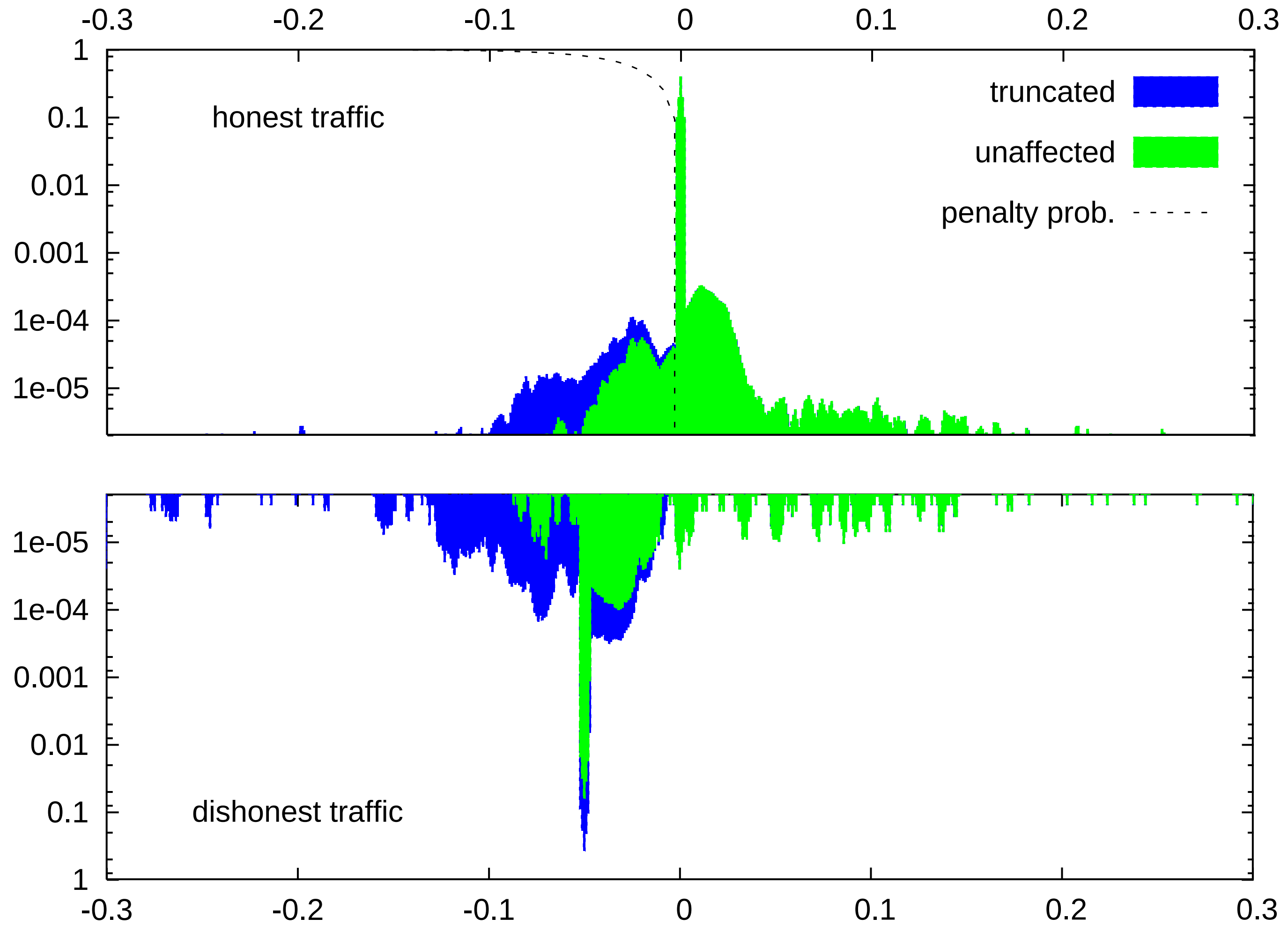
stateless dropper

downstream path metric at rcvr,

if everyone honest minimise false positives

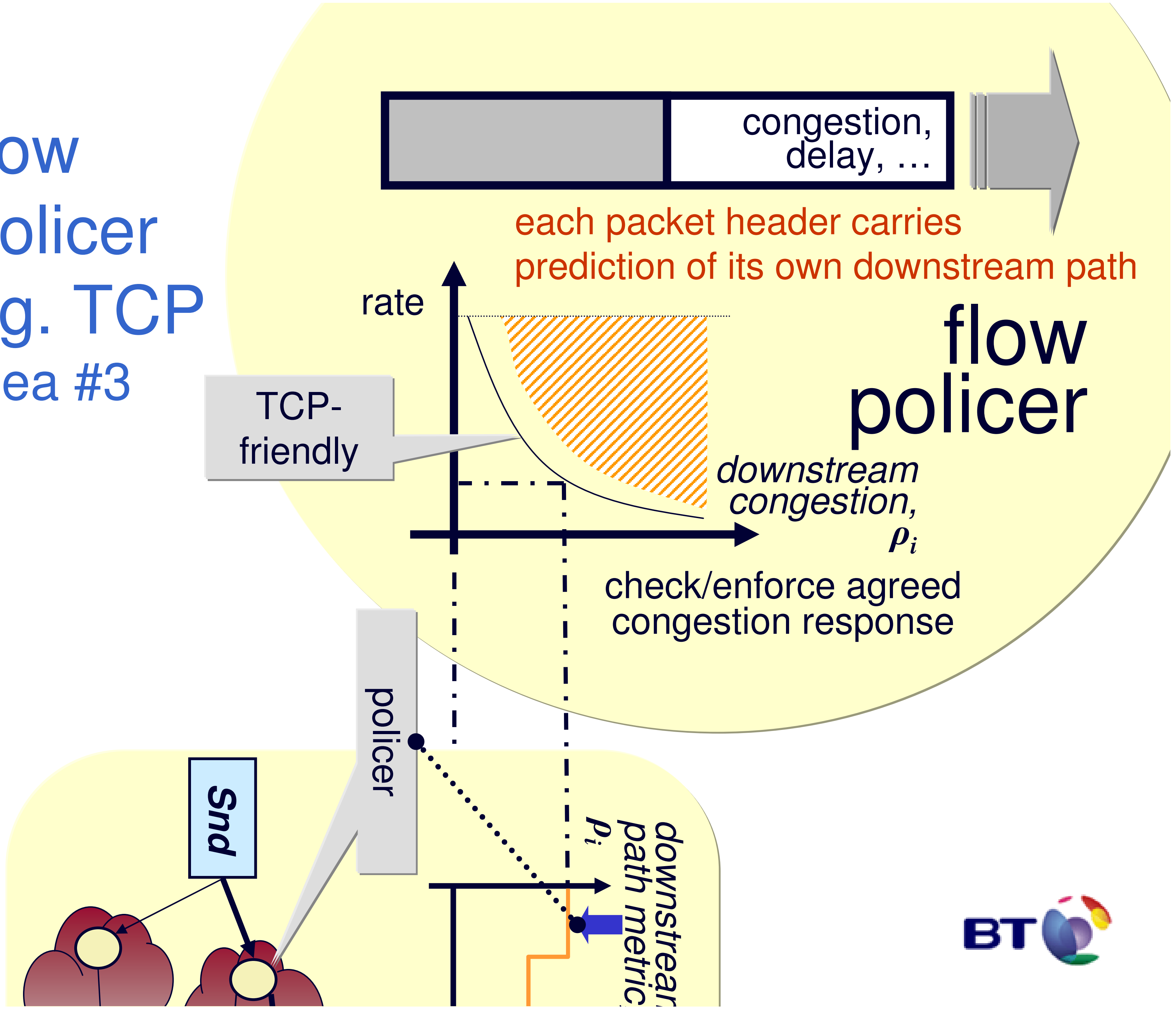


typical dropper simulation (note log scale)



flow policer eg. TCP

idea #3



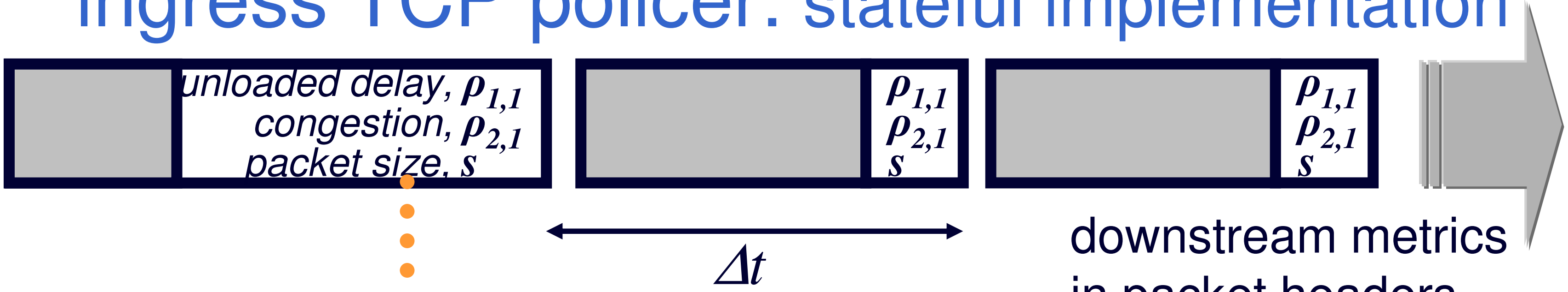
each packet header carries prediction of its own downstream path

check/enforce agreed congestion response

intro
the idea
incentives
deployment
generalise

also bounded flow state policer implemented - using sampling

ingress TCP policer: stateful implementation



downstream metrics
in packet headers
at internetwork ingress

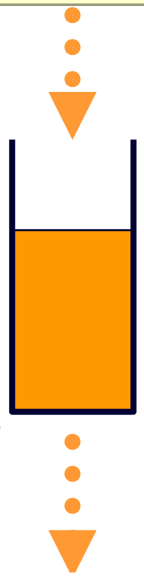
path congestion ≈ *downstr congestion*

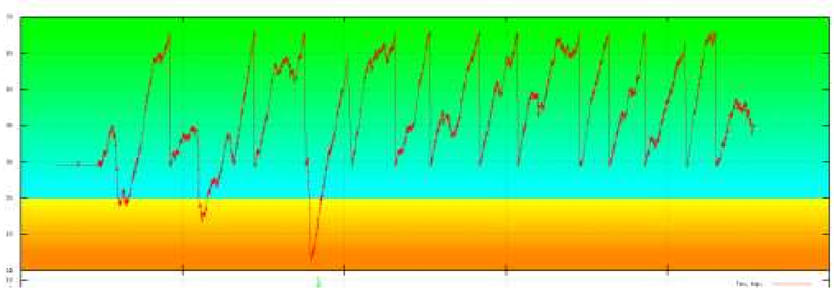
$p \approx \rho_{2,1}$

path RTT ≈ *upstr RTT* + 2 * *downstr delay*

$T \approx T_0 + 2\,\rho_{1,1}$

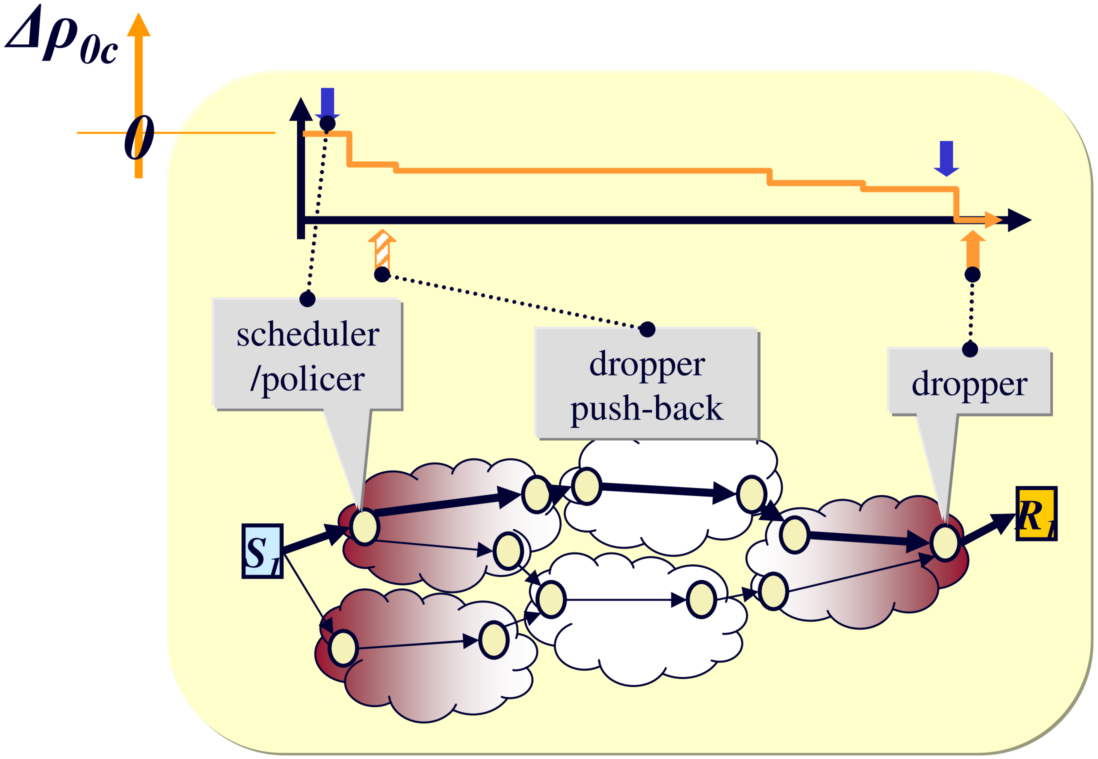
$$x_{TCP} \approx \frac{s}{T}\sqrt{\frac{3}{2p}}$$

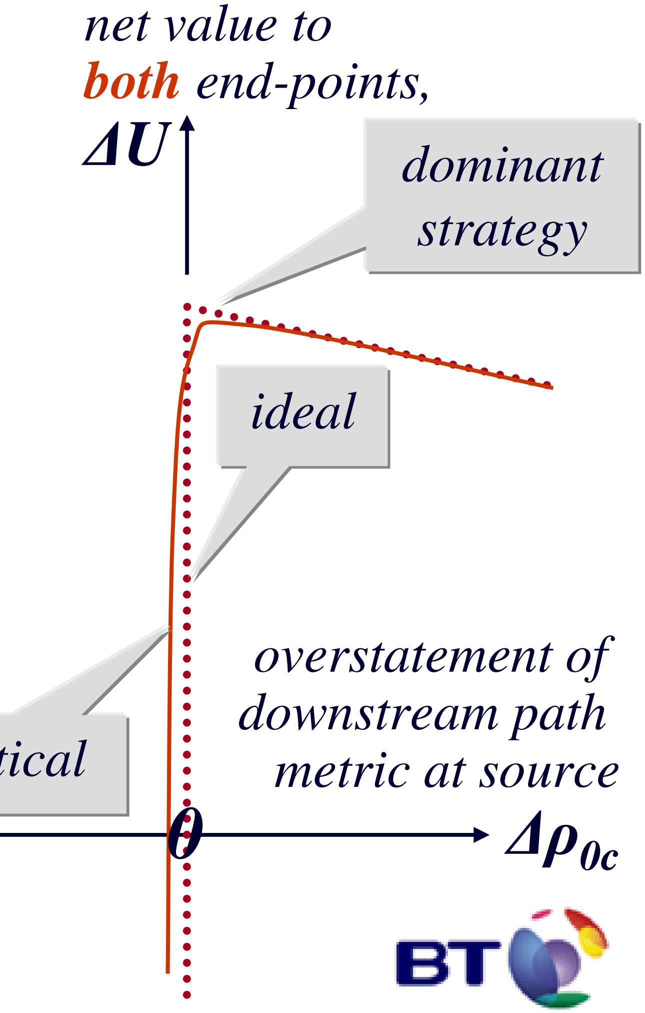




$$x = s/\Delta t$$

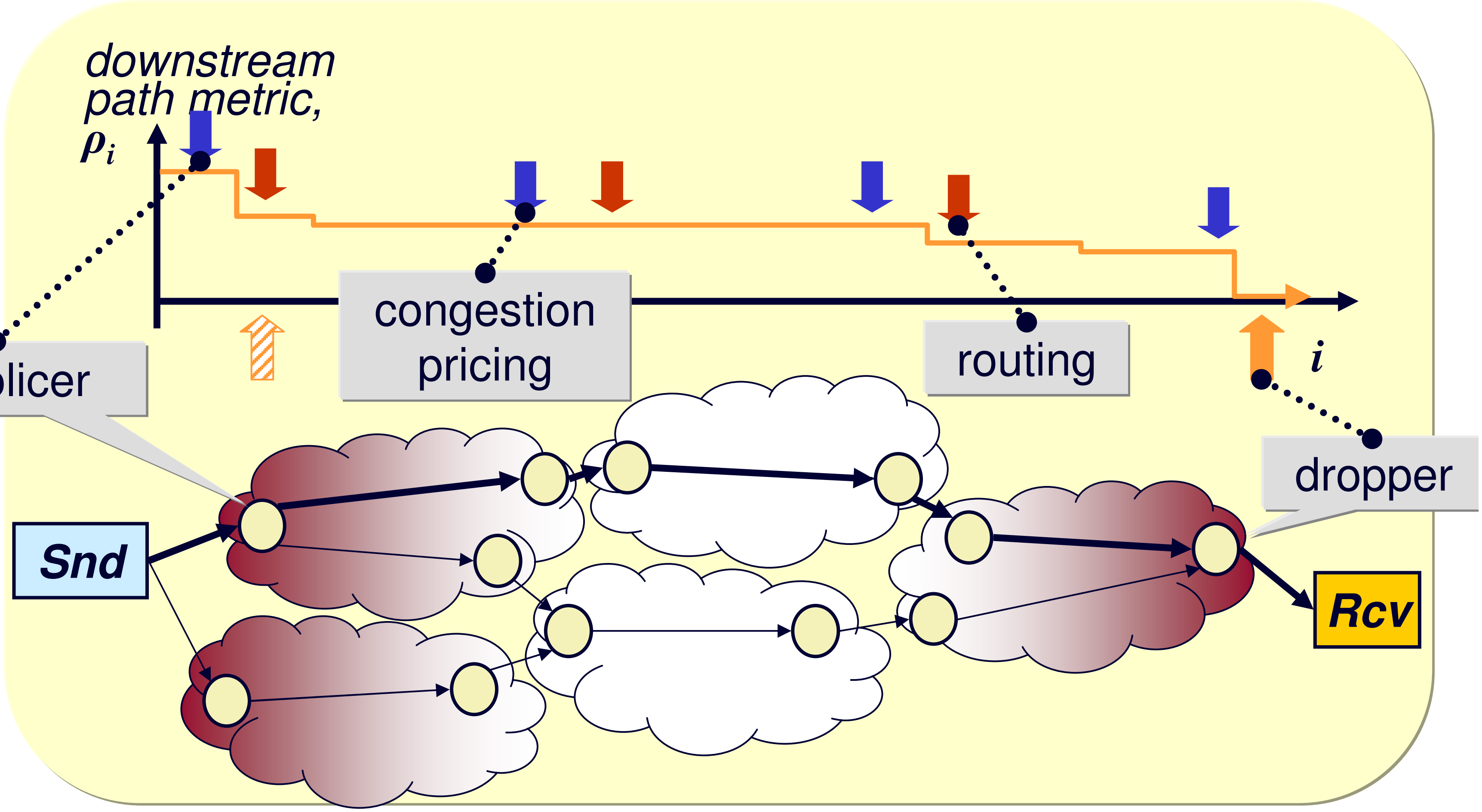
incentive compatibility – hosts





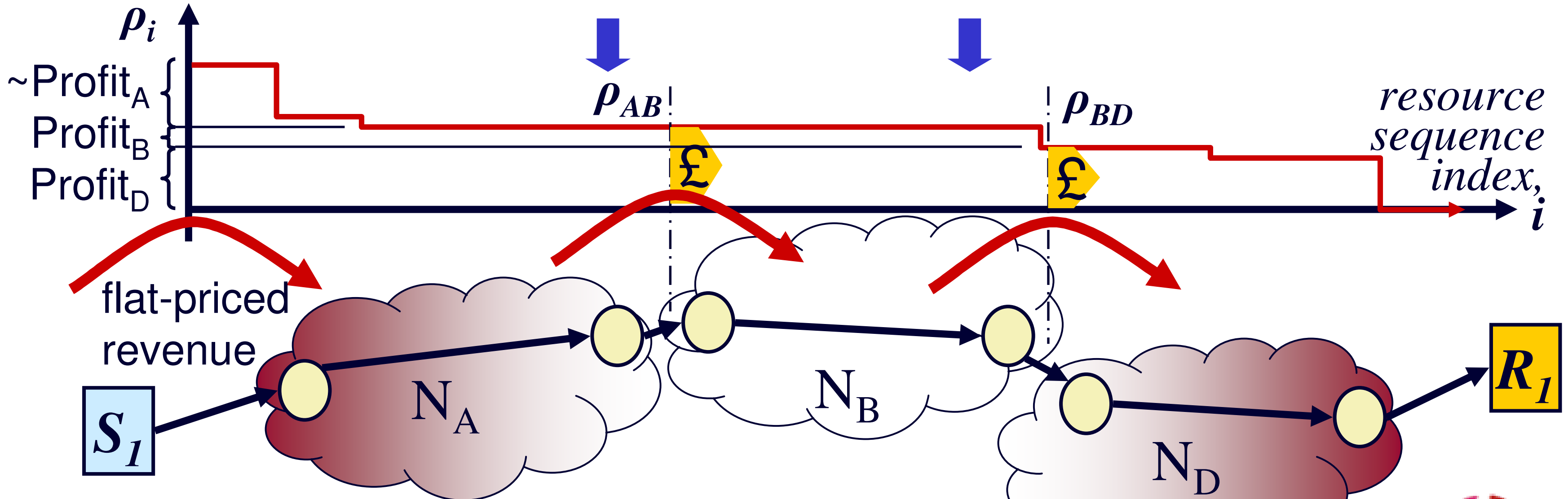
- incentivise:
  - responsible actions
  - honest words

incentive framework



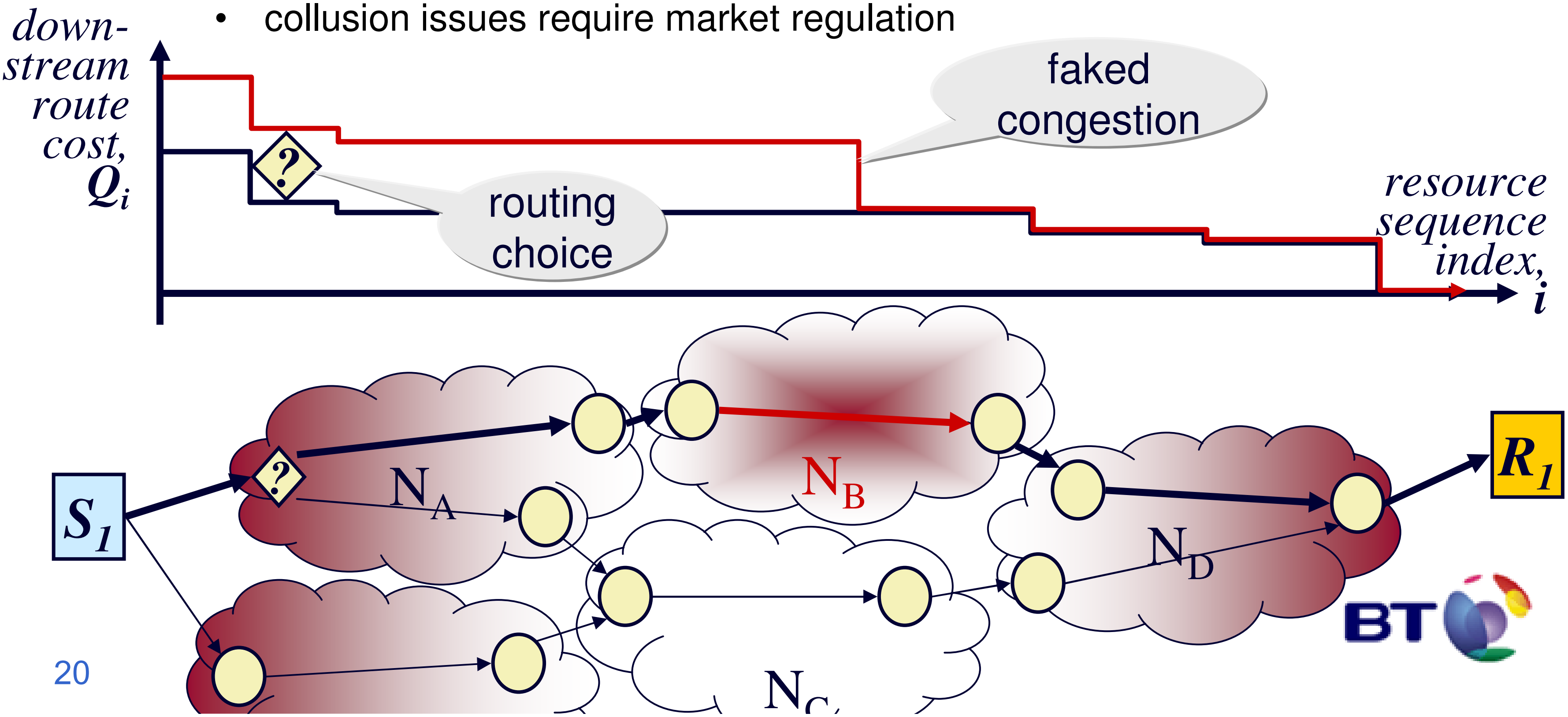
incentives for networks to police their users

- ρ_i is size of each packet factored by its downstream congestion metric
- metered between domains by single bulk counter
- automagically shares congestion revenue across domains, and within domains to direct upgrades
- can approximate congestion pricing with SLAs



congestion competition – inter-domain routing

- if congestion → profit for a network, why not fake it?
  - upstream networks will route round more highly congested paths
  - N_A can see relative costs of paths to R_1 thru N_B & N_C
- the issue of monopoly paths
  - incentivise new provision
  - collusion issues require market regulation

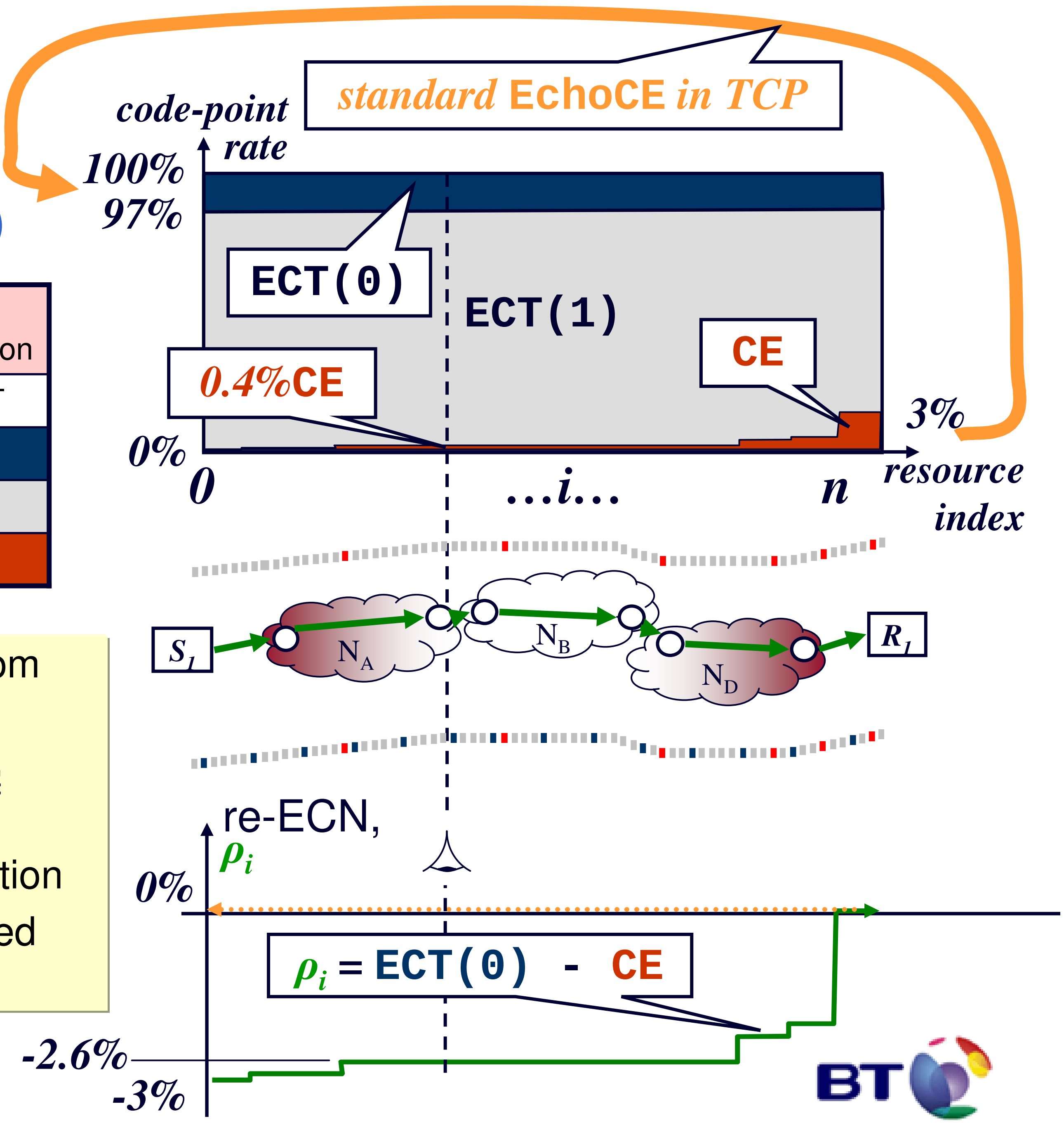


re-ECN

(sketch idea #4)

code-point	standard designation
00	not-ECT
10	ECT(0)
01	ECT(1)
11	CE

- on every EchoCE from TCP, set ECT(0)
- at any point on path, diff between rates of ECT(0) & CE is downstream congestion
- works with unchanged routers

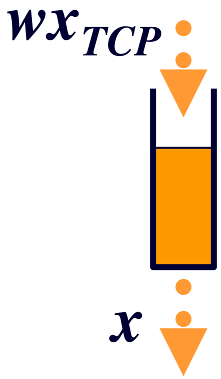


ρ_i = ECT(0) - CE

deployment incentives

- re-ECN deployment by incremental sender upgrades
  - re-TTL can be hacked for legacy receivers
- deploy policers and droppers permissively config'd
  - allows new & legacy behaviours to co-exist
- incrementally increase strictness
  - throttles legacy stacks: upgrade incentive knob
- beware: slow to catch cheaters with one bit re-ECN

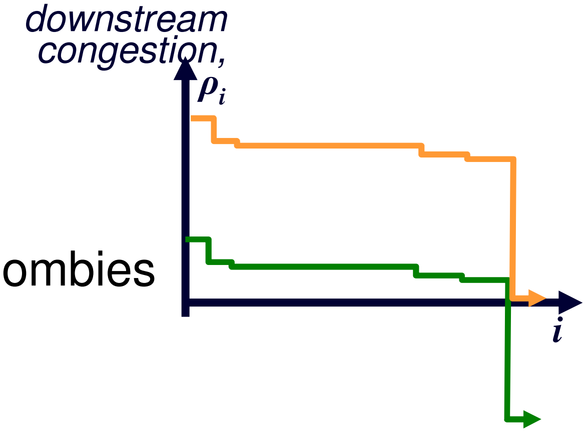
edge QoS = our original motivation



- once timely truthful path visible...
- ingress network can allow spectrum of responses to incipient congestion (*w*-weighted policer)
    - equivalent* to offering differentiated QoS (*caveat: see paper)
    - like [Kelly98] but without the need for congestion pricing of users
- purely by local (sender↔ingress) arrangement
    - no authorisation on any other network elements (equal marking)
    - would need suitable back-pressure – e.g. higher flat fee
- other networks reimbursed automagically
    - by inter-domain congestion pricing (SLA model also possible)

no time for… (see paper)

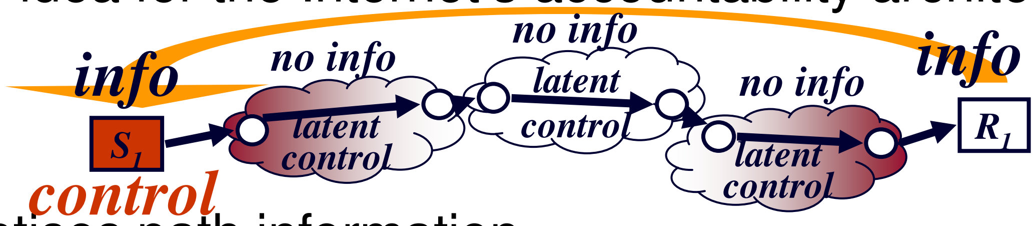
- long term per-user policing (complements per-flow)
  - throttles down sources of persistent long term congestion
  - encourages p2p file-sharing apps to avoid peaks & fill troughs
- DDoS mitigation
  - extreme downstream congestion prompts extreme policing at all ingresses
  - long term per-user policing throttles out zombies



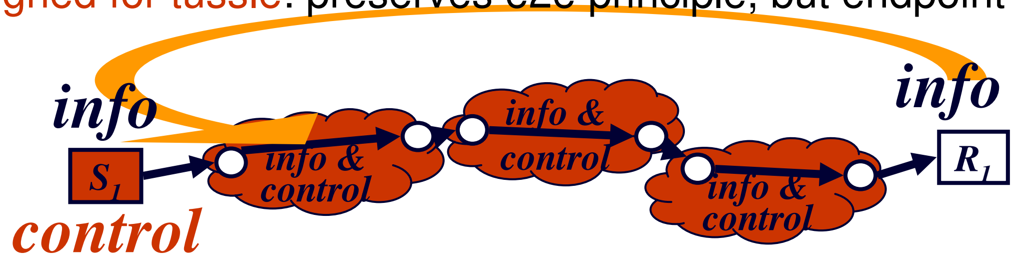
- flow-start incentives
  - deliberate dilemma: downstream metric during flow start?
  - creates slow-start incentive

re-feedback summary

- reinsert feedback to align path characterisations at receiver
- packets arrive at each router predicting downstream path
- arranged for dominant strategy of all parties to be honesty
- incremental deployment + upgrade incentive knob
- hangs new capabilities on ECN deployment, not just performance
- a simple idea for the Internet's accountability architecture



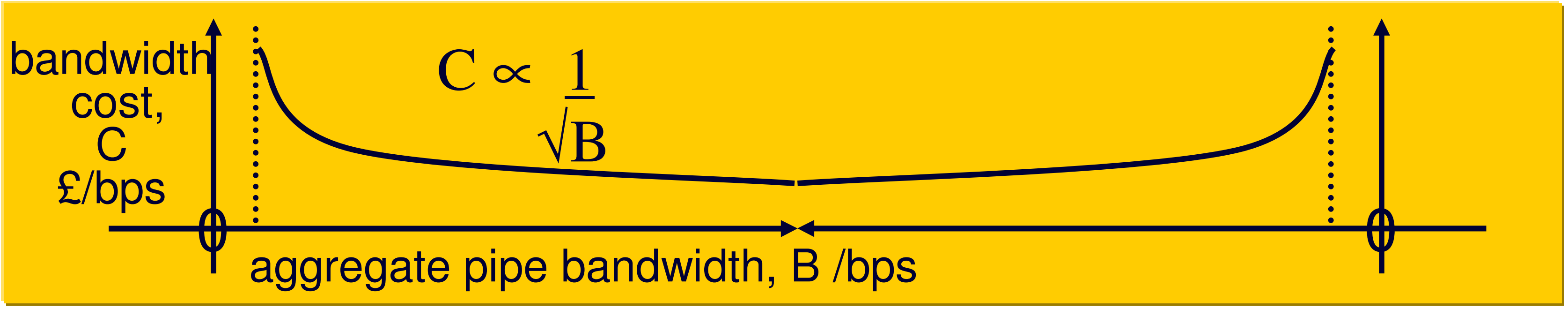
- democratises path information
  - either network or source can control (control requires timely information)
  - designed for tussle: preserves e2e principle, but endpoint control optional

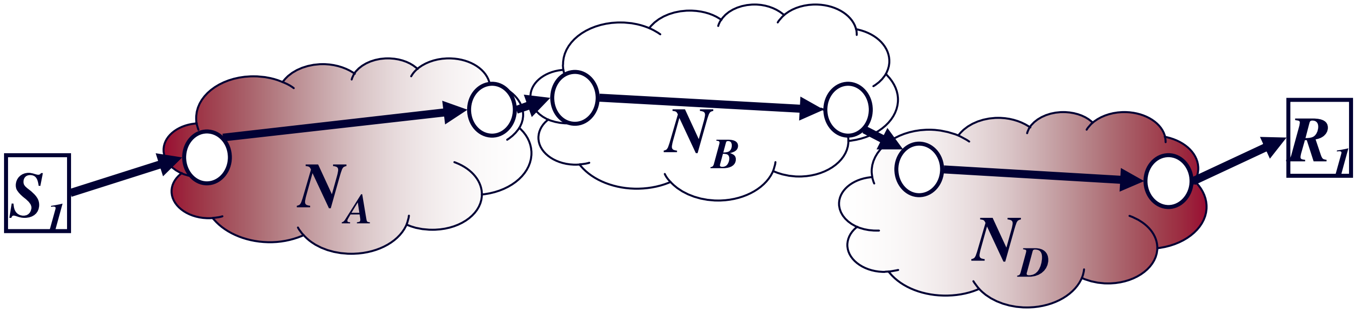


policing congestion response in an internetwork using re-feedback

Q&A

path congestion typically at both edges



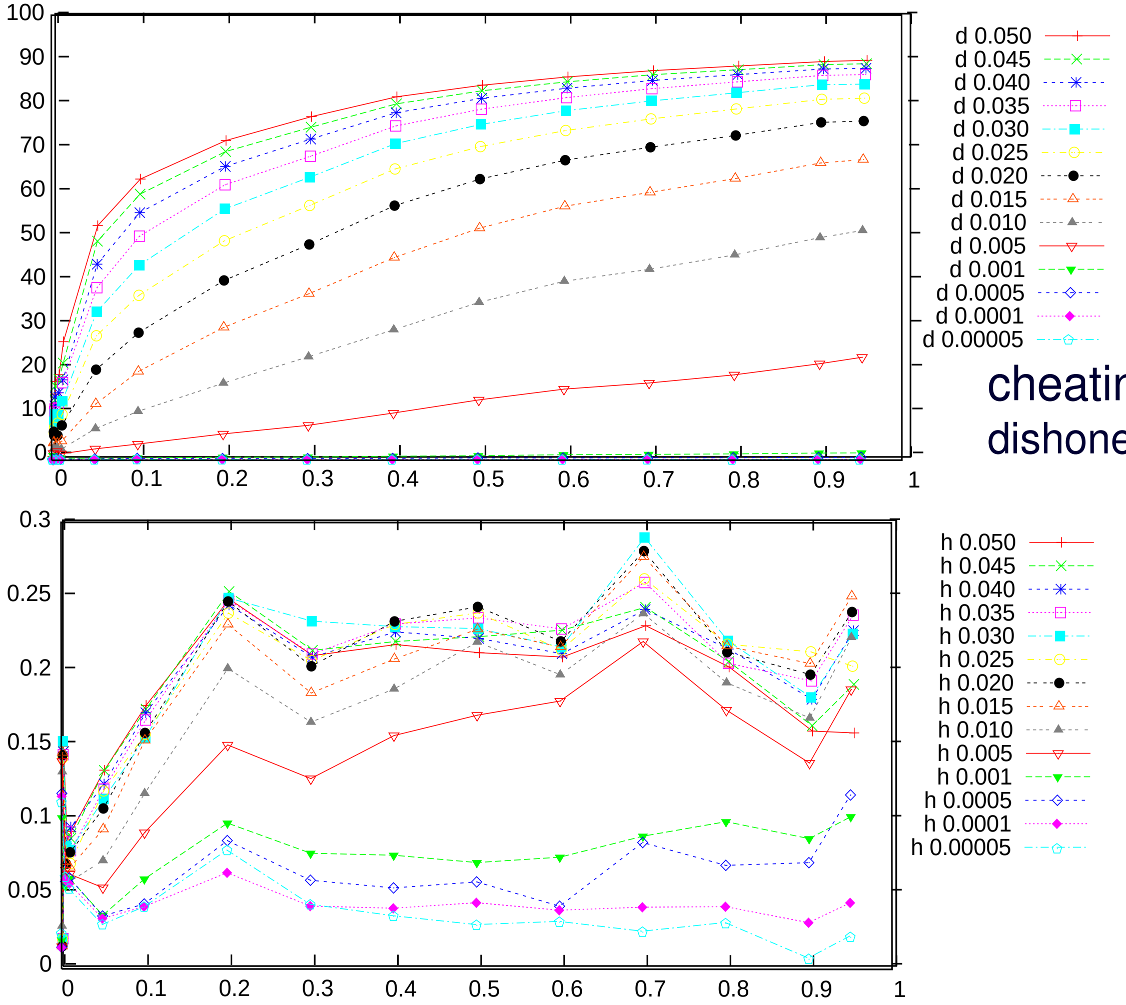


- congestion risk highest in access nets
  - cost economics of fan-out
- but small risk in cores/backbones
  - failures, anomalous demand

last hop dropper: discrimination sensitivity

true positives
truncation rate of
dishonest traffic

false positives
truncation rate of
honest traffic



cheating level of
dishonest sources

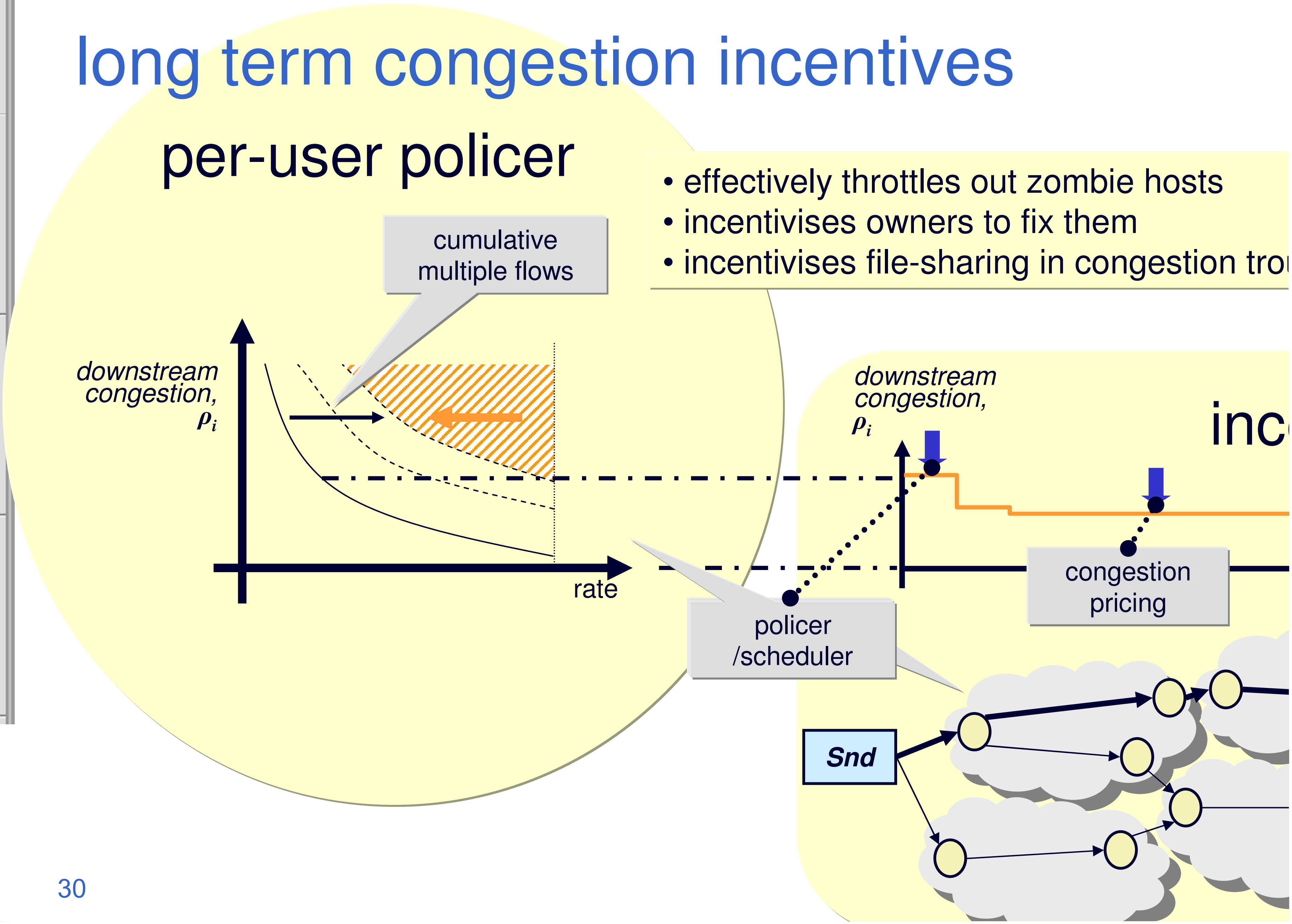
fraction of dishonest arrivals

spawning focused droppers

- use sin-bin technique [Floyd99]
  - examine (candidate) discards for any signature
  - spawn child dropper to focus on subset that matches signature
  - kill child dropper if no longer dropping (after random wait)
- push back
  - send hint upstream defining signature(s)
  - if (any) upstream node has idle processing resource

    test hint by spawning dropper focused on signature as above
- cannot DoS with hints, as optional & testable
  - no need for crypto authentication – no additional DoS vulnerability

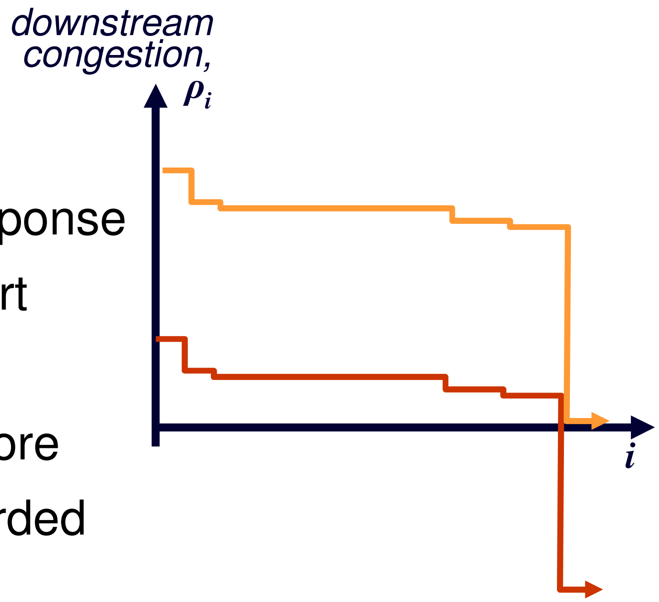
long term congestion incentives


per-user policer
cumulative multiple flows
downstream congestion, ρ_i
rate
downstream congestion, ρ_i
inc
policer /scheduler
congestion pricing
Snd

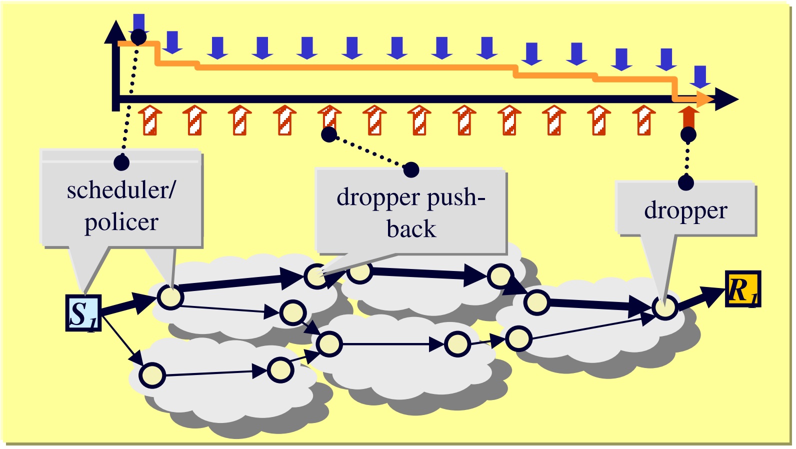
- effectively throttles out zombie hosts
- incentivises owners to fix them
- incentivises file-sharing in congestion tro

distributed denial of service

- merely enforcing congestion response
- honest sources
  - increase initial metric & reduce rate
- malicious sources
  - if do increase initial metric
    - policer at attacker’s ingress forces rate response
    - have to space out packets even at flow start
  - if don’t increase initial metric
    - negative either at the point of attack or before
    - distinguished from honest traffic and discarded
    - push back kicks in if persistent



slow-enough-start



- initial value of metric(s)
  for new flows?
  - undefined – deliberately creates dilemma
  - if too low, may be dropped at egress
  - if too high, may be deprioritised at ingress
- without re-feedback (today)
  - if congested: all other flows share cost equally with new flow
  - if not congested: new flow rewarded with full rate
- with re-feedback
  - risk from lack of path knowledge carried solely by new flow
  - creates slow-start incentive
  - once path characterised, can rise directly to appropriate rate
  - also creates incentive to share path knowledge
  - can insure against the risk (see differentiated service)