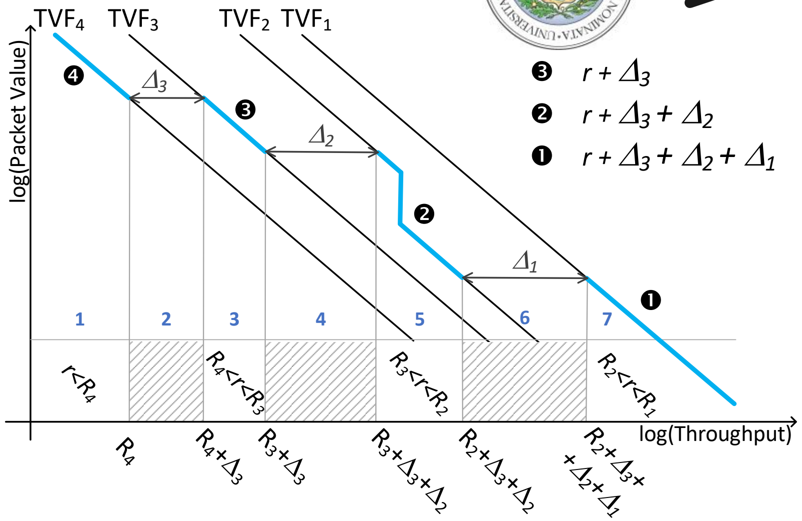
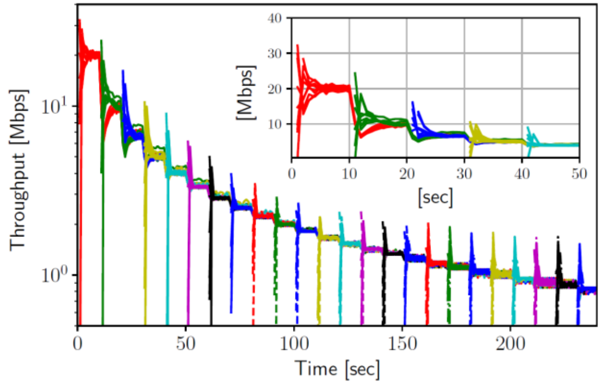
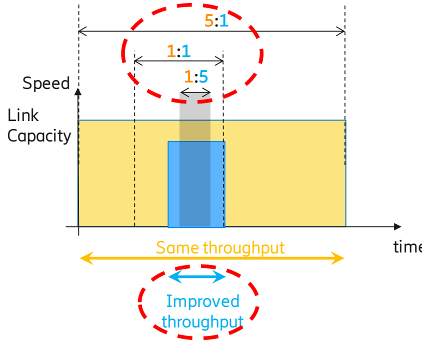
Towards Core-Stateless Fairness on Multiple Timescales

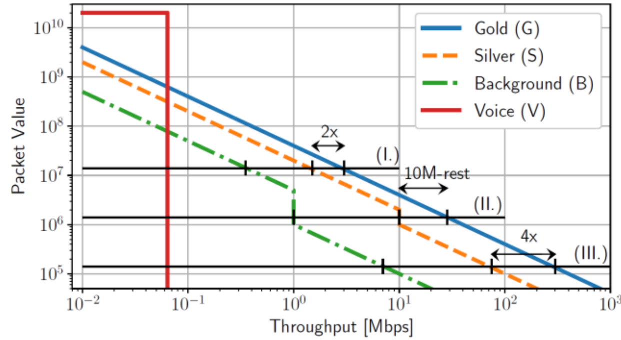




Szilveszter Nádas (Ericsson Research)
Gergő Gombos, Ferenc Fejes and Sándor Laki (ELTE Eötvös Loránd University)

szilveszter.nadas@ericsson.com
http://ppv.elte.hu





Goal: Extend fairness to multiple timescales

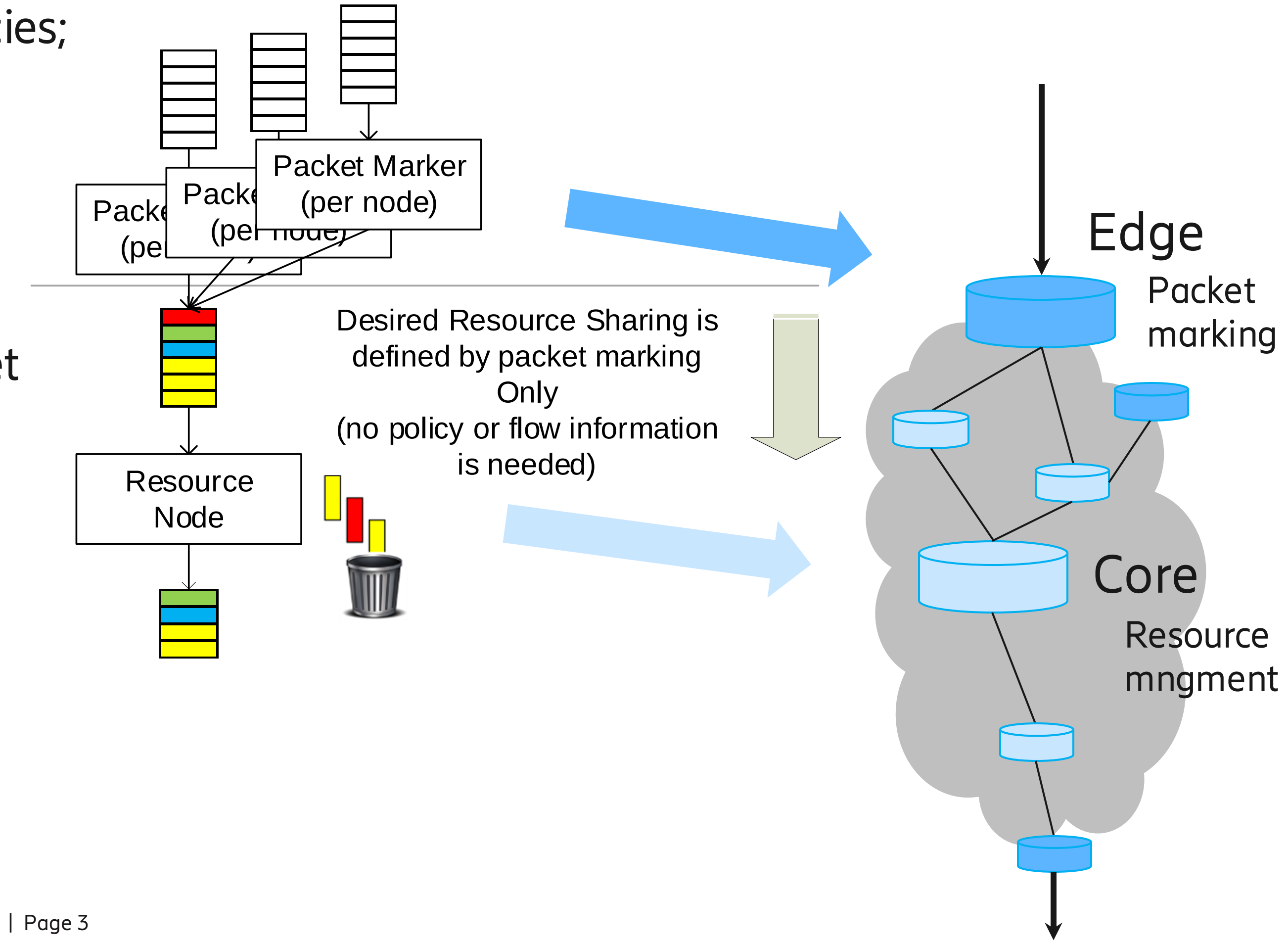
- Define multi-timescale fairness
- Build on existing framework
  - Re-using the existing Per Packet Value-based resource sharing framework
  - Build on Multi-Timescale Bandwidth Profile (MTS-BWP)
- Provide efficient and versatile implementation
  - provide fine-grained fairness on multiple timescales
  - that is independent of traffic mixes and resource bandwidths.
- Demonstrate advantages

- "*getting a scheme to instantly serve web flows for improved performance while maintaining fairness between other persistent traffic remains an open and significant design problem to be investigated*" [1]

[1] Ghulam Abbas, Zahid Halim, and Ziaul Haq Abbas. 2016. Fairness-driven queue management: A survey and taxonomy. *IEEE Communications Surveys & Tutorials* 18, 1 (2016), 324–367.

Overview of Core-Stateless Resource Sharing Example: Per Packet Value based CS RS

- PPV is a Core-Stateless Resource Sharing framework, which
  - allows a wide variety of detailed and flexible policies;
  - enforces those policies for all traffic mixes; and
  - scales well with the number of flows
- Packet Marking at the edge
  - encodes policy into a value marked on each packet
- Resource Node – AQM and Scheduling
  - behavior based on packet marking only
  - no need for
    - policy information
    - flow identification
    - separate queues
  - very fast and simple implementations exist

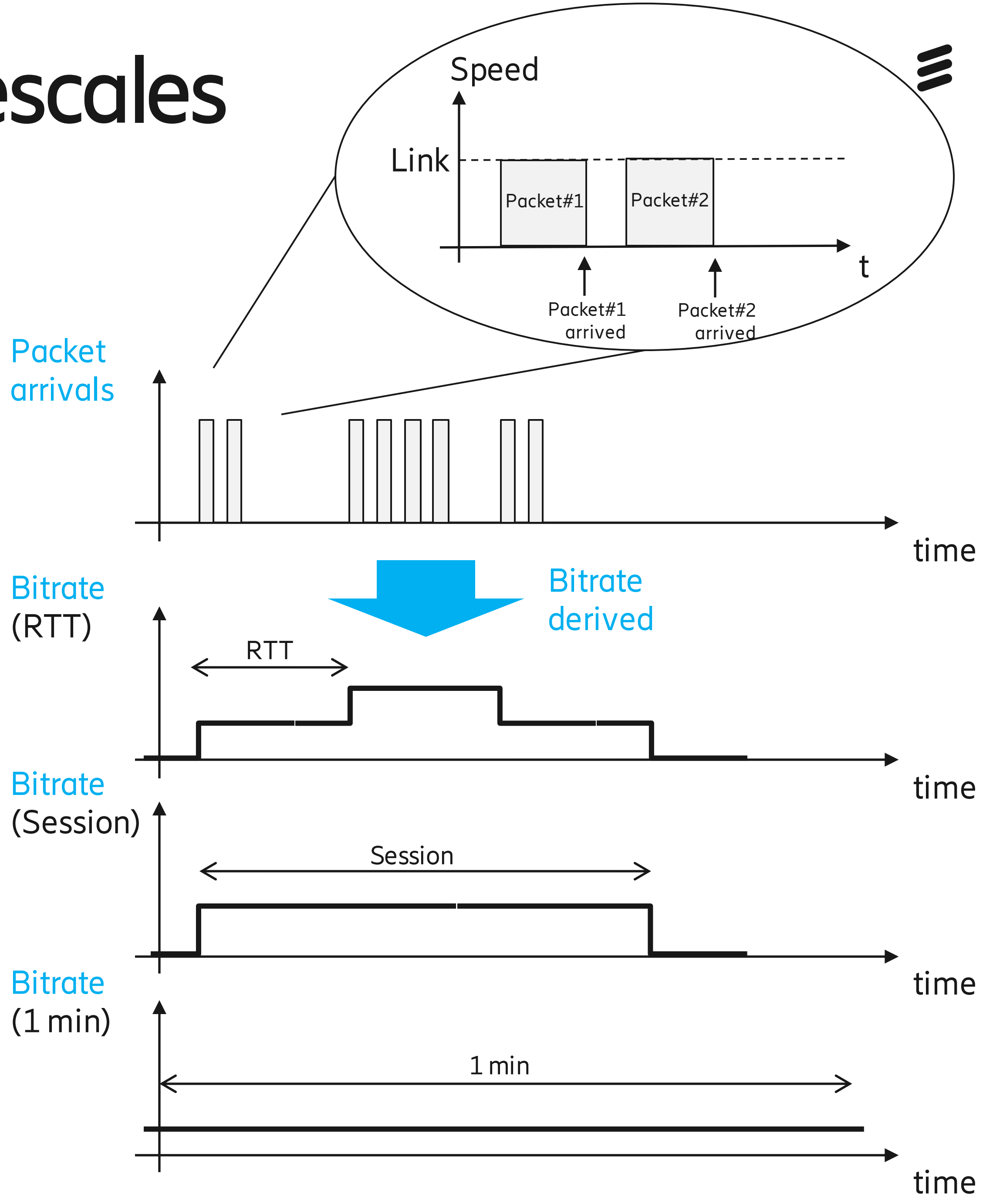


Bitrate measurement and timescales

— Bitrate is a derived measure
  — Discrete packet arrivals are translated to bitrate
  — Bitrate always has a timescale associated

$$Bitrate = \frac{Volume\ (bits)}{Time\ (sec)}$$

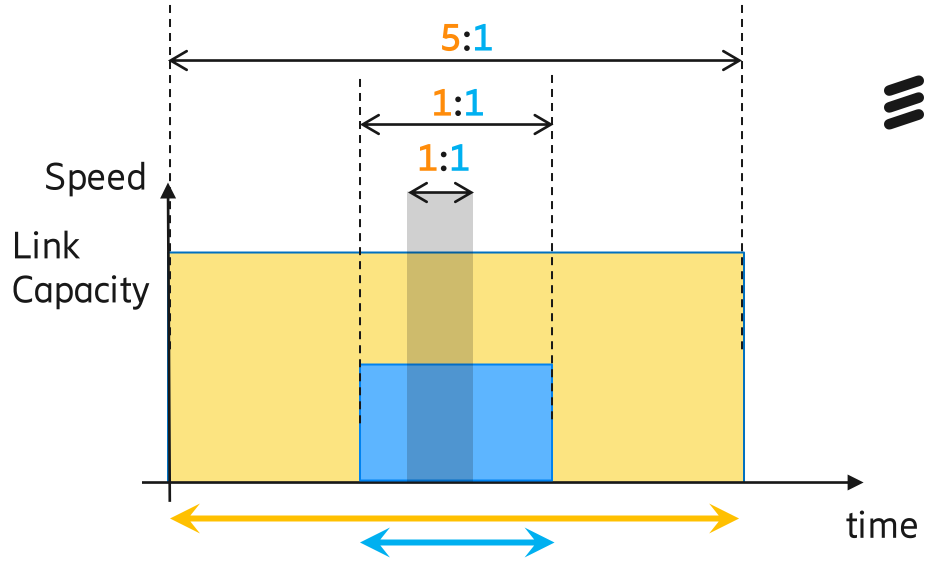
— Natural timescales :
~ RTT
~ 1s – speed shown in apps
~ Session duration (target)
~ 1 minute: short term history and activity
~ 10 minutes: longer term activity
~ Month: monthly cap

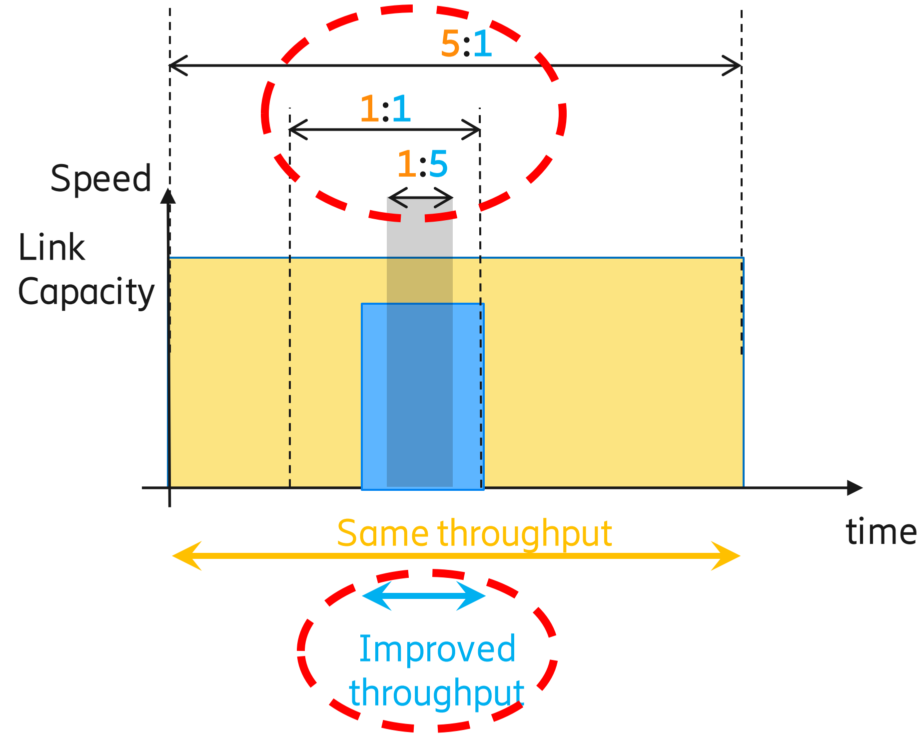


Fairness on multiple timescales

- When to measure bitrate
  - When source is active – to describe performance
  - During both active and inactive periods – to judge the fairness of resource sharing

- Fairness goal on multiple timescales
  - Balanced fairness: multiple timescales are considered
  - Allow higher share on shorter timescales for flows below their fair share in longer timescales
  - We aim at smooth transition as the relations between the rates measured on different timescales changes

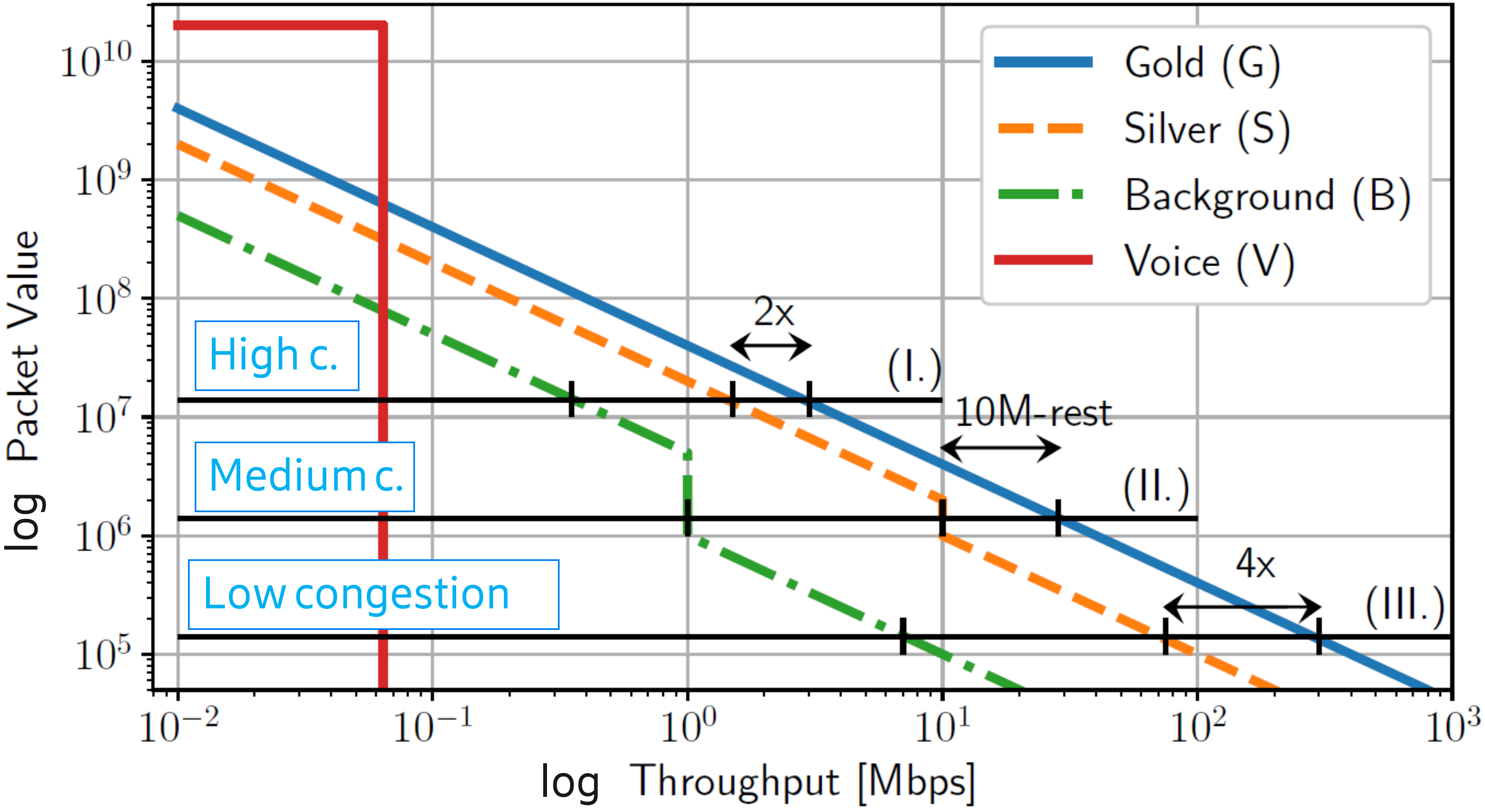




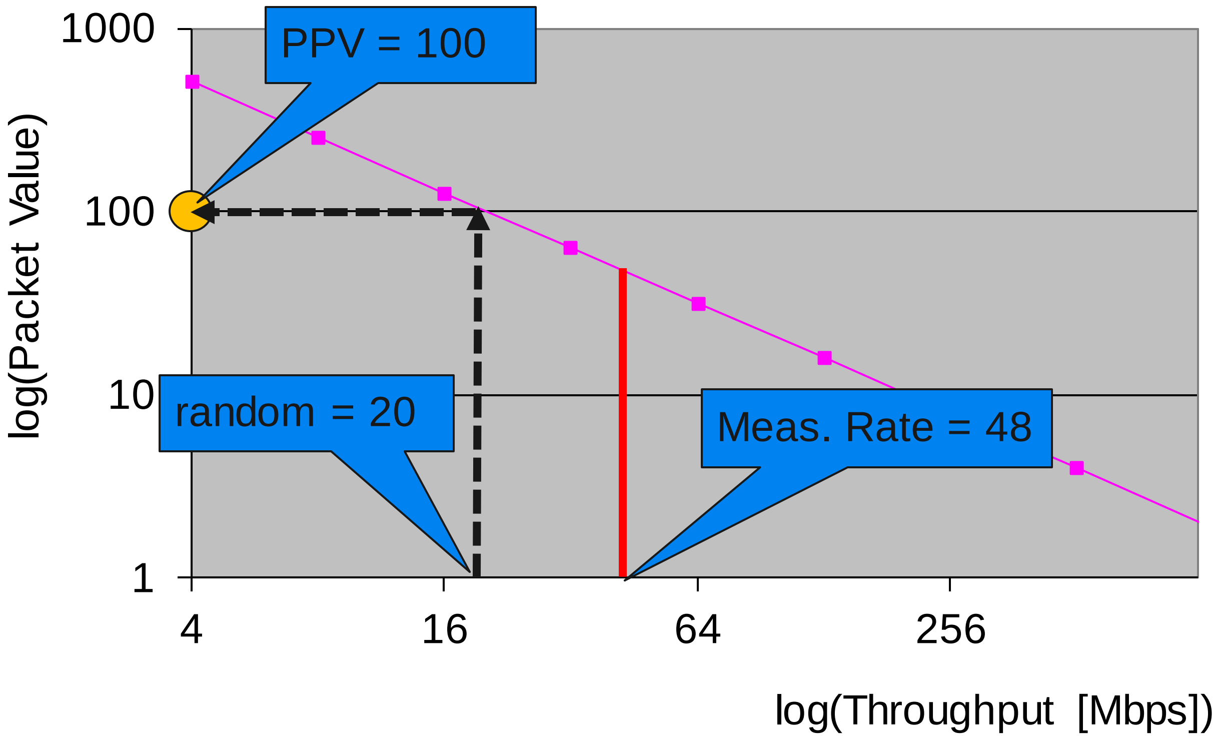
Per Packet Value marking defined by Throughput-Value Functions (TVF)

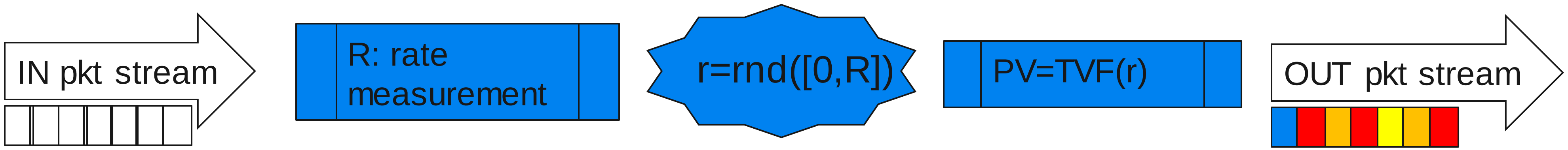
- For a single timescale
- Fine grained control
- Independent of
  - Traffic mix
  - Resource bandwidth

- Each of these result in a Packet Value limit:
  - Congestion Threshold Value (CTV)
- Intersection of the TVFs and the CTV defines desired resource sharing



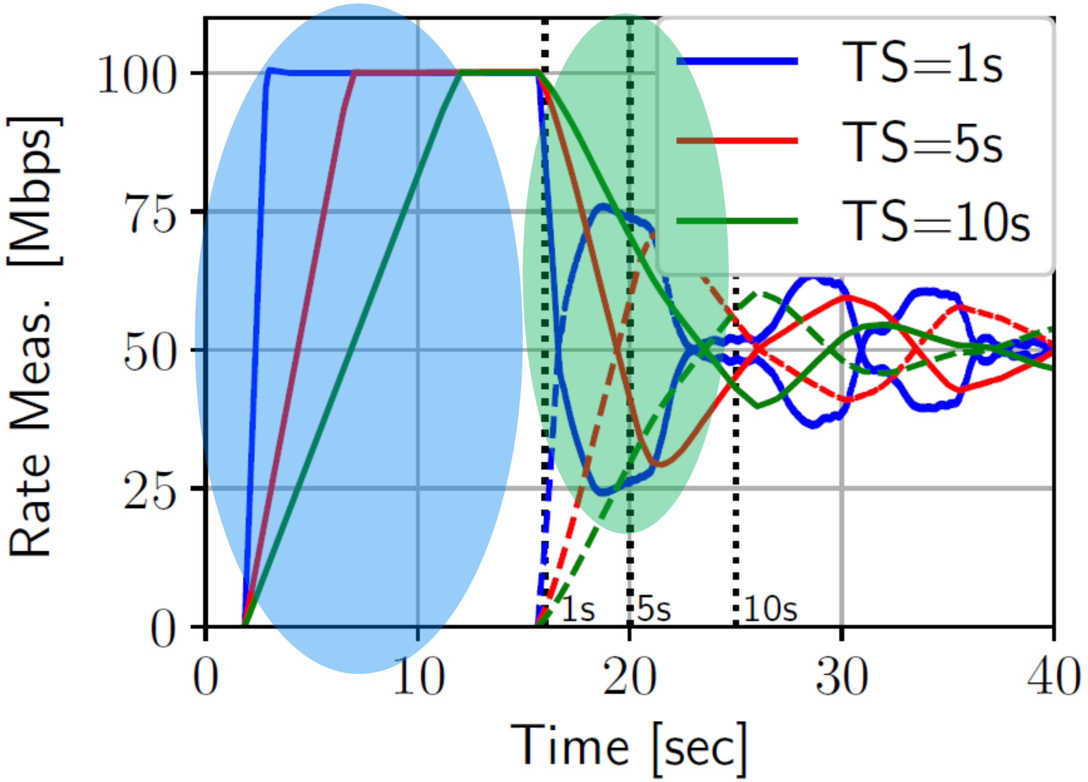
Packet marking based on Rate Measurement





Rate measurement algorithms (RMA) and examples

- Token Bucket Based RMA
  - For the ~RTT times-cale only
  - Models the fair throughput and buffer share at the bottleneck
  - A single Token Bucket
    - its rate changes when empty/full
- Sliding window-based RMA
  - All longer timescales (TS)
  - Rate = "amount of bits arrived in [t-TS,TS]"/TS
  - Efficient approximation of this (see article)
    - Time-Dependent Rate Measurement algorithm with Time Window Moving Average (TDRM-TWMA)

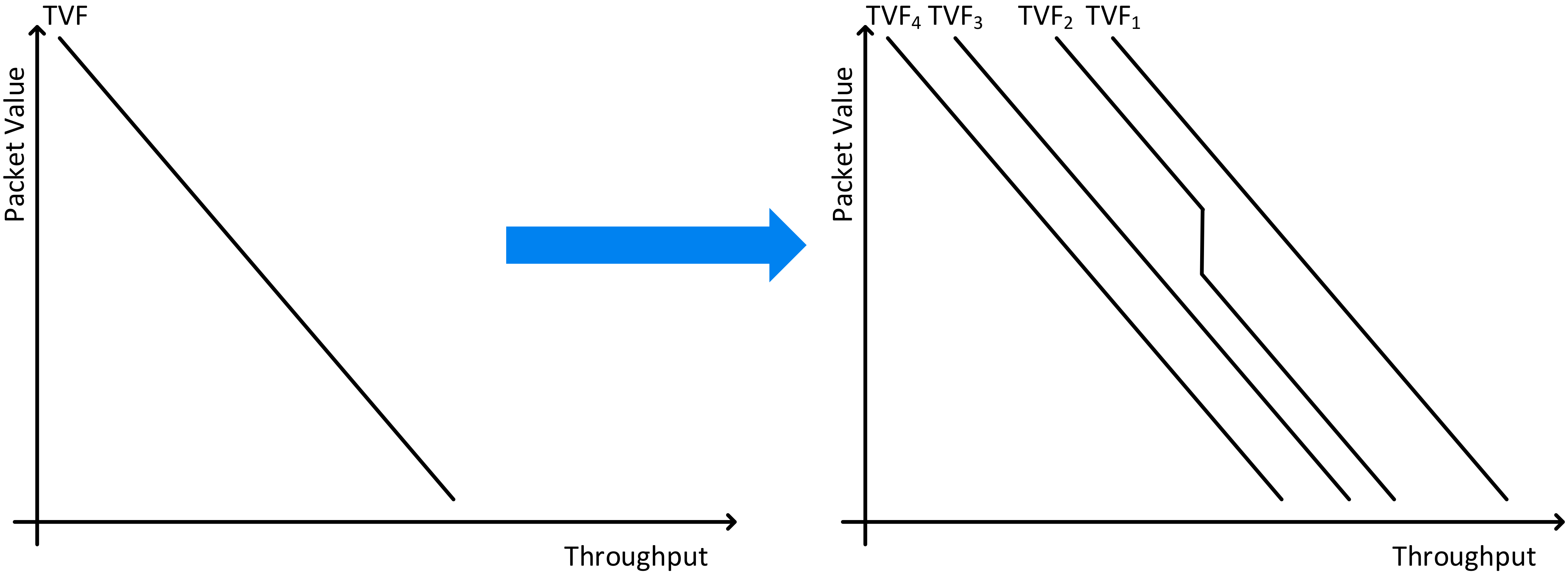


- TS=[10ms, 1s, 5s, 10s]
- When transmission starts
  - $R_1 > R_2 > R_3 > R_4$
- Rate decrease/transmission stop
  - $R_1 < R_2 < R_3 < R_4$

Multi-Timescale Throughput-Value Function (MTS-TVF)

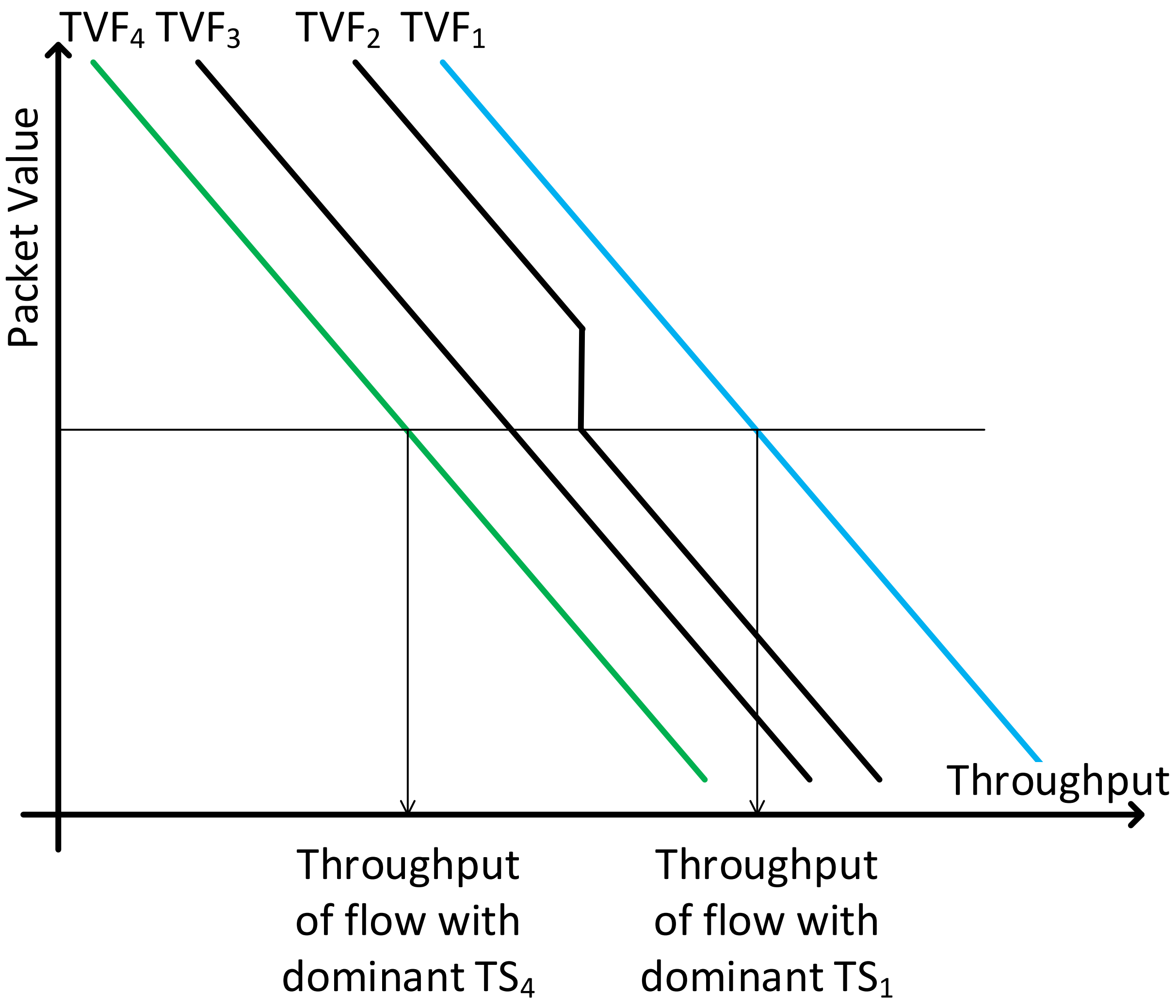
- (Single-TS) TVF
- 1 TVF per flow type

- MTS-TVF
- 1 TVF per TS per flow type



Multi-Timescale Throughput-Value Function (MTS-TVF) Resource Sharing

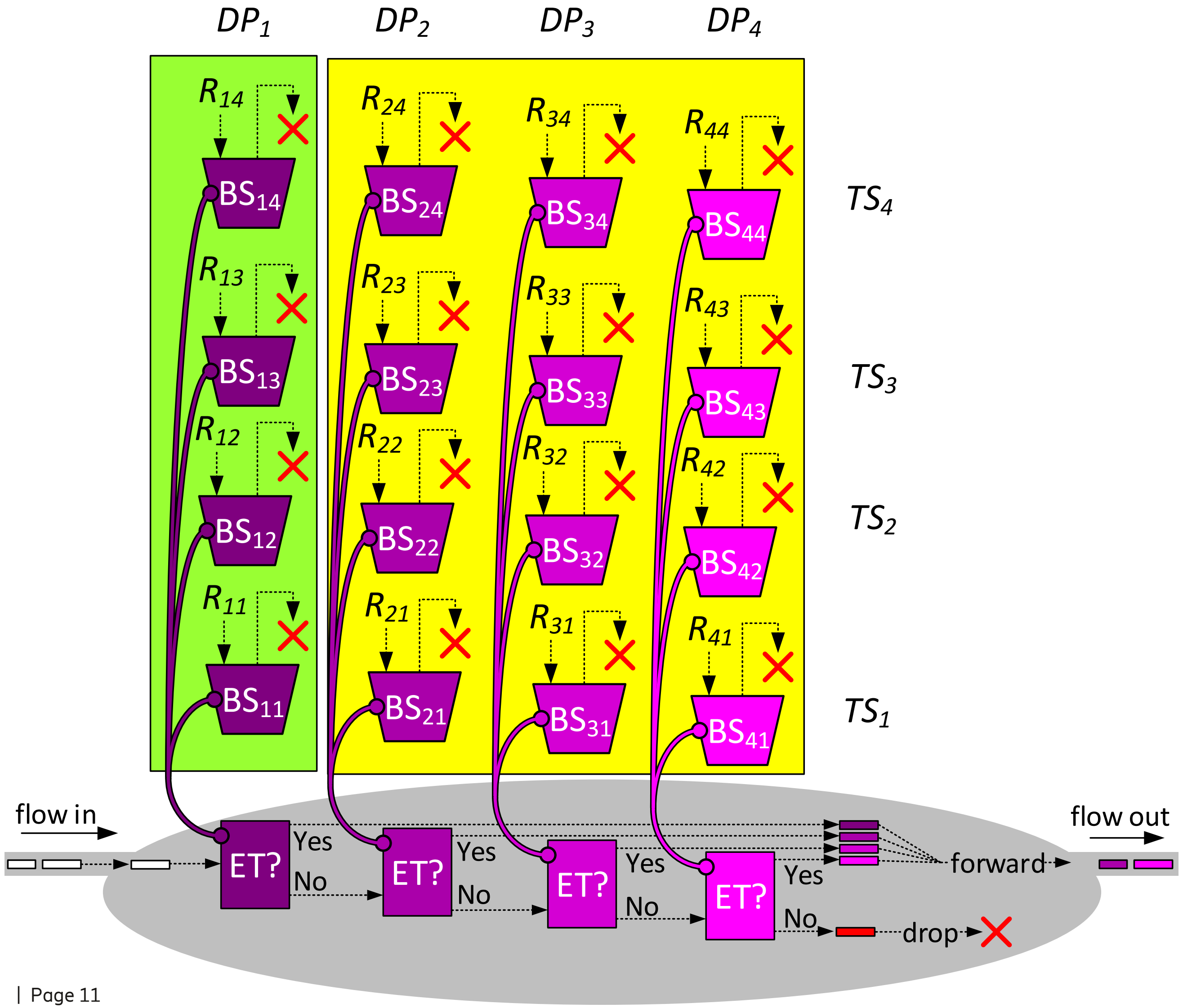
- Dominant timescale (TS_i)
  - When the rate measurement on that timescale (R_i) is the largest
  - (or the longest timescale among largest and roughly equal rate measurements)
- Example: Two flows of the same flow type
  - One has dominant $TS=TS_1$ (just arrived)
  - The other has $TS=TS_4$ (long history)
  - They shall share the bottleneck according to TVF_4 vs. TVF_1
  - as if they would be of different flow types in the single-TS framework
  - But: we aim at smooth transitions when relation between R_is change



Multi-Timescale Bandwidth Profile (MTS-BWP)

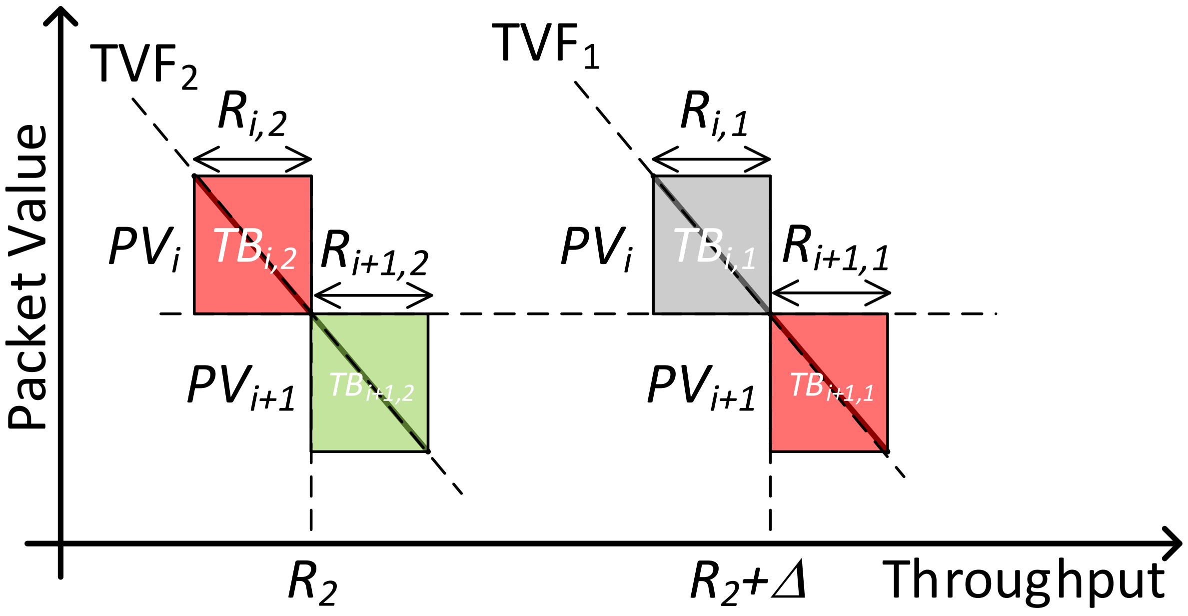
- Provides multi-timescale fairness among flows
- Only in well defined scenarios
  - Number of flows
  - System capacity
- A 4 timescale, 4 Drop precedence example
  - (ET = Enough tokens)

- Any MTS-TVF can be quantized to an MTS-BWP
  - Not practical implementation
  - E.g. 65k different PVs → 65k*4 token buckets



Practical packet marking using MTS-TVF

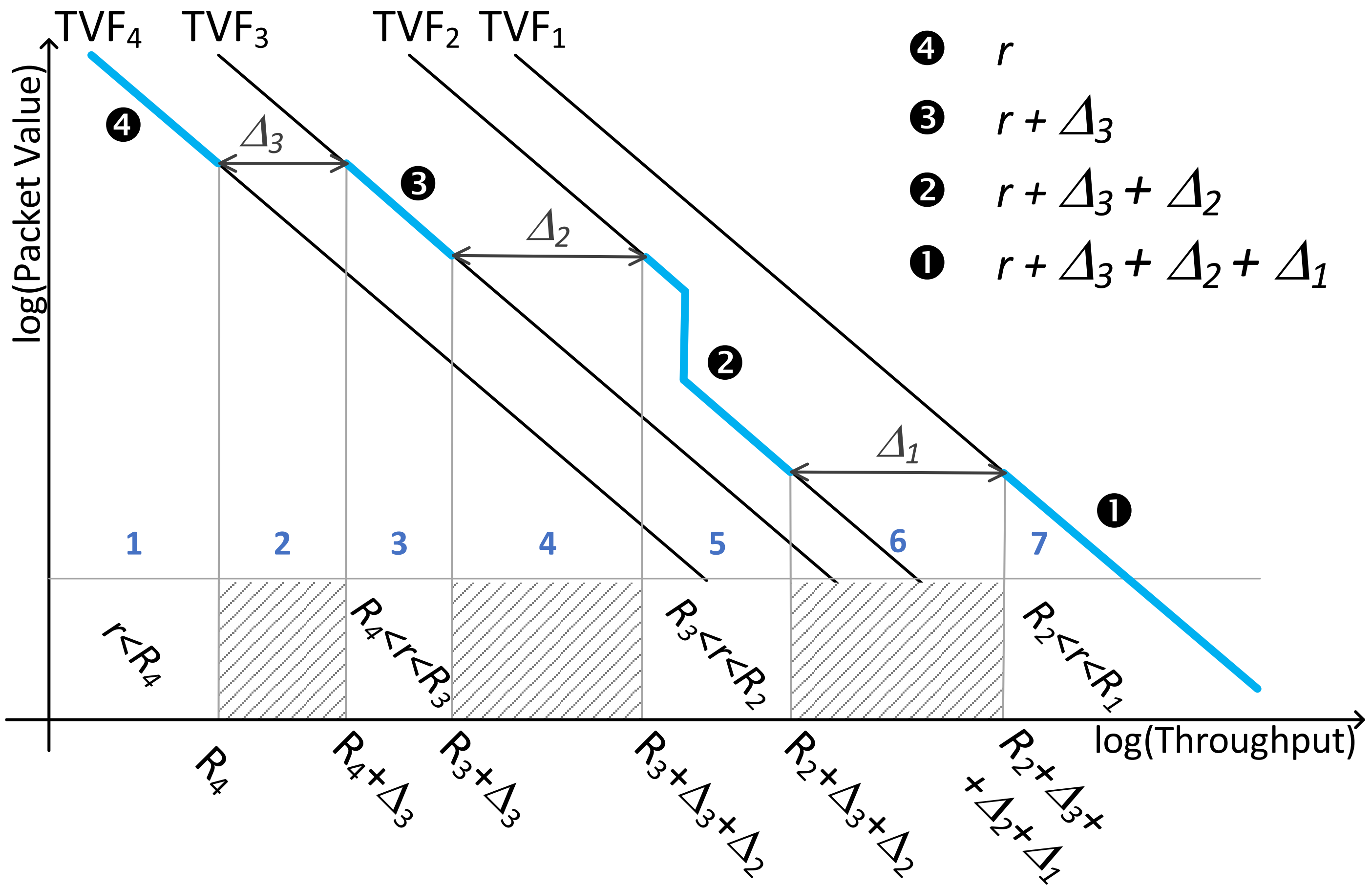
- Using a quantized MTS-TVF to MTS-BWP
- Multi-Timescale Bandwidth Profile (MTS-BWP)
  - Also measures rate on each timescale (indirectly)
  - The limiting Token Buckets determine the rate measurement
  - At these rate measurements it switches between timescales,
    - i.e. between TVFs



Efficient packet marking based on Multi-Timescale Throughput-Value Function

- Measure rate for all the timescales
  - R_4, R_3, R_2, R_1
- At R_is determine distance between the TVFs
- Blue region of the TVFs are used
  - Changes as R_is change
- Algorithm
  - r is is a uniform rnd $[0, R_1]$
  - Determine right region i= ❶ - ❹
    - Relation between R_is and r
  - Determine Δ_is

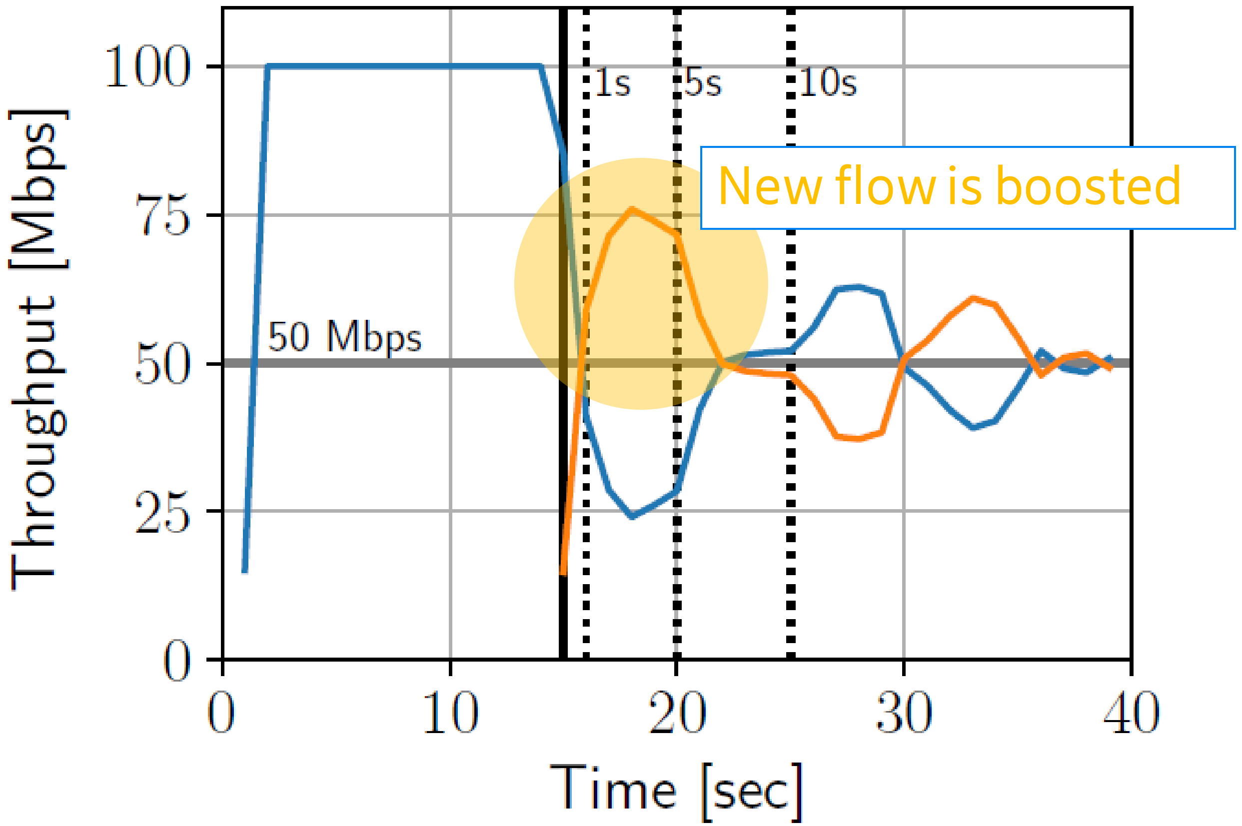
$$PV = TVF_i\left(r + \sum_{j=i}^{k-1} \Delta_j\right)$$



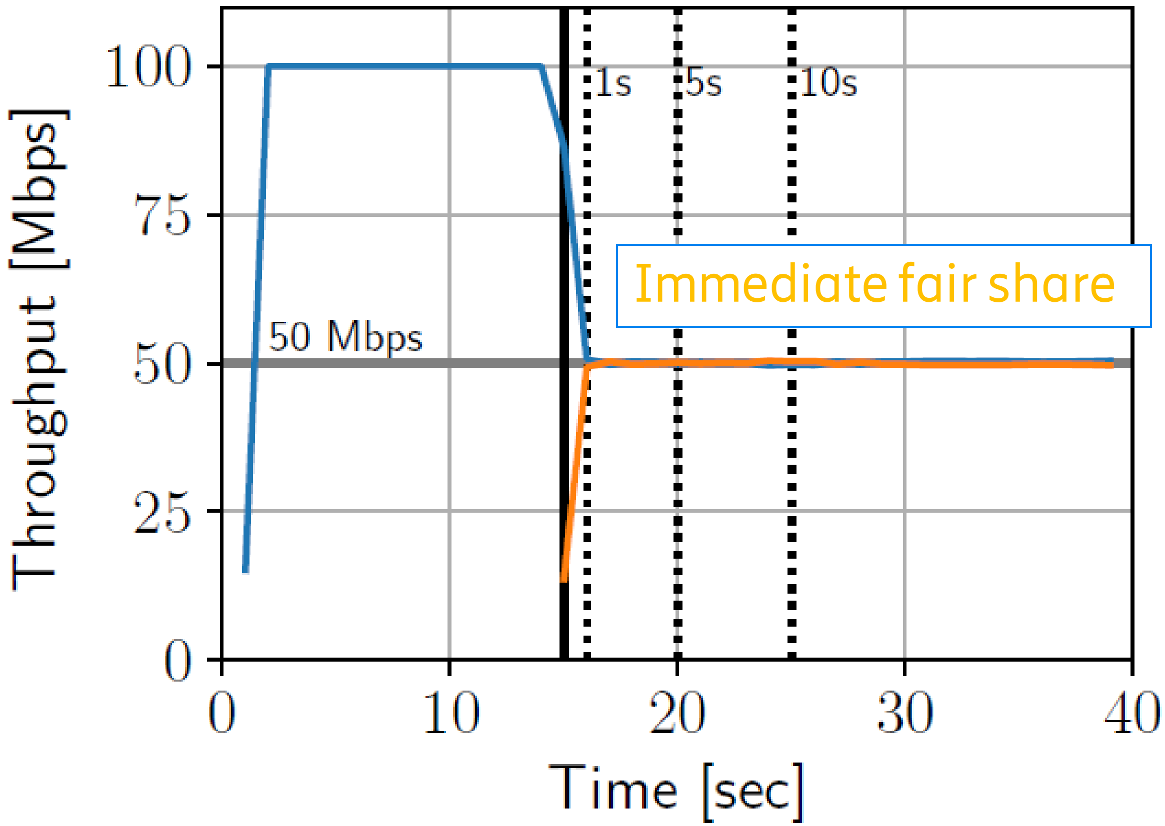
Simulations

- NS-3, NS-3 DCE (TCP Congestion Control)
- Core scheduler unchanged from our article "Towards a Congestion Control-Independent Core-Stateless AQM"
  - 10 ms delay target
- A flow consist of either
  - 1 DCTCP connection or
  - 4 Cubic TCP connections (faster slow start)
- TS=[10ms, 1s, 5s, 10s]
- TVF_4 is Gold or Silver as single-TS TVF
  - Shorter TSs weights 2, 4, 8, i.e. $TVF_3(x) = TVF_4(x/2)$, $TVF_2(x) = TVF_4(x/4)$, $TVF_1(x) = TVF_4(x/8)$.

Greedy flows of the same traffic class (DCTCP)

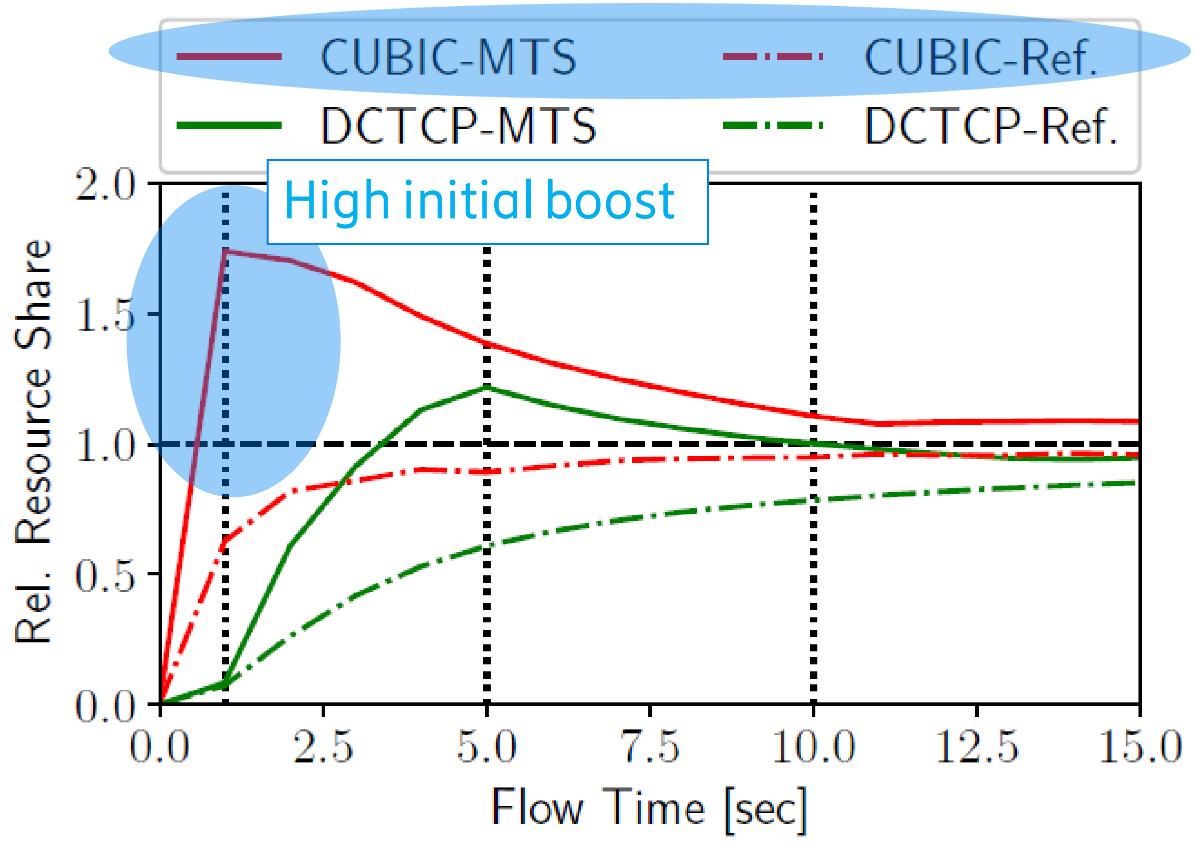


Multi-Timescale PPV

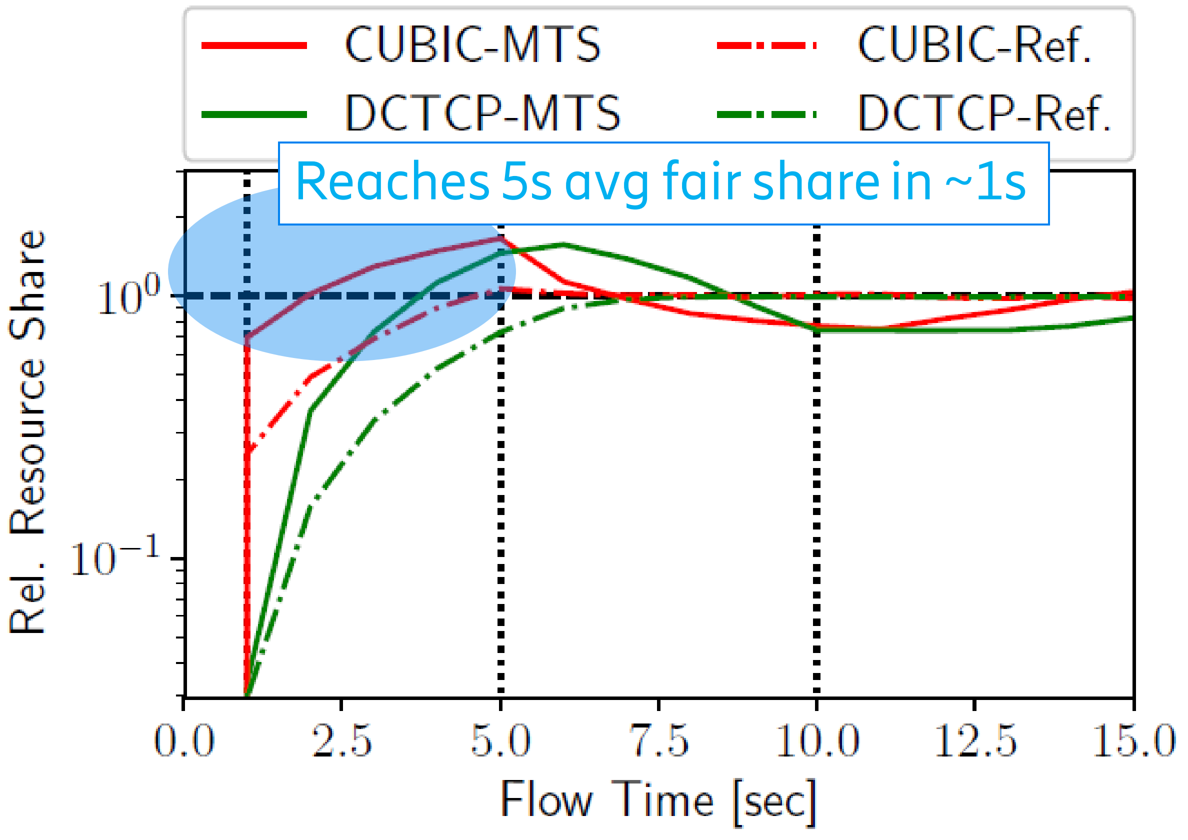


Reference: Single-Timescale PPV

Greedy flows of the same traffic class



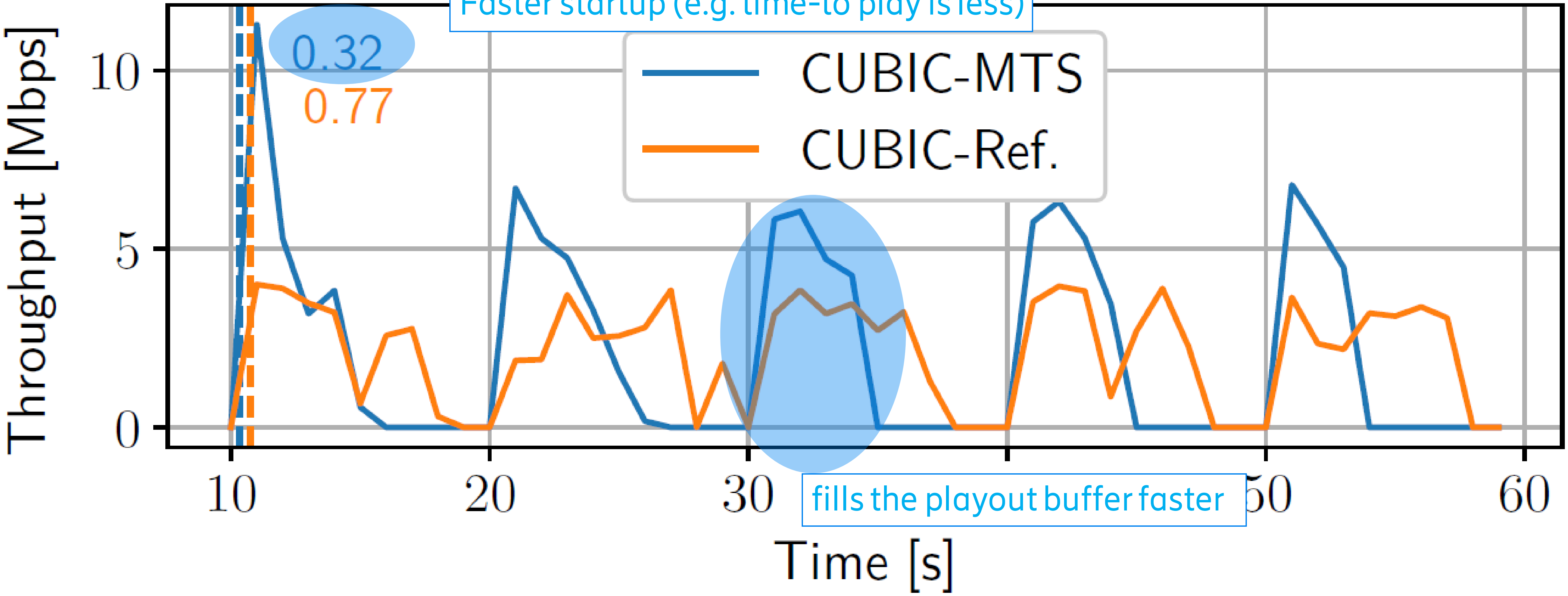
(a) Flow-Time Average



(b) 5s Time-Window Average

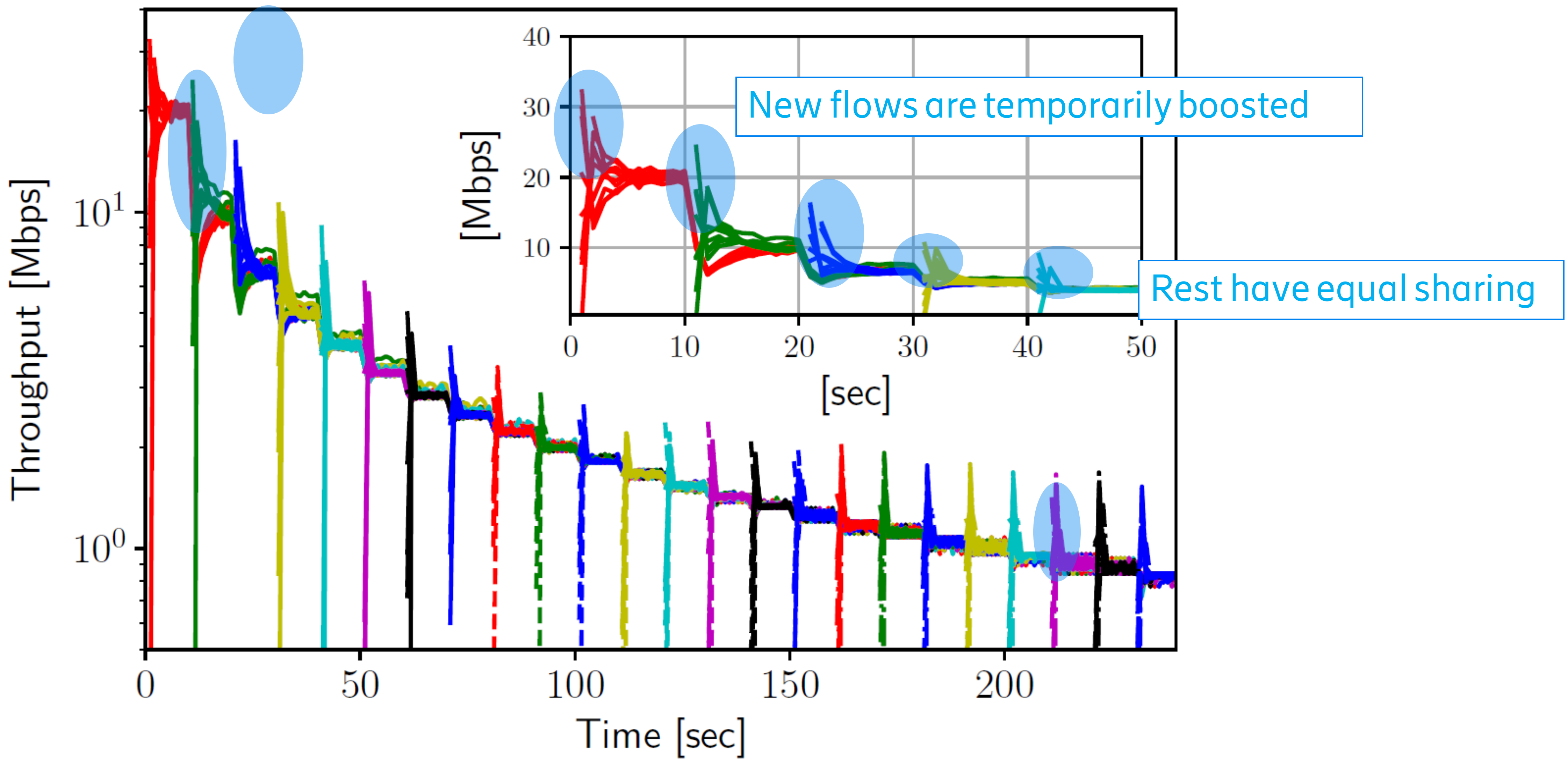
Simple adaptive streaming model
MTS fairness for on-off pattern

Faster startup (e.g. time-to play is less)



fills the playout buffer faster

Continuous arrival: 10 new flows every 10s



Discussion

- Initial results look promising
  - Multi-timescale fairness works
  - Significant performance gains
    - Advantage for new flows/starting phase
    - Better long term fairness for flows with on-off behavior

- Future work
  - What is the practical number of timescales to be used?
  - How shall the timescales be dimensioned?
  - How to design multi-timescale TVFs?
  - Does it make sense to use a different kind of policy at various timescales?
  - What further policies that have practical relevance can be described in this model?

http://ppv.elte.hu