FLOWER
an Innovative Fuzzy LOWer-than-Best-EffoRt Transport Protocol

Si Quoc Viet Trang

03/12/2015

Funded by

Centre National d'Études Spatiales (CNES)

and

Thales Alenia Space (TAS)

Directed by

Emmanuel Lochin (ISAE)

Cédric Baudoin (TAS)

Emmanuel Dubois (CNES)

Patrick Gélard (CNES)

#1

Motivation

“Can we manage to exploit the unused satellite link capacity without disturbing commercial traffic?”

– CNES & TAS

The objectives are:

- Fully use the satellite link
- Add a signaling or non-commercial traffic
- Provide a new service for new business
- **And…**

To provide a free Internet Access for All

83% of 10000 respondents consider Internet access as an **Human Right**

of the world population **don't have** an Internet connection **60%**

Yes, we can...

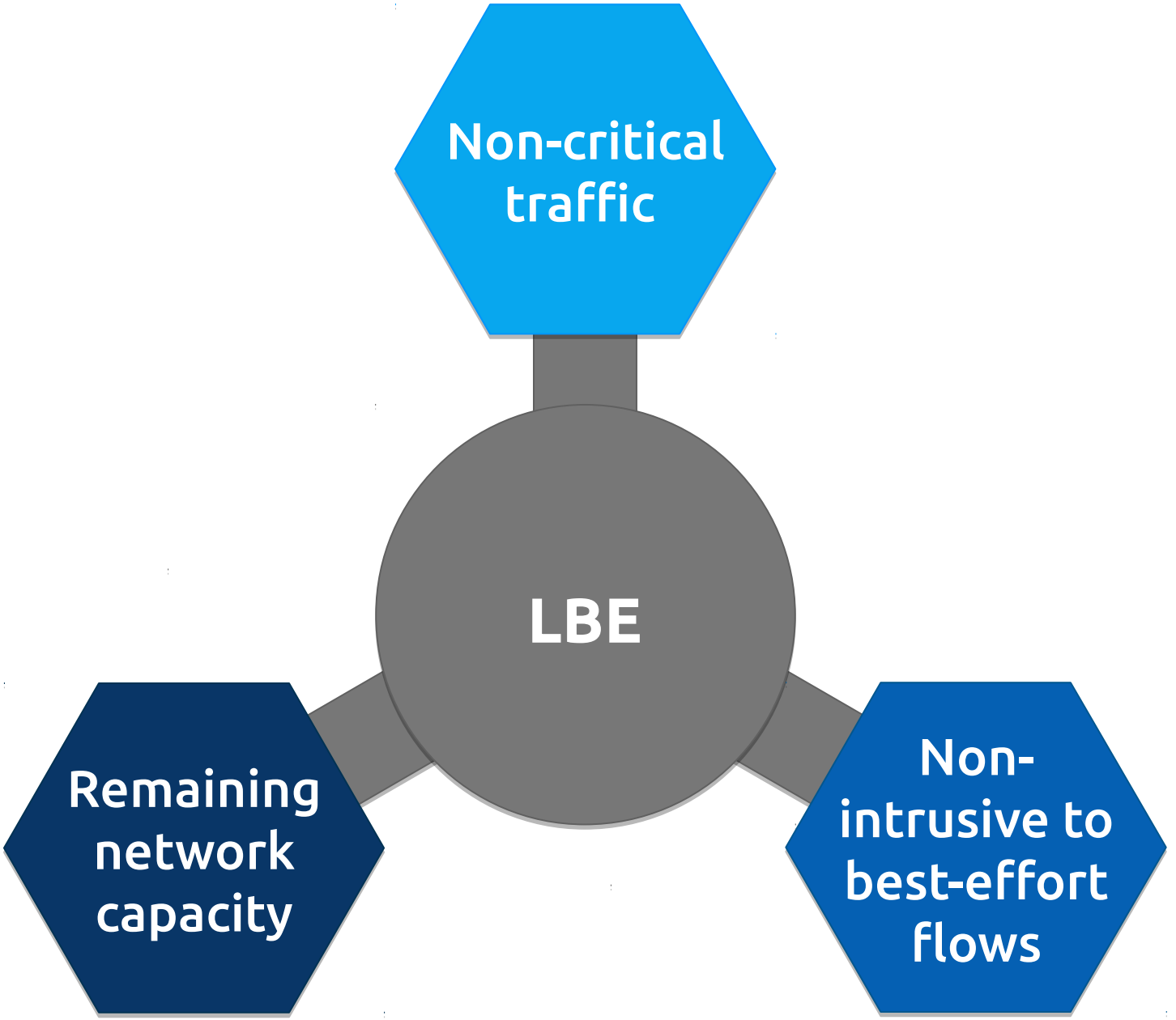
with Low than BE...

with LEDBAT!

(Low Extra Delay Background Transport)

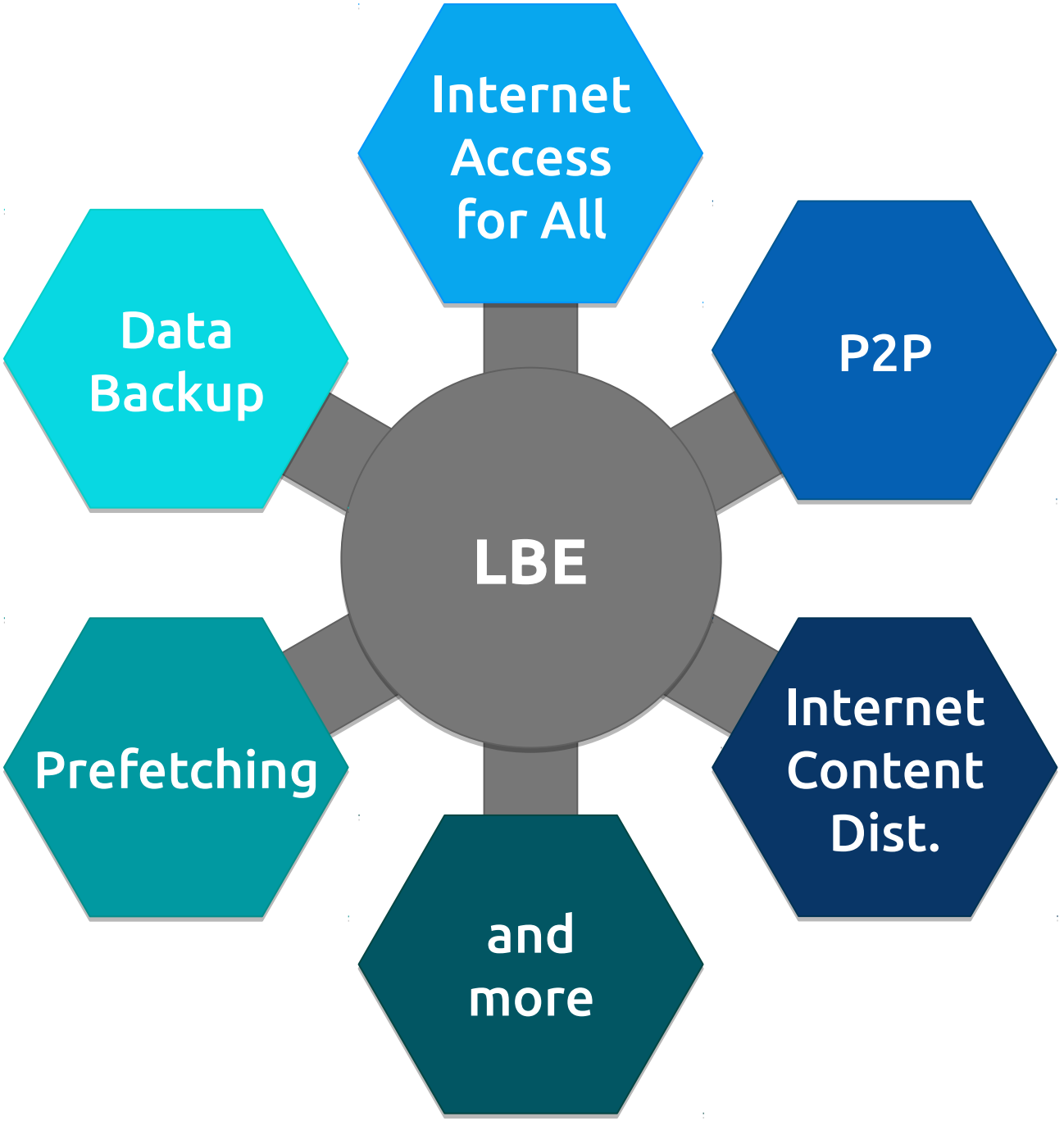
#2

Lower than Best-Effort (LBE) Service


Non-critical traffic
LBE
Remaining network capacity
Non-intrusive to best-effort flows

It's worse than best-effort.
Why do we bother?

LBE + Satellites
= Internet Access for All !


Internet Access for All
Data Backup
P2P
LBE
Prefetching
Internet Content Dist.
and more

We just have introduced the LBE service

Let's have a look now at the **most popular LBE protocol** that implements such service

#3
LEDBAT

LEDBAT is a congestion control (CC) protocol that enables an LBE service

LEDBAT is **delay-based** CC while TCP is loss-based CC

Implements a **P-type controller** to control the sending rate as a function of the **end-to-end queuing delay**

Objective is to respond to congestion **before** standard TCP

Reduces the rate when it detects that the queuing delay is **beyond a threshold** (**target queuing delay**)

Parameters

target τ – maximum queuing delay that LEDBAT may introduce in the network

gain σ - factor for amplifying the response to variations of queuing delay estimated (Qd)

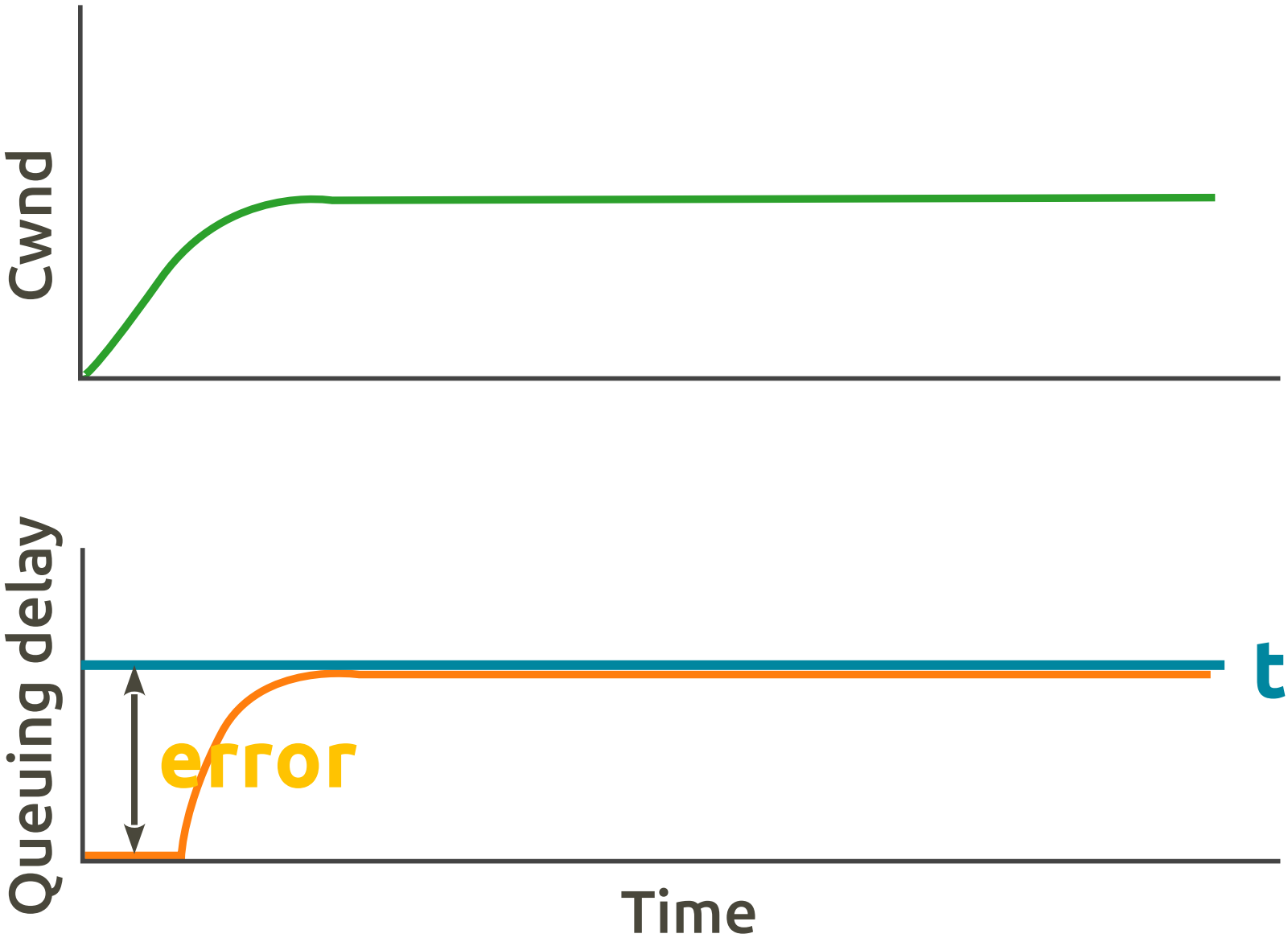
Algorithm

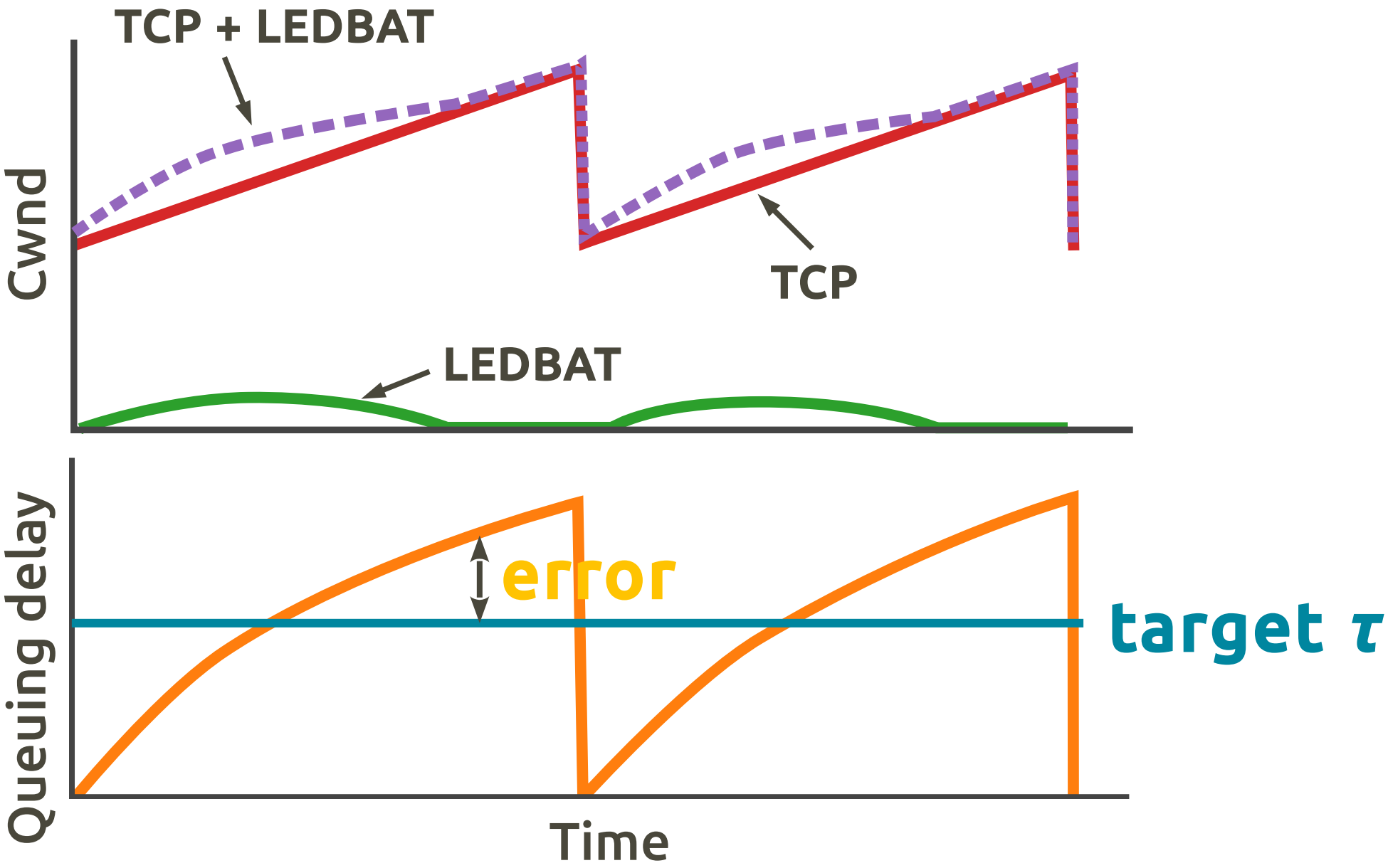
$$error = \tau - Qd$$

$$cwnd = cwnd + (\sigma * error)/cwnd$$

Proportional term

cwnd – congestion window: number of packets sent by time unit as a function of the congestion level

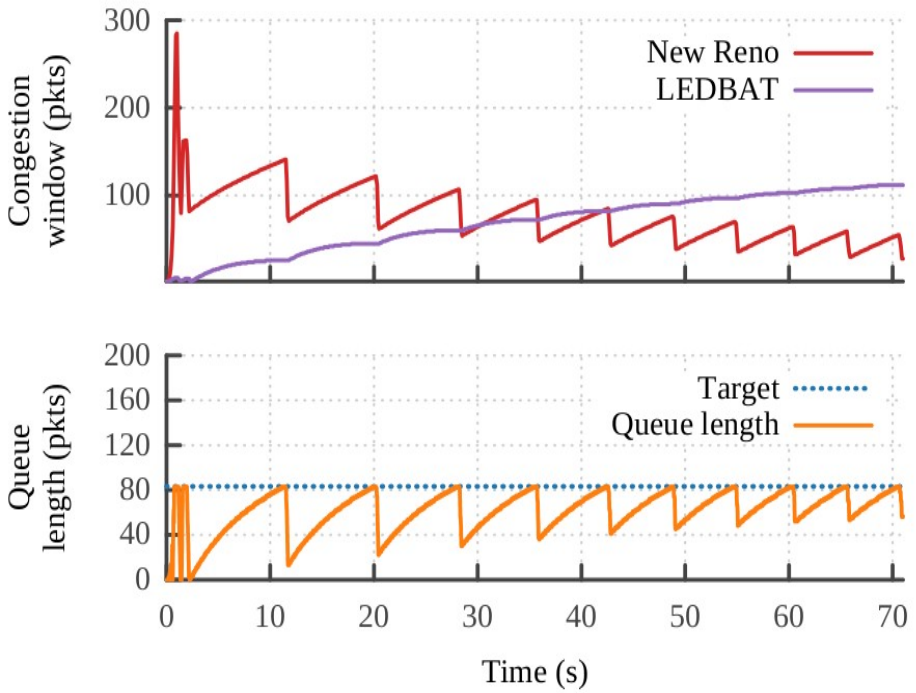

Cwnd
Queuing delay
target τ
error
Time


TCP + LEDBAT
Cwnd
TCP
LEDBAT
Queuing delay
error
target τ
Time

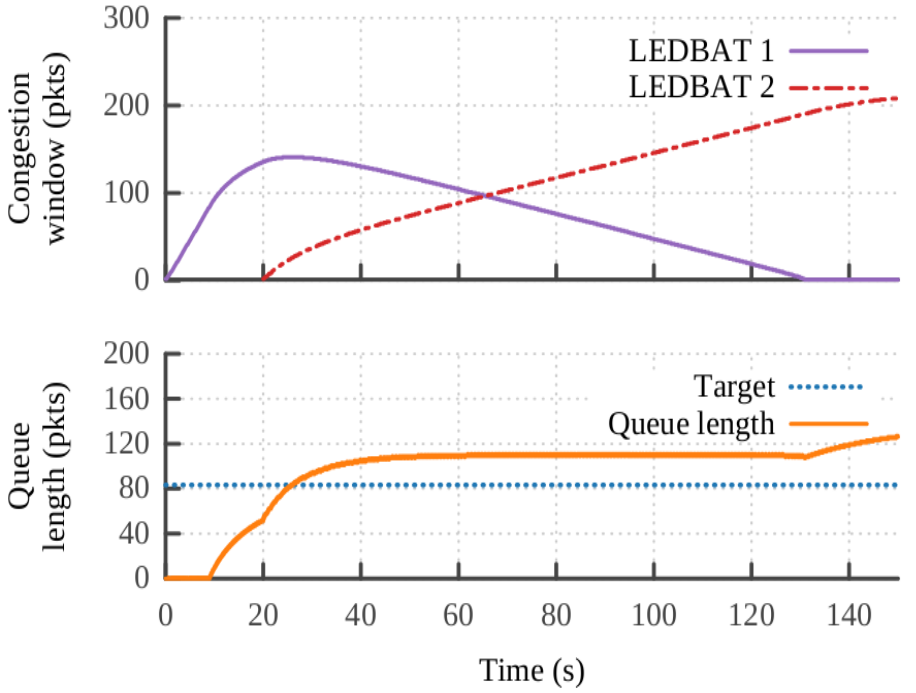
So ... let's use LEDBAT over satellite link!


Houston
we have a
Problem

Two Main LEDBAT Problems



Aggressiveness



Latecomer unfairness

C = 10 Mb/s, Owd = 50 ms, B = BDP

#4

How to choose LEDBAT parameters?

Experimental conditions

Objective seeking an **“optimal” combination** of parameters that **fits in most/all network configurations**

We tests several cases by varying the bottleneck characteristics in terms of

- Capacity (from 1 to 50Mb/s)
- One way delay (from 10 to 250ms)
- Buffer size (as a function of the BDP products)
- With various target and decrease gain

Cluster Analysis

B = BDP, if ηTCP ≥ 0.8 then ηref = ηLEDBAT

For other B, Δ = |ηLEDBAT – ηref|

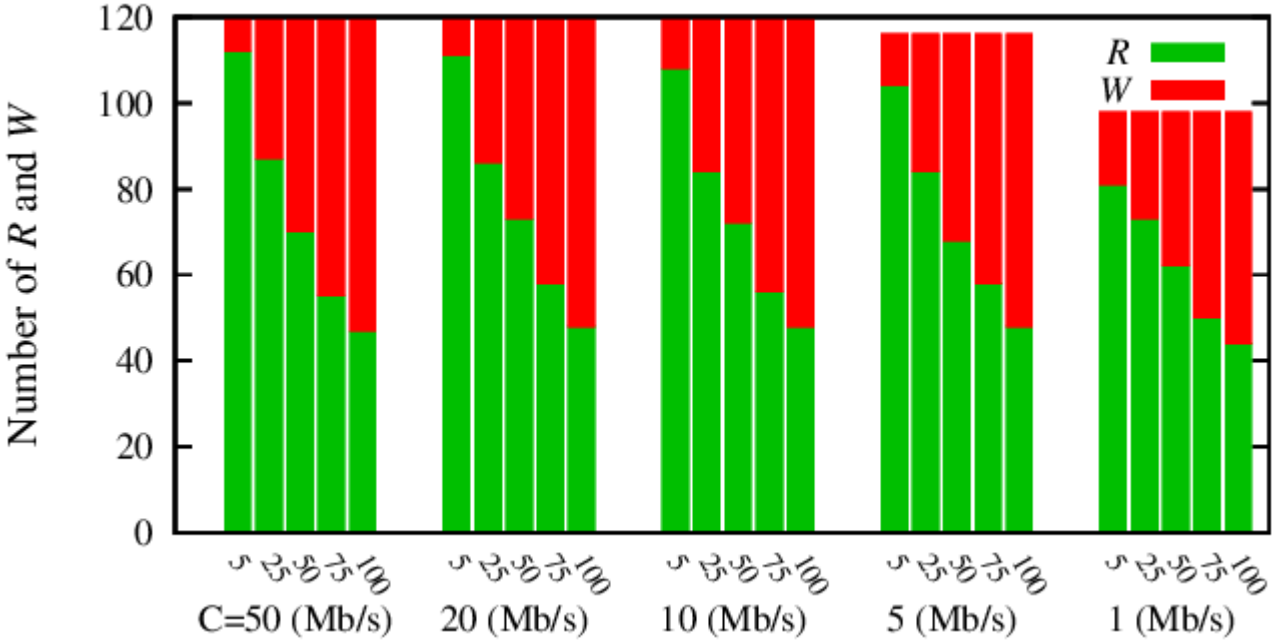
R

Δ ≤ ε and ηTCP ≥ 0.8

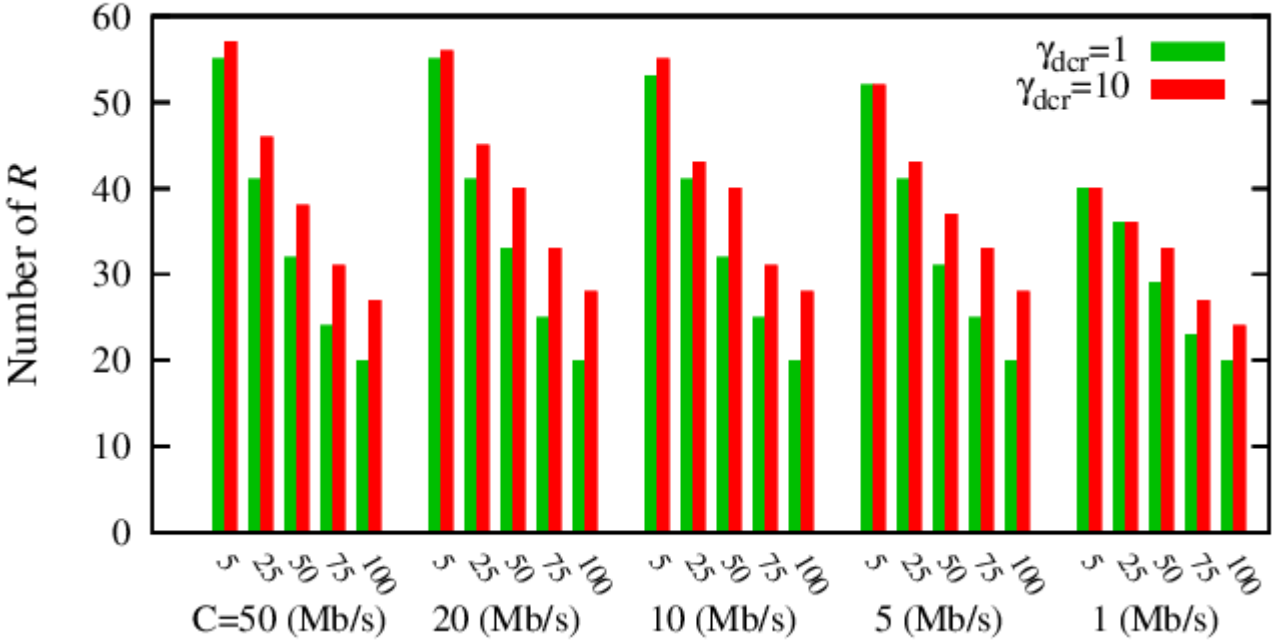
W

Δ > ε or ηTCP < 0.8

ε = 0.15



Optimal Target = 5 ms



Optimal Gain = 10

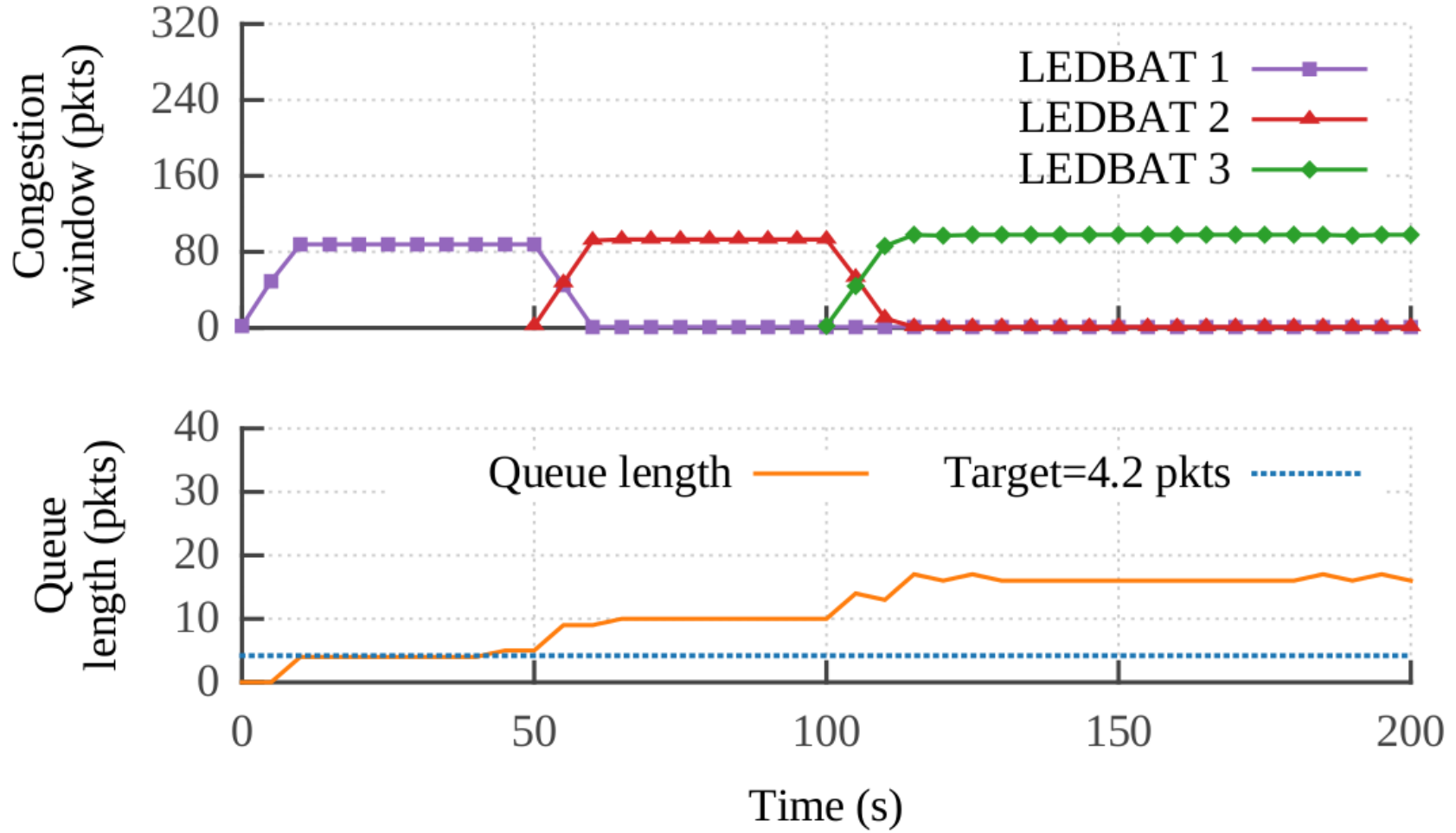
Conclusion

(5ms, 10) is **statistically** optimal in all test cases

Setting LEDBAT parameters is a **trade-off**


Houston, we have another Problem
NASA

Latecomer unfairness still persists!



LEDBAT is **not adaptive** and only implements a P-type controller to control the sending rate as a function of the delay. Both parameters (target and gain) are **fixed**

One possible solution would be to implement an adaptive functionality inside this P-type controller but such adaptation would require a fine grained analytical model

“All these characteristics make this problem a perfect candidate for **FUZZY LOGIC!**”

– The fuzzy expert

Why Fuzzy Logic?

No mathematical model required

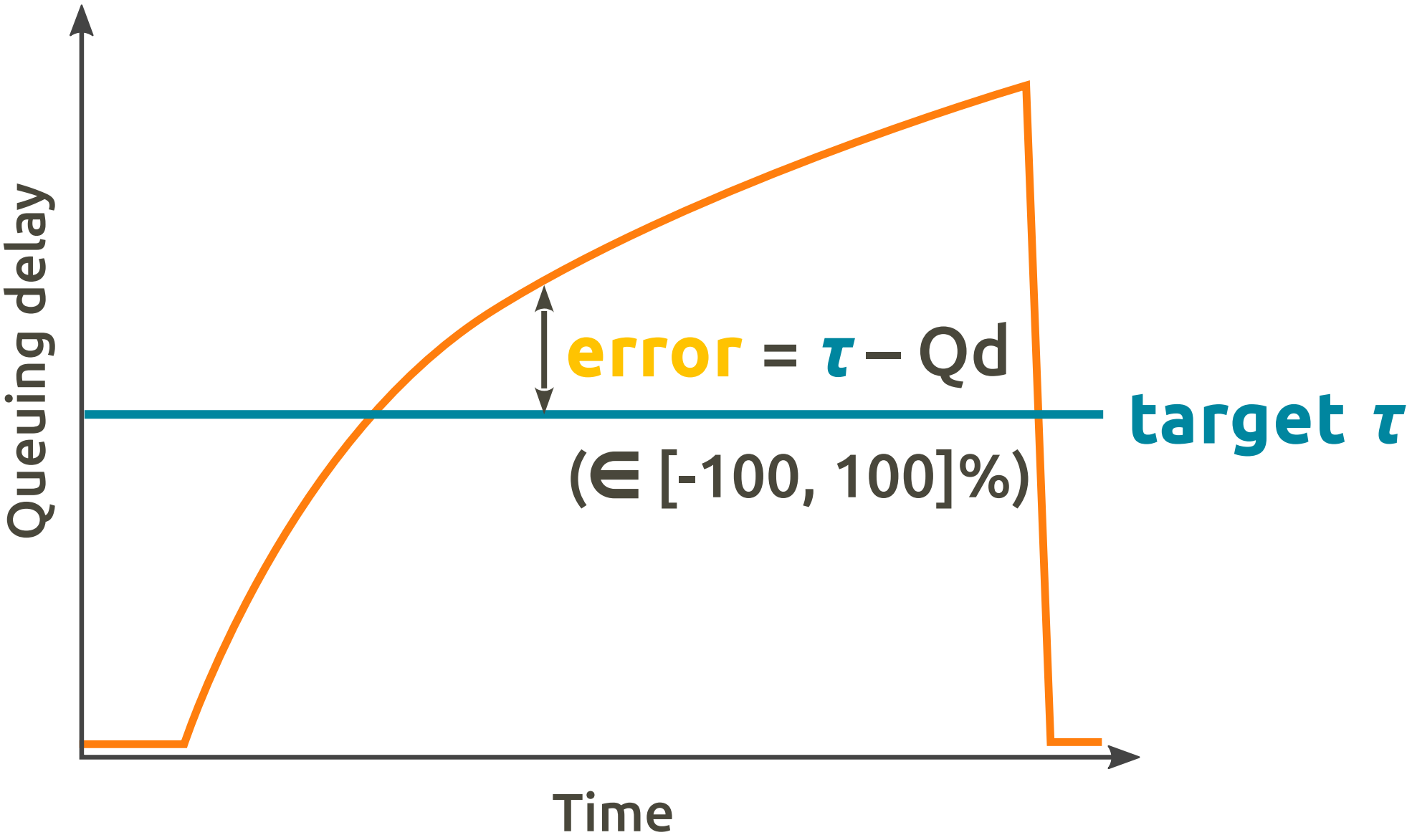
Allows to incorporate our **heuristic knowledge** about how to control the system

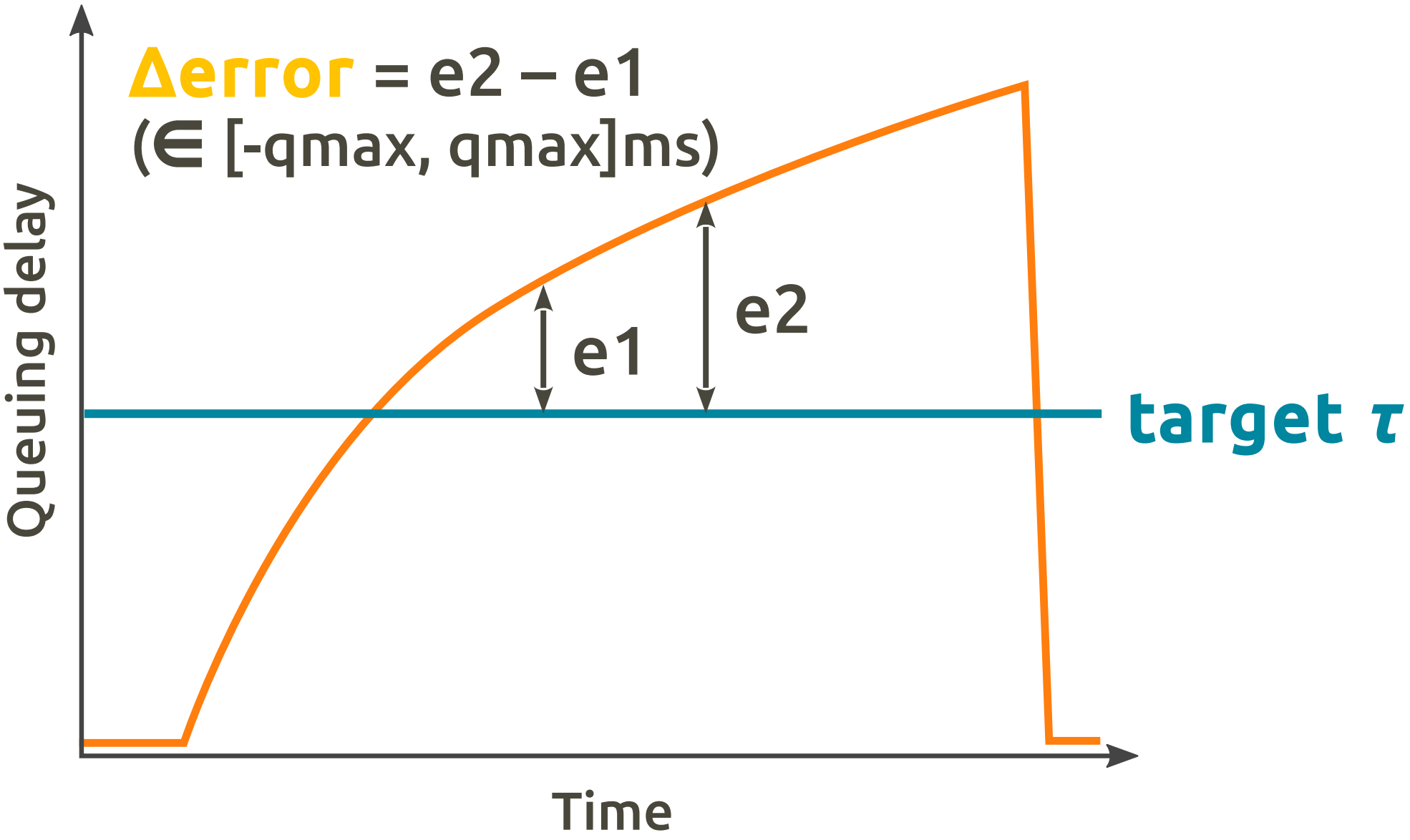
Wider range of operating conditions than PID controllers

#5

FLOWER

How to control the queuing delay?


Queuing delay
error = τ – Qd
target τ
(∈ [-100, 100]%)
Time


Δerror = e2 – e1
(∈ [-qmax, qmax]ms)
Queuing delay
e1
e2
target τ
Time

How to express this ?

$$\Delta cwnd = f(e, \Delta e) \; ?$$

$(\in [-1, 1])$

Basic Principle of Fuzzy Logic

We use the **linguistic rules** to express the expert's knowledge about how to control the process

The general form of the linguistic rules is:

If premise **Then** consequent

Basic Principle of Fuzzy Logic (ctd)

e, Δe, Δcwnd are **linguistic variables** which take on **linguistic values**:

NVL, NL, NM, NS, NVS, Z, PVS, PS, PM, PL, PVL

(P: Positive; N: Negative; V :Very; L: Large; M: Medium; S: Small; Z: Zero)

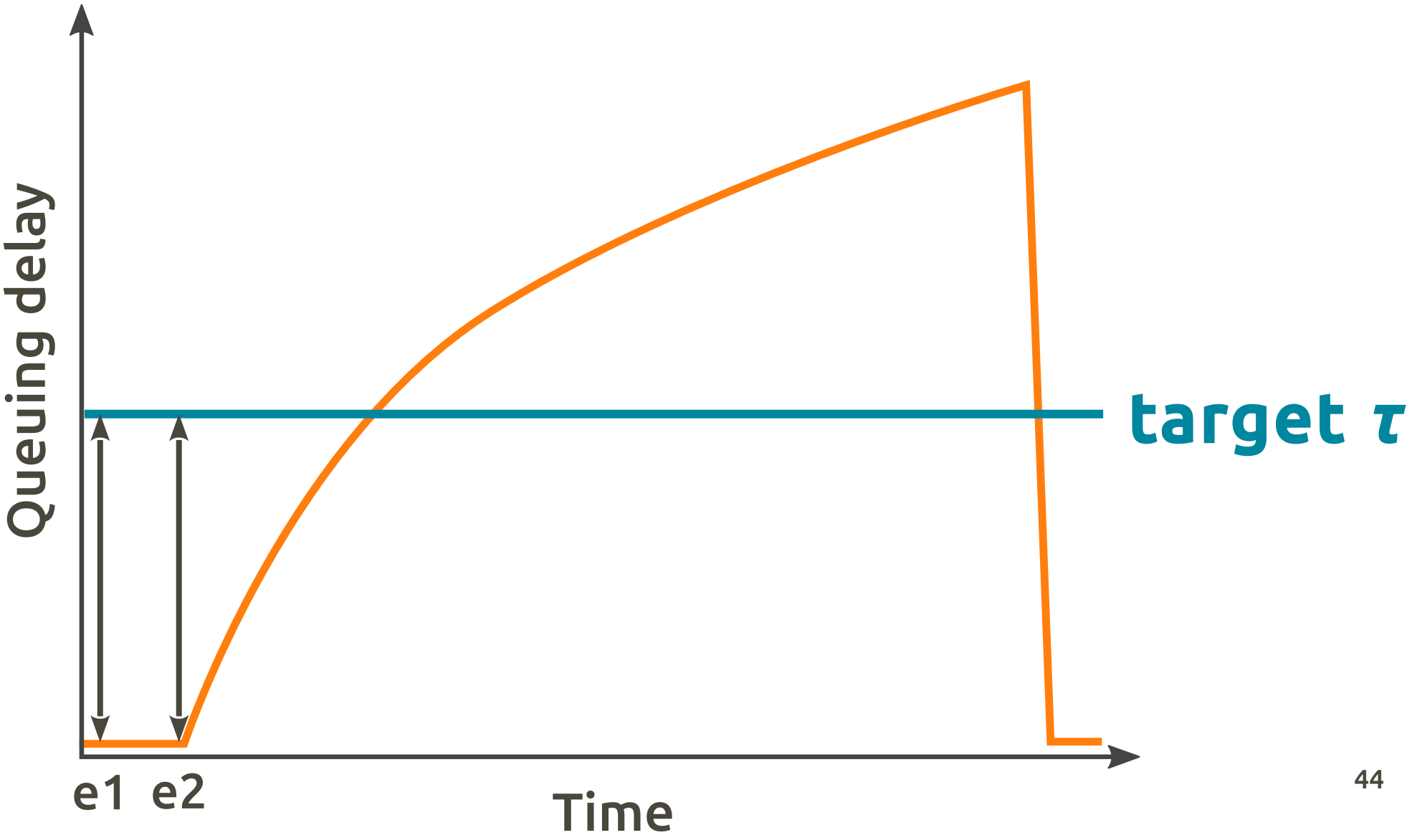
For a shorter description, we could use **linguistic-numeric values**:

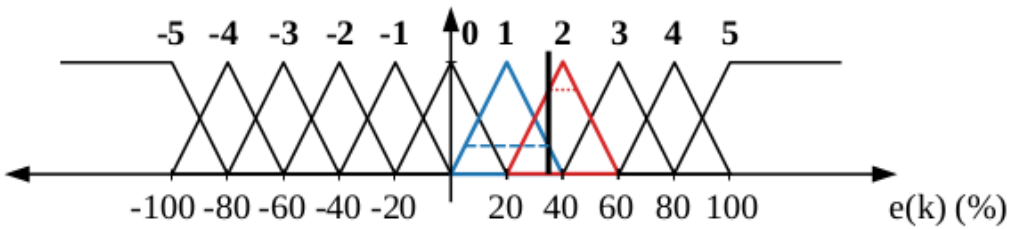
NVL = -5; NL = -4; …; Z = 0; …; PL = 4; VPL = 5;

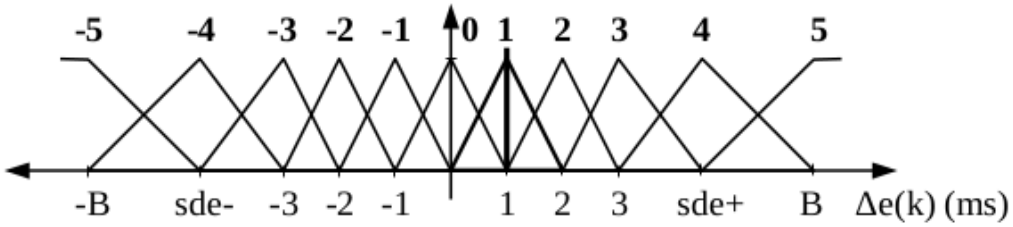
Basic Principle of Fuzzy Logic (ctd)

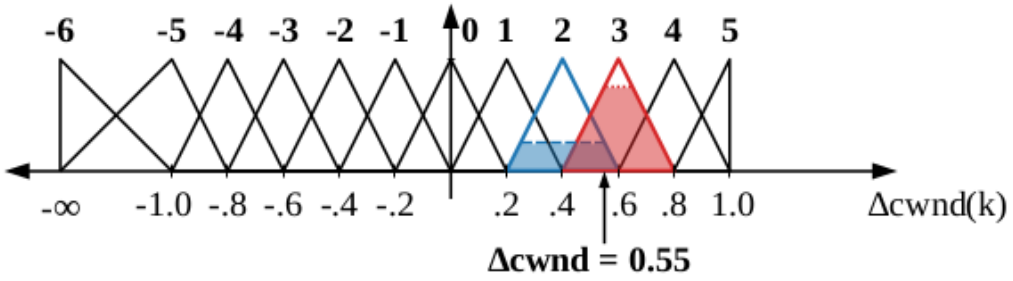
Membership functions quantify the certainty that linguistic variables *(e, Δe, Δcwnd)* **can be classified** as linguistic values (NVL, NL, …, PL, PVL)

If error is positive very large
and Δerror is zero
then Δcwnd is positive very large









$$e(k) = \tau - q(k)$$

$$\Delta e(k) = e(k) - e(k-1)$$

$$\Delta cwnd(k) = \frac{\sum_i b_i \mu_{R_i}(\Delta cwnd(k))}{\sum_i \mu_{R_i}(\Delta cwnd(k))}$$

Example:

$$\Delta cwnd = \frac{(0.4)(0.25)+(0.6)(0.75)}{0.25+0.75} = 0.55$$

Δcwnd		**Δe**										
		-5	**-4**	**-3**	**-2**	**-1**	**0**	**1**	**2**	**3**	**4**	**5**
e	**-5**	-5	-5	-5	-5	-5	-5	-4	-3	-2	-1	**-6**
	-4	-5	-5	-5	-5	-5	-4	-3	-2	-1	0	**-6**
	-3	-5	-5	-5	-5	-4	-3	-2	-1	0	1	**-6**
	-2	-5	-5	-5	-4	-3	-2	-1	0	1	2	**-6**
	-1	-5	-5	-4	-3	-2	-1	0	1	2	3	**-6**
	0	-5	-4	-3	-2	-1	**0**	1	2	3	4	**-6**
	1	-4	-3	-2	-1	0	1	**2**	3	4	5	**-6**
	2	-3	-2	-1	0	1	2	**3**	4	5	5	**-6**
	3	-2	-1	0	1	2	3	4	5	5	5	**-6**
	4	-1	0	1	2	3	4	5	5	5	5	**-6**
	5	0	1	2	3	4	5	5	5	5	5	**-6**

Legend

-6 = NVVL, **-5** = NVL, **-4** = NL, **-3** = NM, **-2** = NS, **-1** = NVS,
0 = Z,
1 = PVS, **2** = PS, **3** = PM, **4** = PL, **5** = PVL

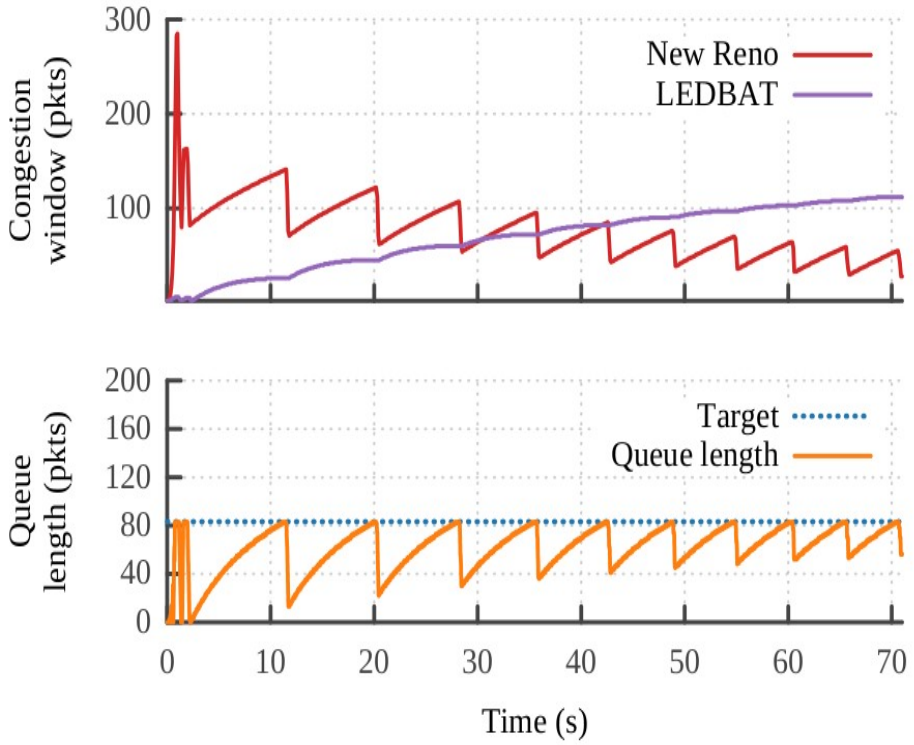
(**P**: Positive, **N**: Negative, **V**: Very,
Z: Zero, **S**: Small, **M**: Medium, **L**: Large)

B: Buffer, **sde-**: smoothed negative Δe, **sde+**: smoothed positive Δe

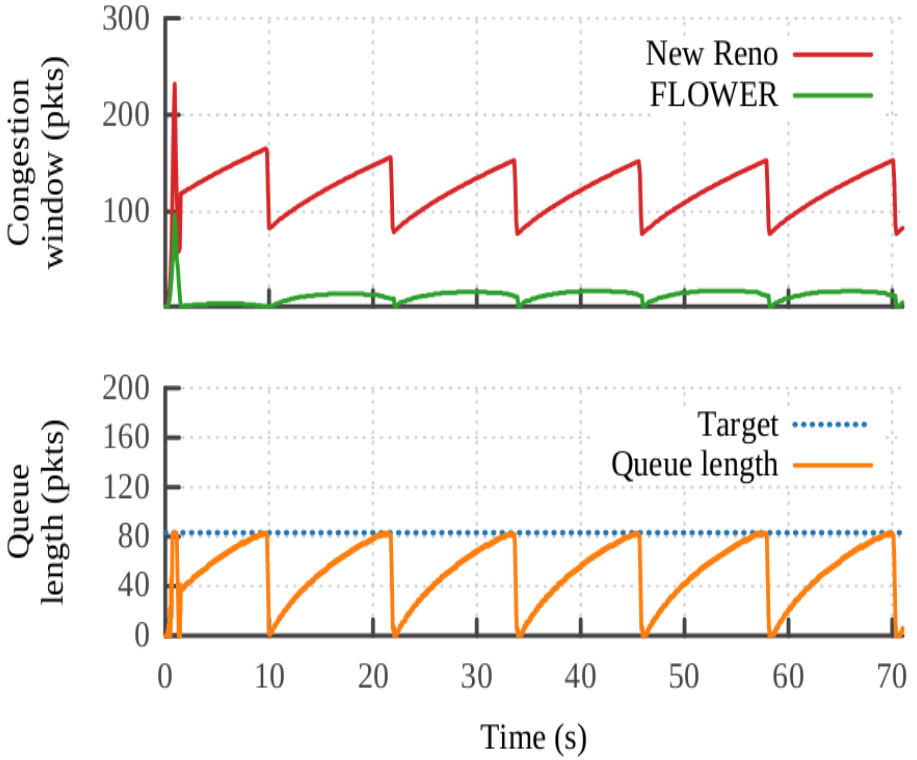
#6

Performance Evaluation of FLOWER

Illustration of FLOWER Behavior



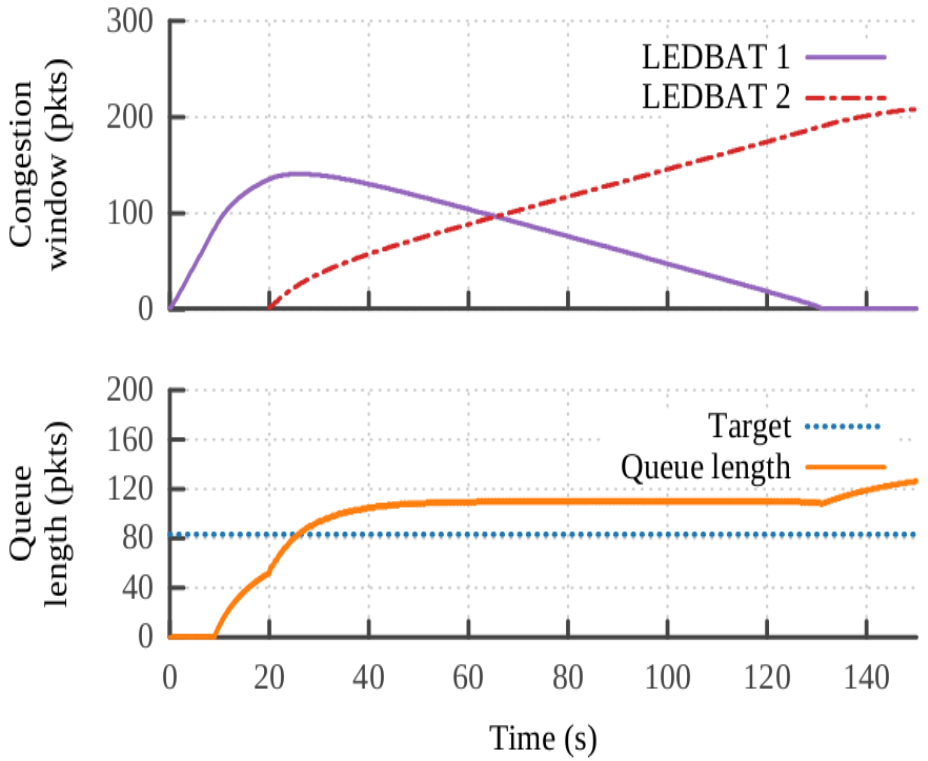
LEDBAT vs. NewReno



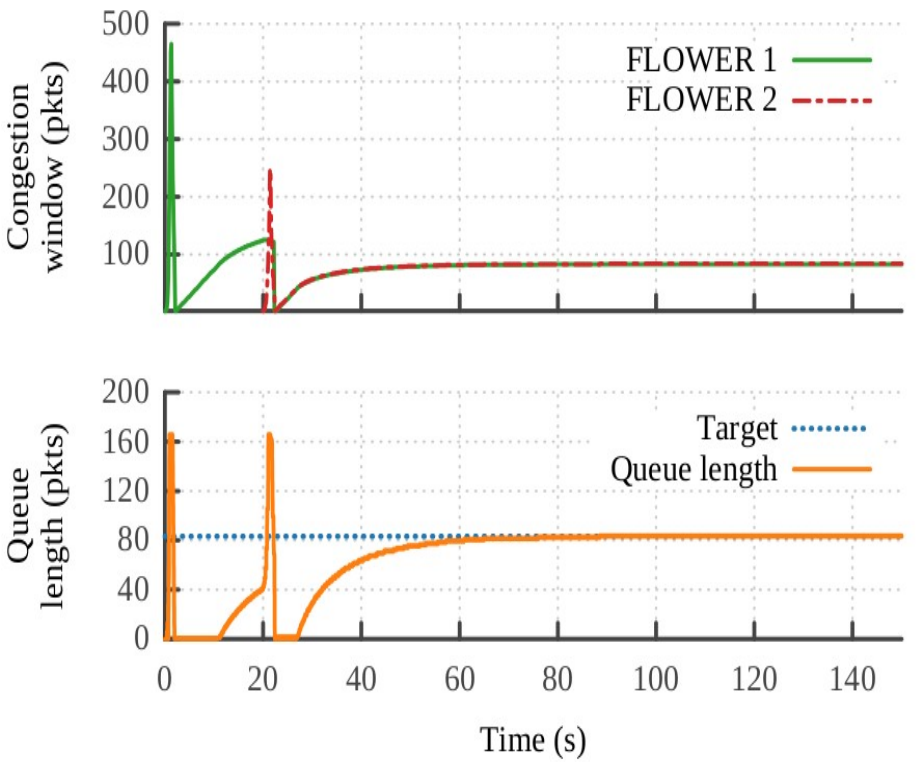
FLOWER vs. NewReno

C = 10 Mb/s, Owd = 50 ms, B = BDP

Latecomer Issue



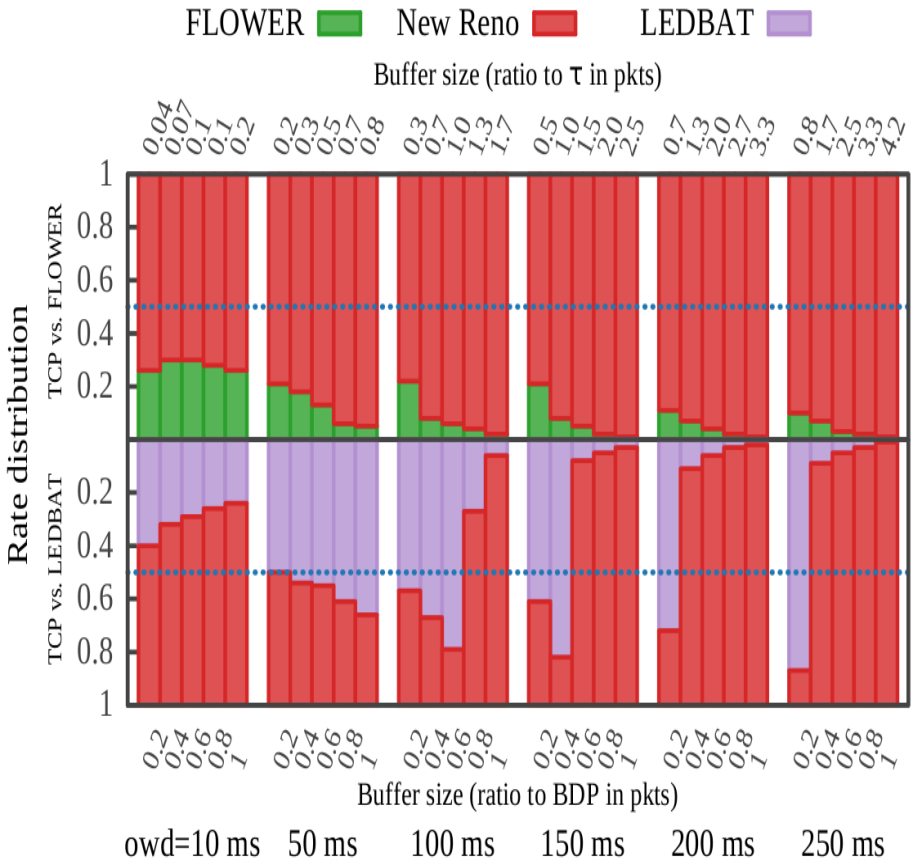
LEDBAT vs. LEDBAT



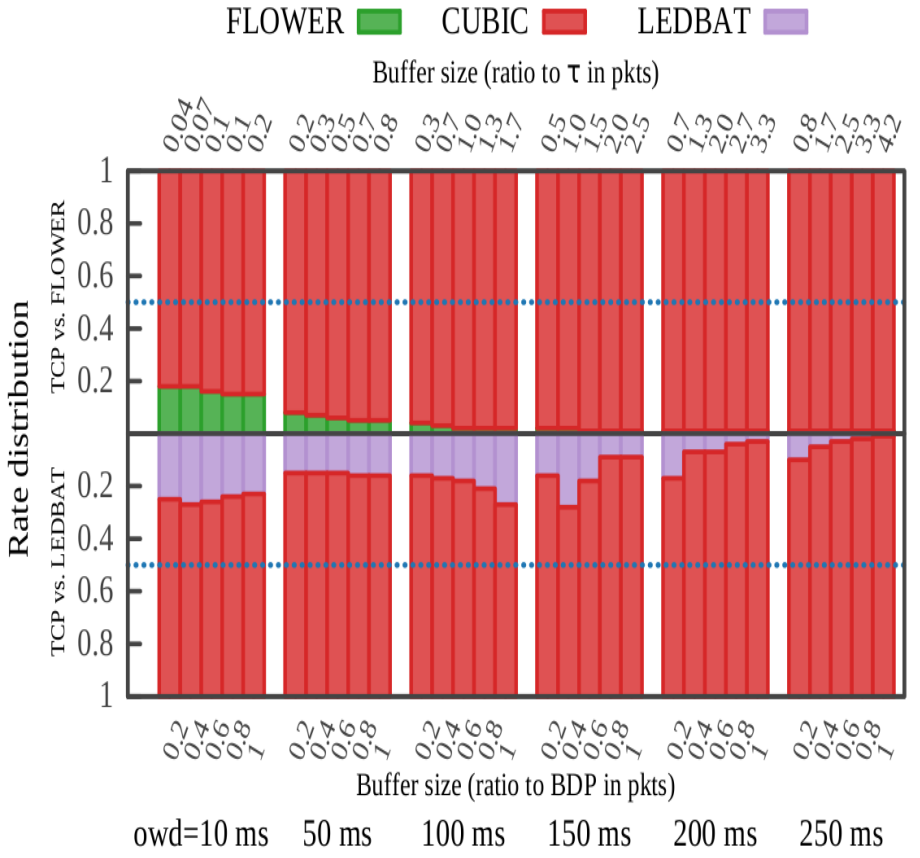
FLOWER vs. FLOWER

C = 10 Mb/s, Owd = 50 ms, B = BDP

LBE Performance



LBE vs. TCP NewReno



LBE vs. TCP CUBIC

C = 10 Mb/s

We have a novel protocol **BUT !!**

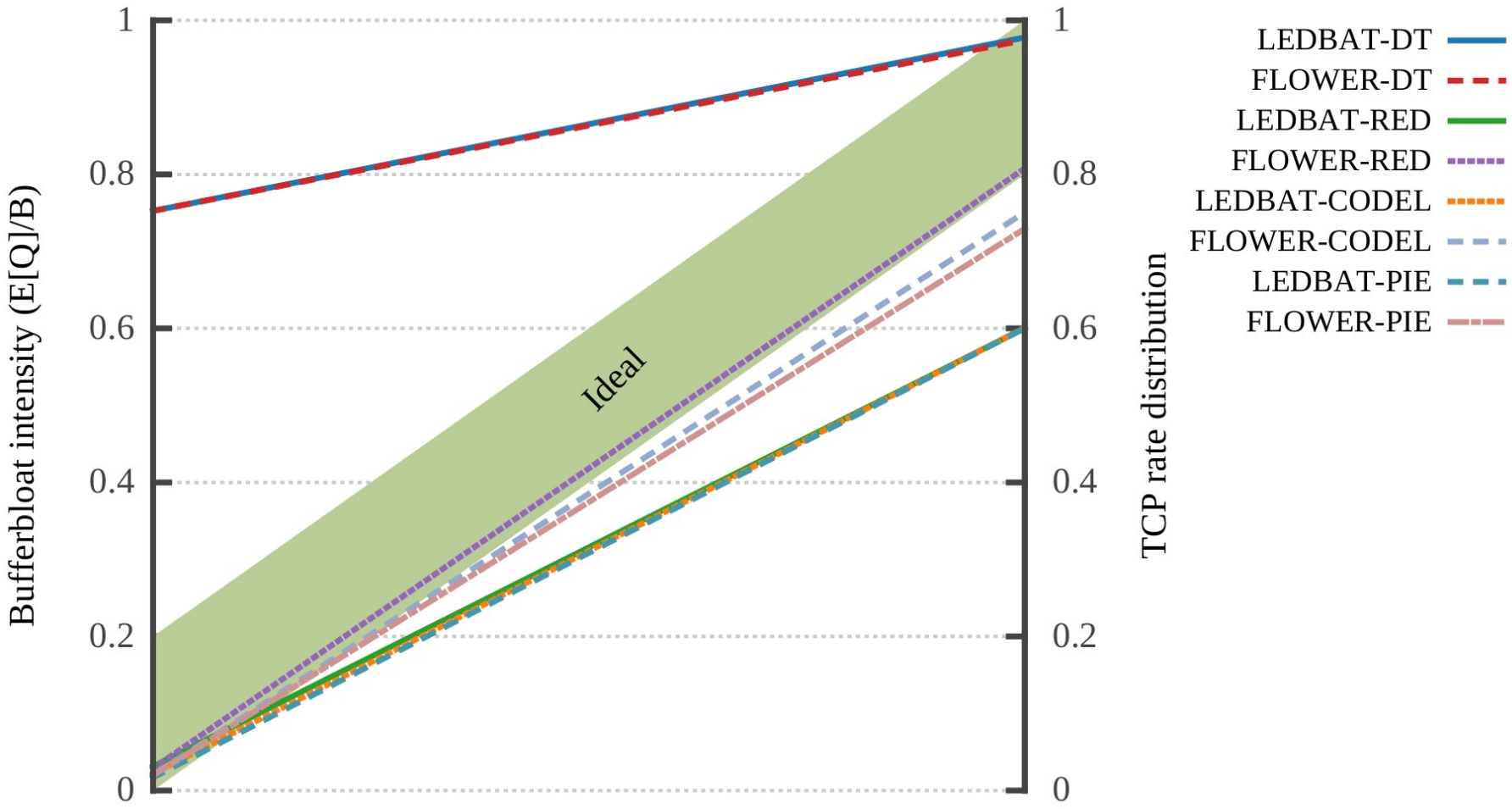
Novel LBE proposals (or transport protocol in general) should consider **the impact of AQMs**

We have a novel protocol **BUT !!**

Novel LBE proposals (or transport protocol in general) should consider **the impact of AQMs**

Bufferbloat (up-to-date problem): the phenomenon of low throughput and high latency caused by excessive buffering

Bufferbloat and LBE



Experiment done following Elsevier Computer Network
Y.Gong, D.Rossi, C.Testa, S.Valenti, M.D.Taht,
"Fighting the bufferbloat: on the coexistence of AQM and low priority congestion control"

#7

Conclusion

LEDBAT tuning is **very difficult** and **highly depends** on the network condition

FLOWER: the first existing solution that **solves both LEDBAT issues**

Implementation in Linux kernel done (under tests)

- **Low computational complexity**

Perspectives

Test FLOWER over CESAR testbed

Novel ideas with **Neural-Fuzzy**

New AQM

Publications

On The Existence Of Optimal LEDBAT Parameters
IEEE ICC 2014

FLOWER — Fuzzy Lower-than-Best-Effort Transport Protocol
IEEE LCN 2015

FLOWER — An Innovative Fuzzy Lower-than-Best-Effort Transport Protocol
Submitted to Elsevier Computer Networks

Non-Renegable Selective Acknowledgments (NR-SACKs) for TCP
To be submitted