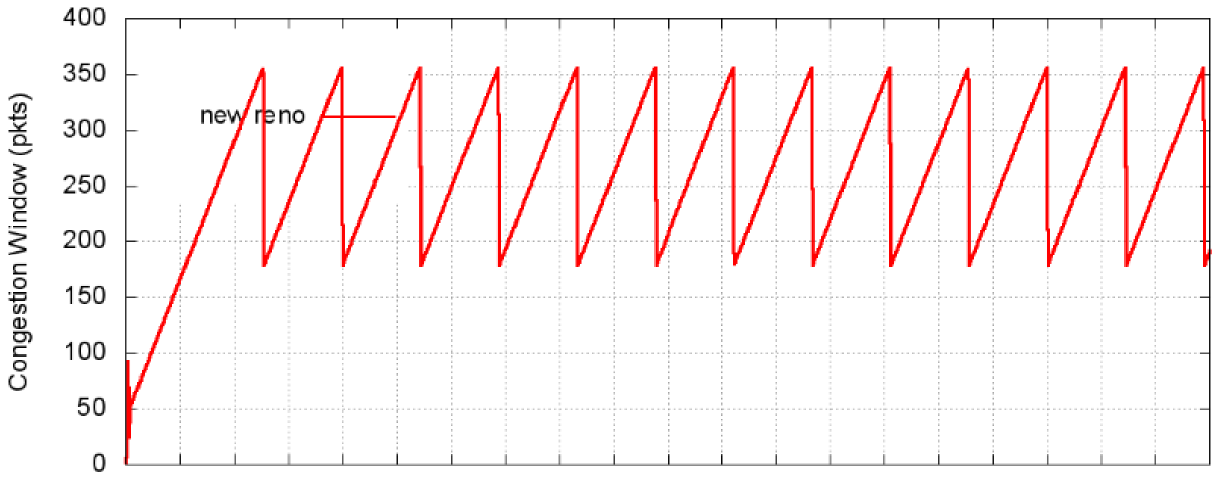
When Fuzzy Logic meets LEDBAT: FLOWER, a *F*uzzy *LOW*ER than Best-*E*ffo*R*t Protocol

Si Quoc Viet TRANG

Objective

- TCP New Reno: AIMD (Addictive Increase/Multiplicative Decrease) congestion control → the sawtooth behavior results in the usage of ~75% of the link capacity
- Over the satellite link, the capacity must be fully used to optimize the cost of the link

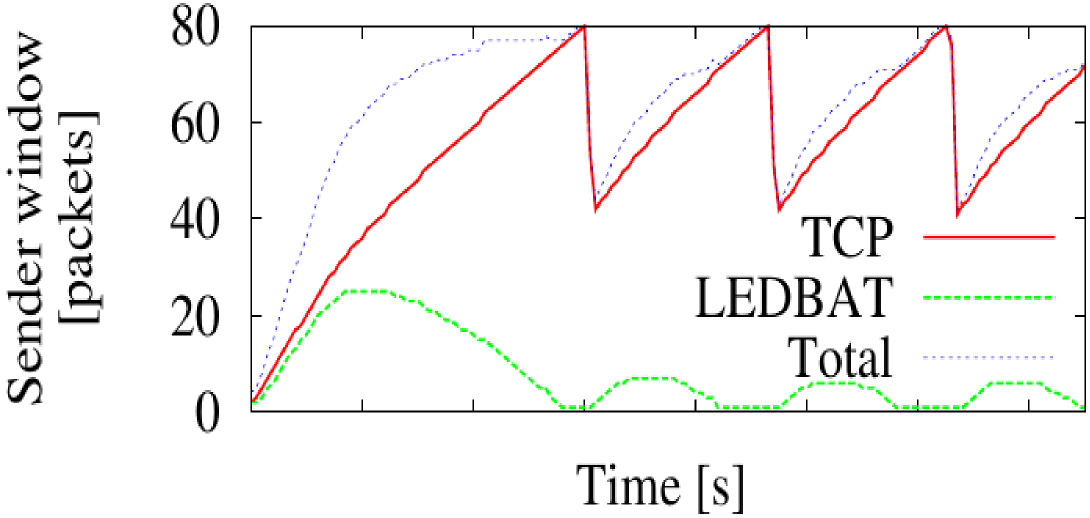


Objective

- Our objective is to grab this unused link capacity by studying algorithm and studying new protocols that enable a LBE service
- What does LBE mean?
  - Take the remaining capacity without disturbing commercial traffic
  - Example: background backup/update, P2P traffic, measuring or signaling traffic

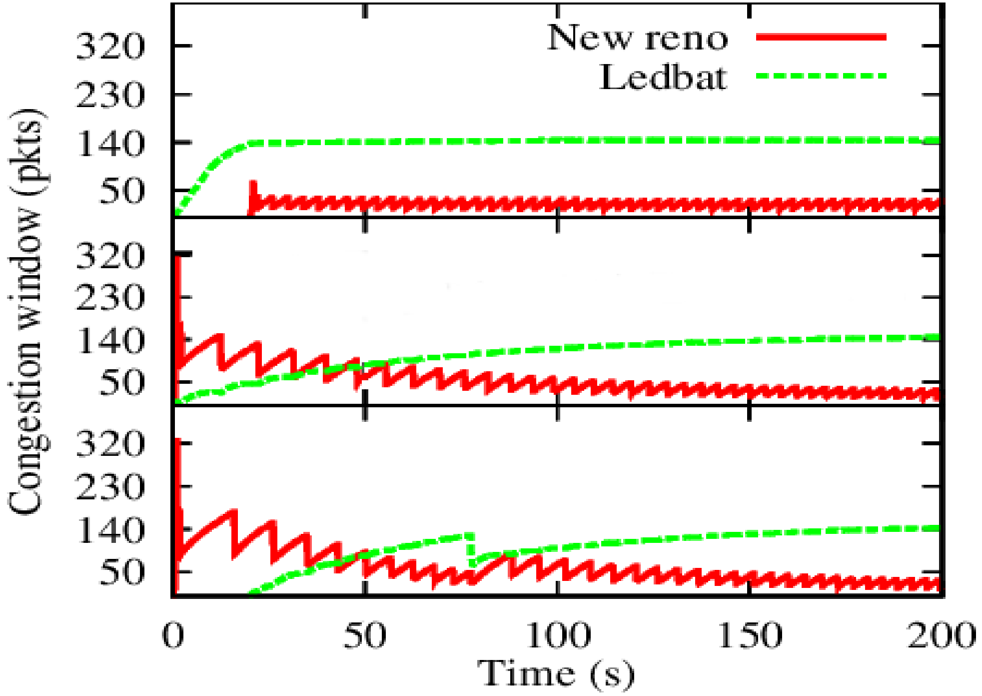
LEDBAT overview

- Ledbat is the most commonly used LBE protocol
- Designed for data transfer with a priority « Less than Best Effort »
- Reduces the rate when it detects that the delay is beyond a threshold (target queuing delay)
- Responds to congestion before standard TCP
- How does it perform over long delay link?
- Is LEDBAT a good candidate?



LEDBAT problems

- We have identified several problems
  - LEDBAT has been revealed to be difficult to configure
  - its tuning highly depends on the network condition
  - may become more aggressive than TCP in case of misconfiguration
  - might not be used safely over any networks
- **“On the existence of optimal LEDBAT parameters” (ICC 2014)**



Towards an adaptive LBE protocol

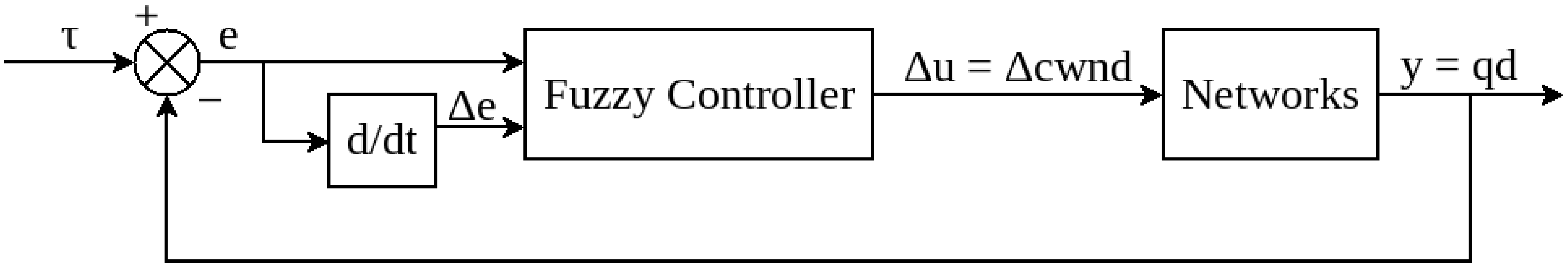
- Ledbat does not have an optimal configuration and needs to be tuned as a function of the network state
- Ledbat is not adaptive and only implements a P-type controller to control the sending rate as a function of the delay. Both parameters (target and gain) are fixed
- One possible solution would be to implement an adaptive functionality inside this P-type controller but such adaptation would require a fine grained analytical model

Towards an adaptive LBE protocol

- Our goal is to overtake this difficulty by using a fuzzy algorithm to perform such task
- Our proposal, named FLOWER :
  - Aims at proposing a novel LBE transport protocol based on a fuzzy logic algorithm
  - Aims at reacting better than Ledbat front to congestion event

Fuzzy LBE Controller

- The controller goal is to limit the queuing delay to a target delay τ by controlling the number of packets sent over the networks
- By measuring the one-way delay (OWD), the controller infers the current queuing delay
- By comparing the inferred queuing delay with the target delay, the controller adjusts the congestion window *cwnd* using fuzzy logic



How to control the queuing delay?

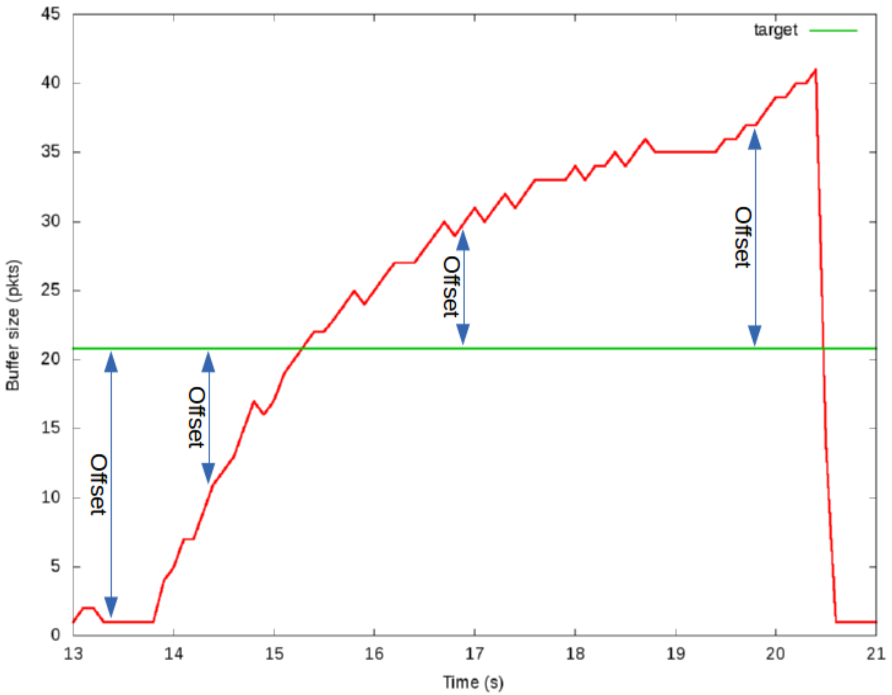
- Inputs:
  - Error:

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

  - Change of error (gradient of error):

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

- Output: change of cwnd ($\Delta cwnd(k)$)
- The inputs and the output are related by fuzzy rules



Linguistic variables / values

- *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)

- Example:
    - "*e* is NVL" means "e is negative very large"
- For a shorter description, we could use linguistic-numeric values:

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

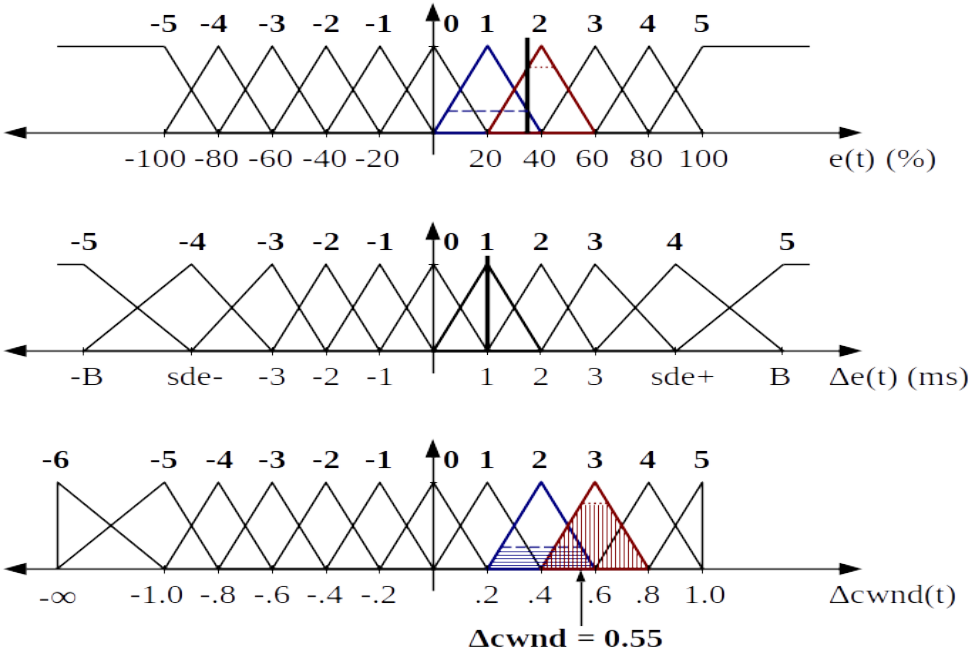
Rules

- 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
- Examples:
  - If e is PVL and Δe is Z then Δcwnd is PVL
  - If e is NVL and Δe is NVL then Δcwnd is Z

Membership functions (MF)

- The controller uses membership functions to quantify the certainty that linguistic variables *(e, Δe, Δcwnd)* can be classified as linguistic values (NVL, NL, …, PL, PVL)

MF and Rule Table



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

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

$$\Delta cwnd(t) = \frac{\sum_i b_i \mu_{premise(i)}}{\sum_i \mu_{premise(i)}}$$

Example:

$$\Delta cwnd = \frac{(0.2)(0.25) + (0.4)(0.75)}{0.25 + 0.75} = 0.55$$

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

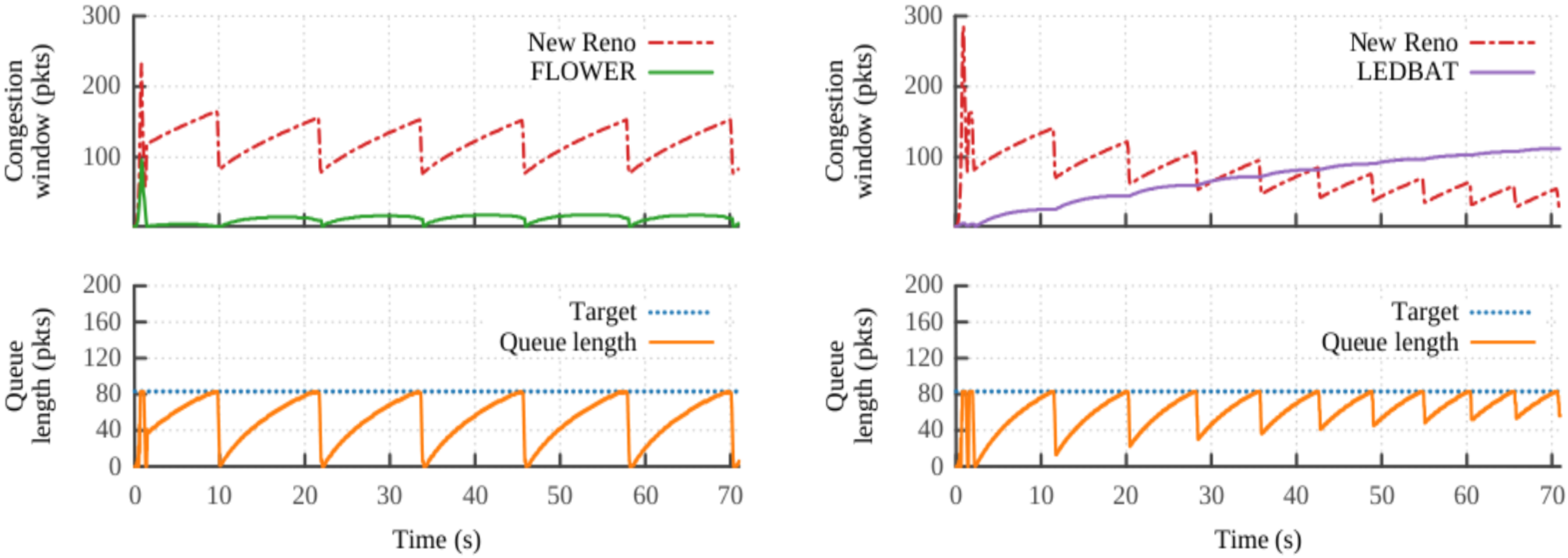
(**P**: Positive, **N**: Negative, **V**: Very,
Z: Zero, **S**: Small, **M**: Medium, **L**: Large)

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

Simulation 1

- 1 flows TCP v.s. 1 flows LBE
- TCP = New Reno
- LBE = {LEDBAT, FLOWER}
- C = 10 Mb/s
- Owd = 50 ms
- B = BDP
- Duration = 75 ms
- TCP and LBE flows start at the same time

1 LBE – 1 New Reno



(a) FLOWER vs. TCP New-Reno

(b) LEDBAT vs. TCP New-Reno

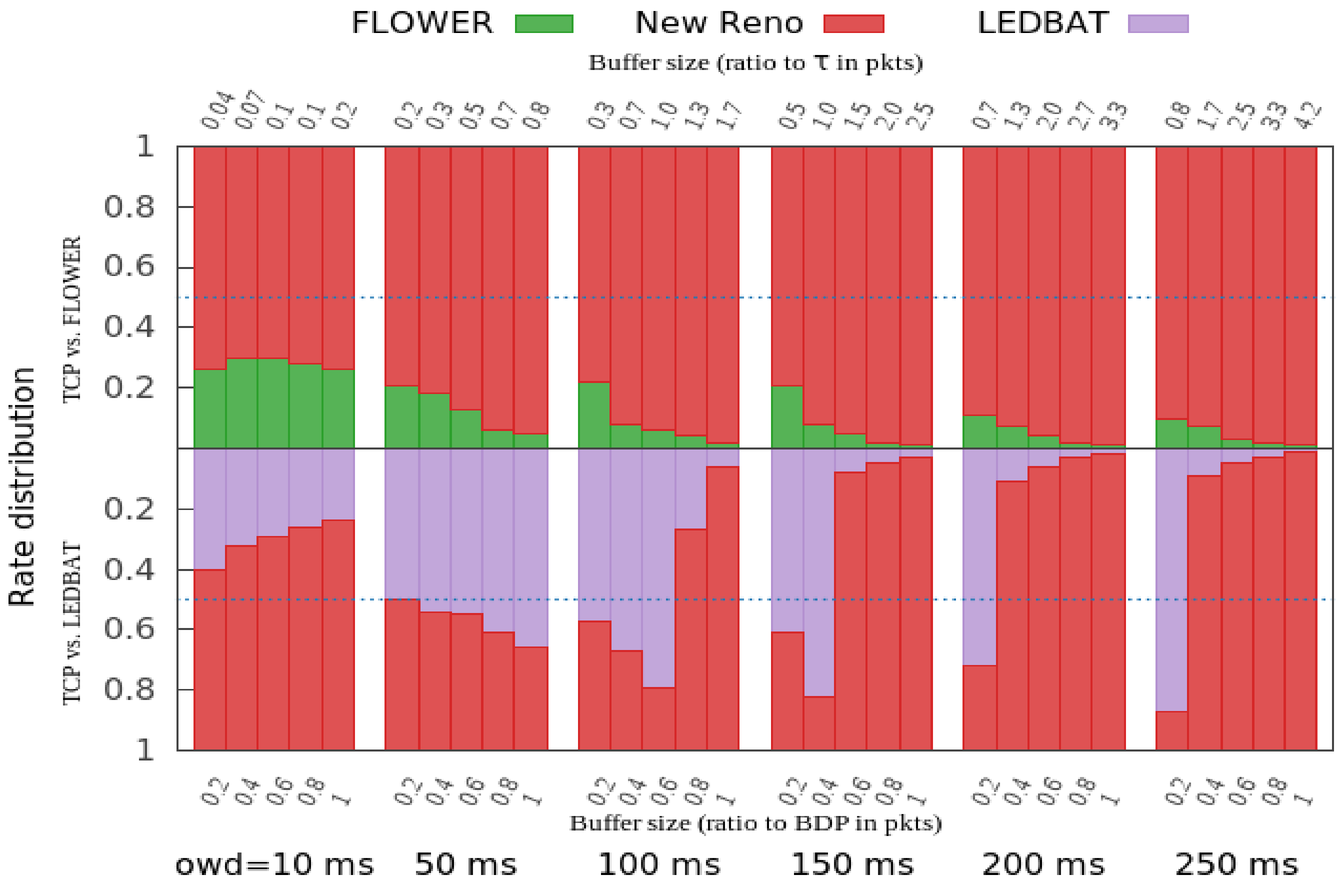
Simulation 2

- 5 flows TCP v.s. 5 flows LBE
- TCP = New Reno
- LBE = {LEDBAT, FLOWER}
- C = 10 Mb/s
- Owd = {10, 50, 150, 200, 250} ms
- B = {0.2, 0.4, 0.6, 0.8, 1.0} x BDP
- Duration = 1200 ms
- TCP flows start consecutively at the beginning
- LBE flows start randomly between 350 s and 450 s

Simulation 2

- For each scenario, the simulation runs 10 times
- Using “*rate distribution*” as metric.
- The final measured value is the mean of the 10 metric values.

5 LBE – 5 New Reno



Conclusion

- There is still a place on the network for LBE service
- LEDBAT tuning is very difficult and highly depends on the network condition
- Our proposition – FLOWER – is a promising alternative of LEDBAT