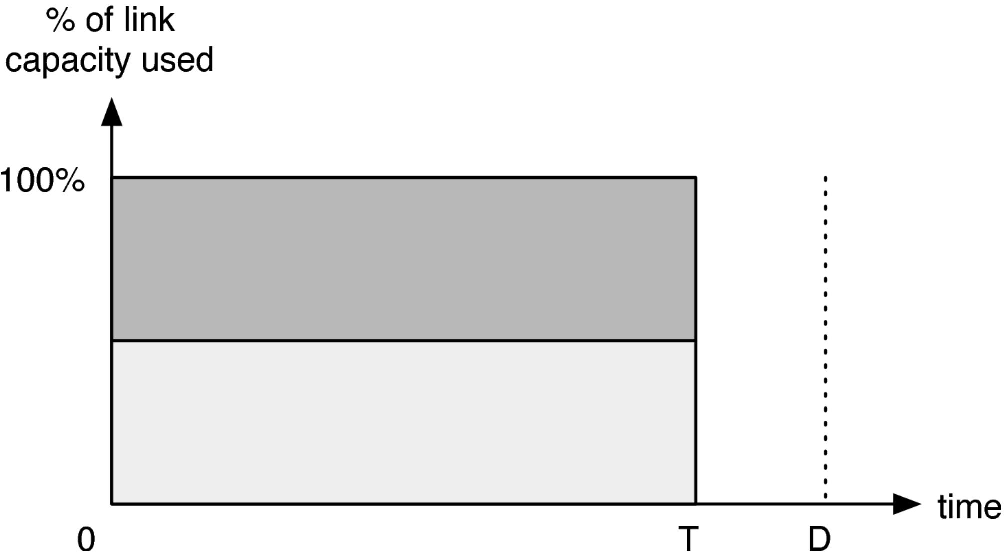
Evaluating CDG as a Deadline-Aware Less-than-Best-Effort Transport Service

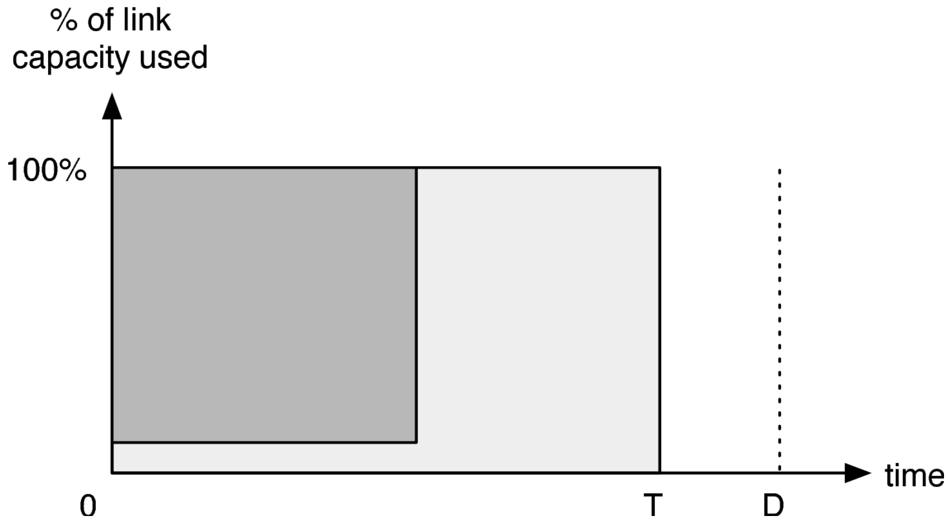
Tor-Christian Tangenes, David Hayes, Andreas Petlund, David Ros
email: tctangen@student.matnat.uio.no, {davidh,apetlund,dros}@simula.no

Horizon 2020
European Union funding
for Research & Innovation

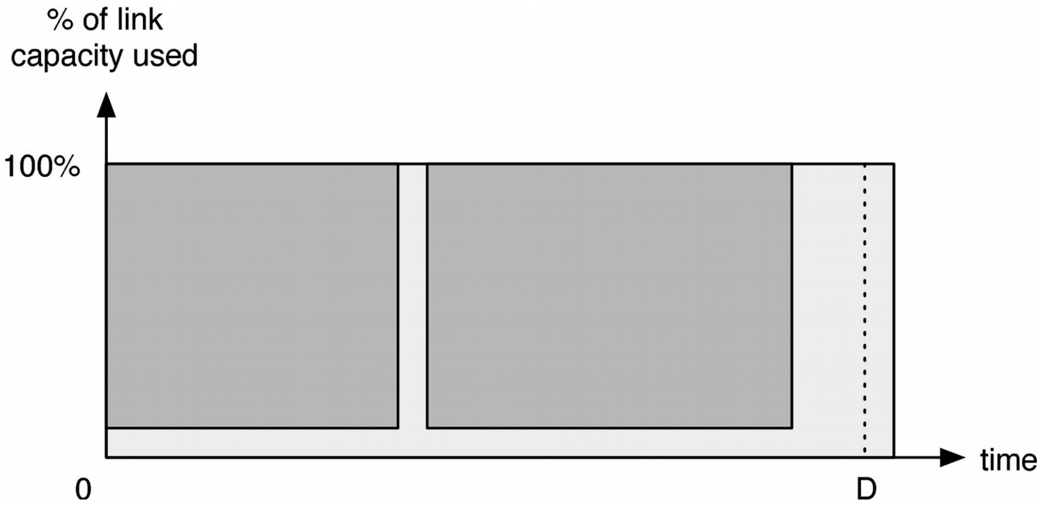
What is Deadline-Aware Less-than-Best-Effort?



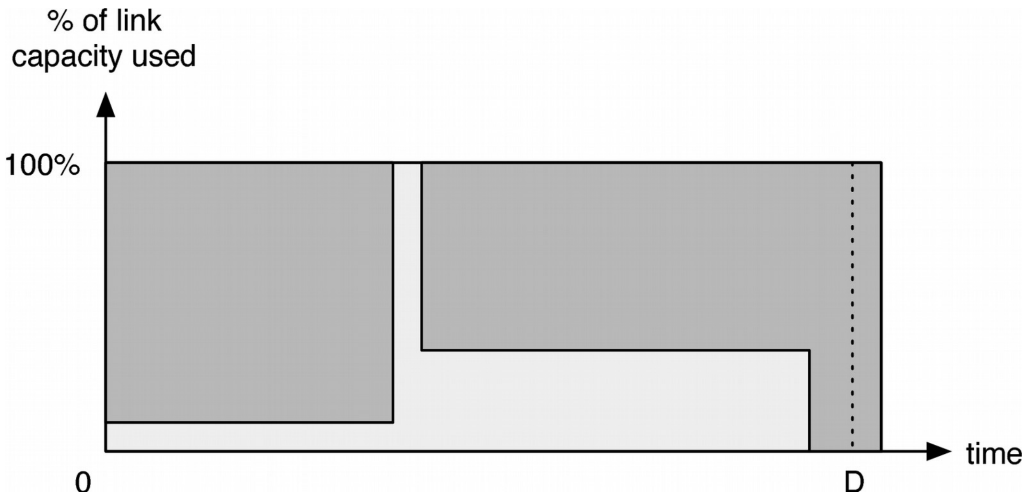
Dark grey: BE, light grey: BE



Dark grey:BE, light grey: LBE



Dark grey: BE, light grey: LBE



Dark grey: BE, light grey: DA-LBE

Background and motivation

- LBE: In the presence of BE, yield capacity toward BE. Otherwise, use left-over capacity
- DA-LBE: Dynamically adapting, deadline aware. Transmission aggressiveness depends on:
  - Available spare capacity
  - A specified soft deadline (ideal time for transfers to be completed within)
- DA-LBE example target application:

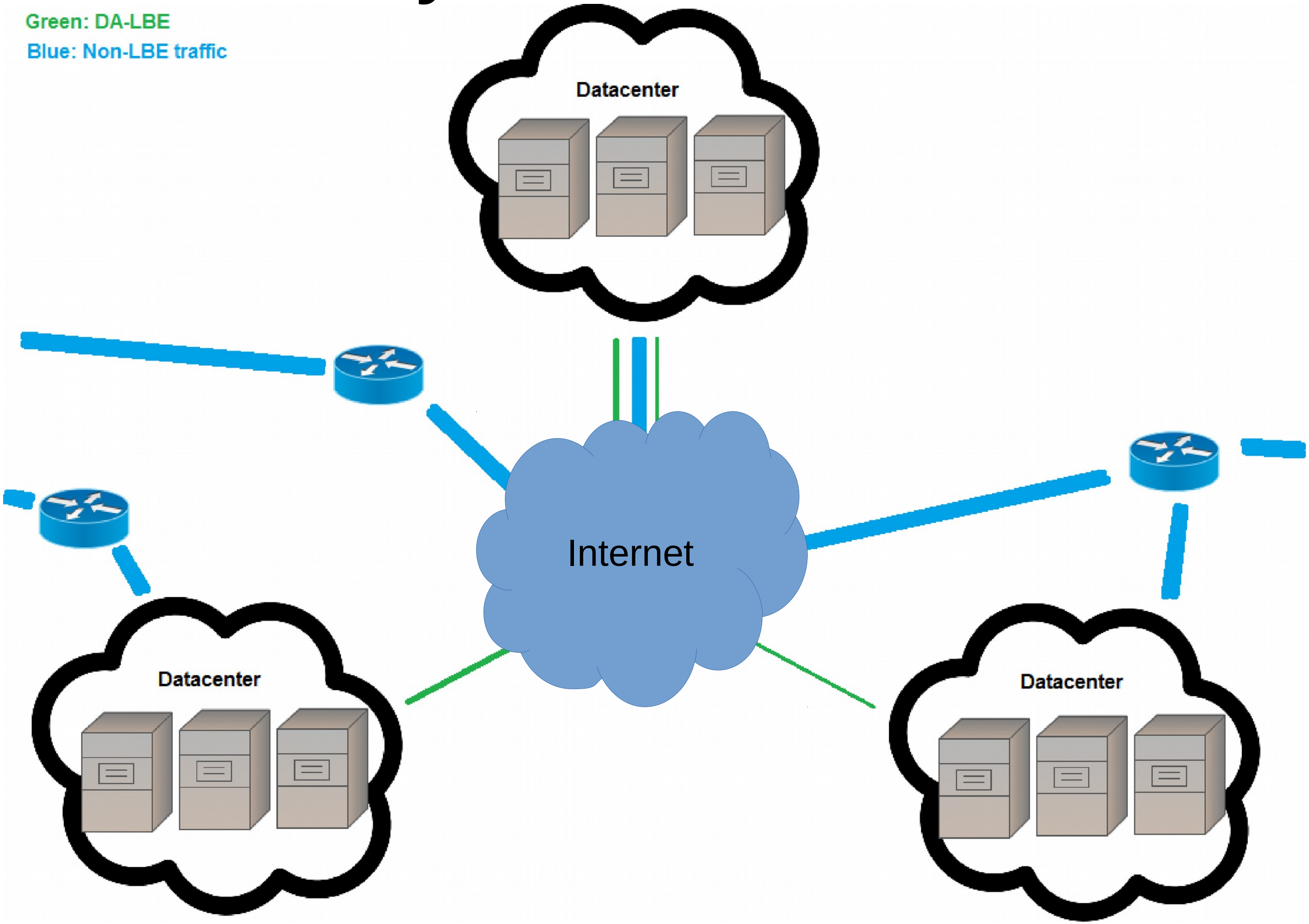
  - Synchronization between CDN data centres

  - Distributed backup tasks

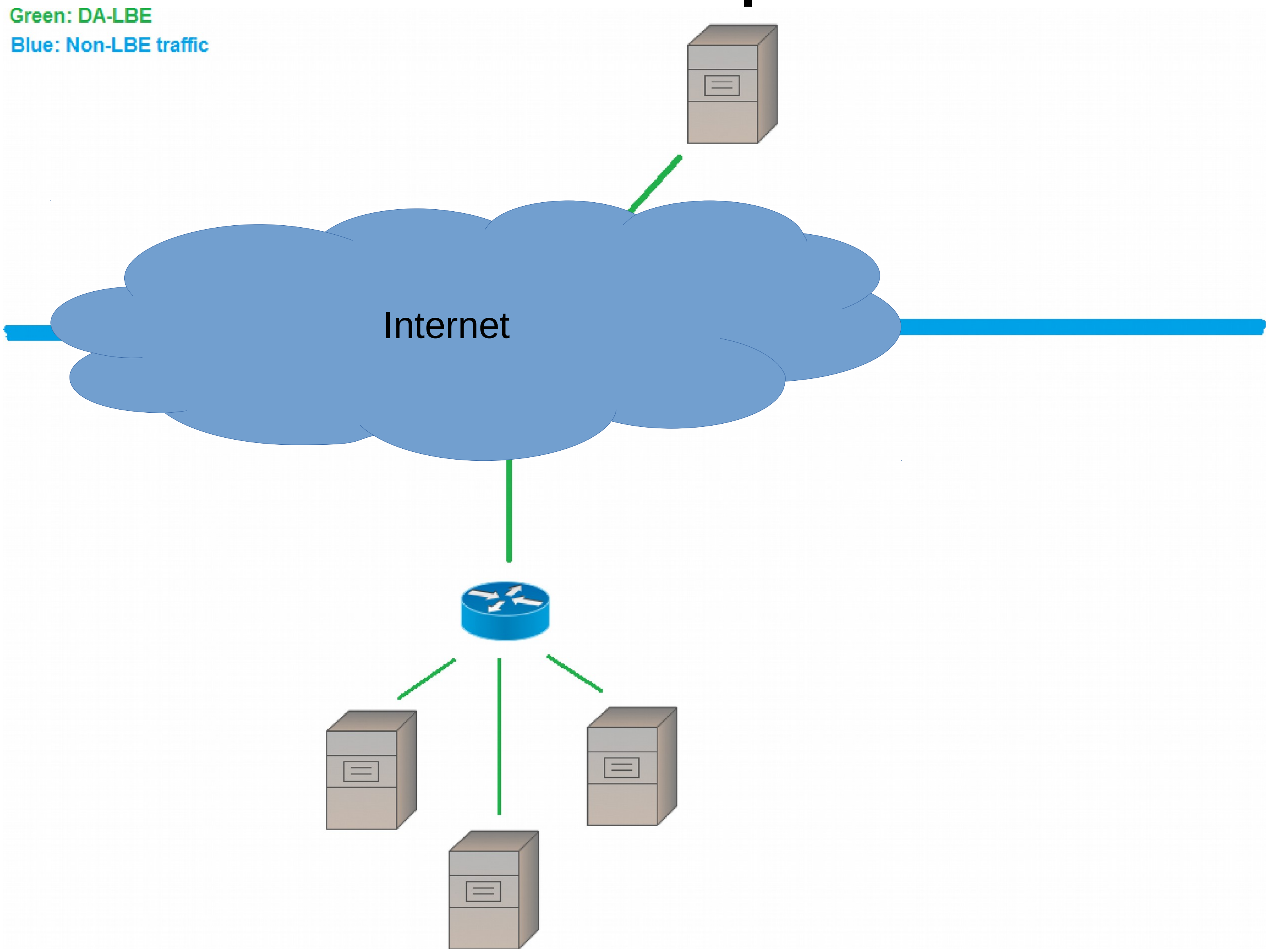
DA-LBE in a future Internet transport stack

- Possible incarnations in a future Internet transport stack
  - New transport protocols
  - New TCP congestion control (requires additional information)
  - “Meta” protocol that works on top of TCP
- Our work focus on the meta protocol approach
  - See “A framework for less than best effort congestion control with soft deadlines” for how this can be done
- NEAT: A New, Evolutive API and Transport-Layer Architecture for the Internet
  - Provides API to applications to make several networks services available
  - Let application take advantage of new network functionality without having to be rewritten

Use-case example: CDN synchronization



Use-case example: Distributed backup



Designing DA-LBE

- Design from scratch or use an existing available CC. The latter option would ease deployment
- A DA-LBE should be:

  - Able to exhibit LBE behaviour
  - Have tunable transmission aggressiveness
  - Be available in a stock operating system
- Several delay-based congestion controls have already been designed for carrying LBE traffic.
  - LEDBAT
  - TCP-Nice
  - TCP-LP
- Delay-threshold CC problem: Base delay estimation, latecomers advantage

CAIA Delay-gradient Congestion Control

- Probabilistic delay gradient congestion control

$$P[\text{Backoff}] = 1 - e^{-\bar{g}/G}$$

- The moving average $\bar{g}$ is calculated from the gradients to reduce measurement noise.
- The probability *P* of CDG backing off (yielding) increases exponentially with increasing values of *g*.
- Uses heuristics to compete with loss-based CC
- The *P* can be scaled with an exponential weight G (set to 3 by default)

Why use CDG for DA-LBE

- CDG has knobs to enable LBE-mode (disables co-existence heuristics)
- The G scaling parameter can be used to tune CDG's aggressiveness
  - Decrease G → Increases LBE behaviour
  - Increase G → Towards Reno behaviour
- Currently available in:

  - Linux
  - FreeBSD
- Does not suffer from latecomers advantage

Evaluating the tunability of CDG’s aggressiveness

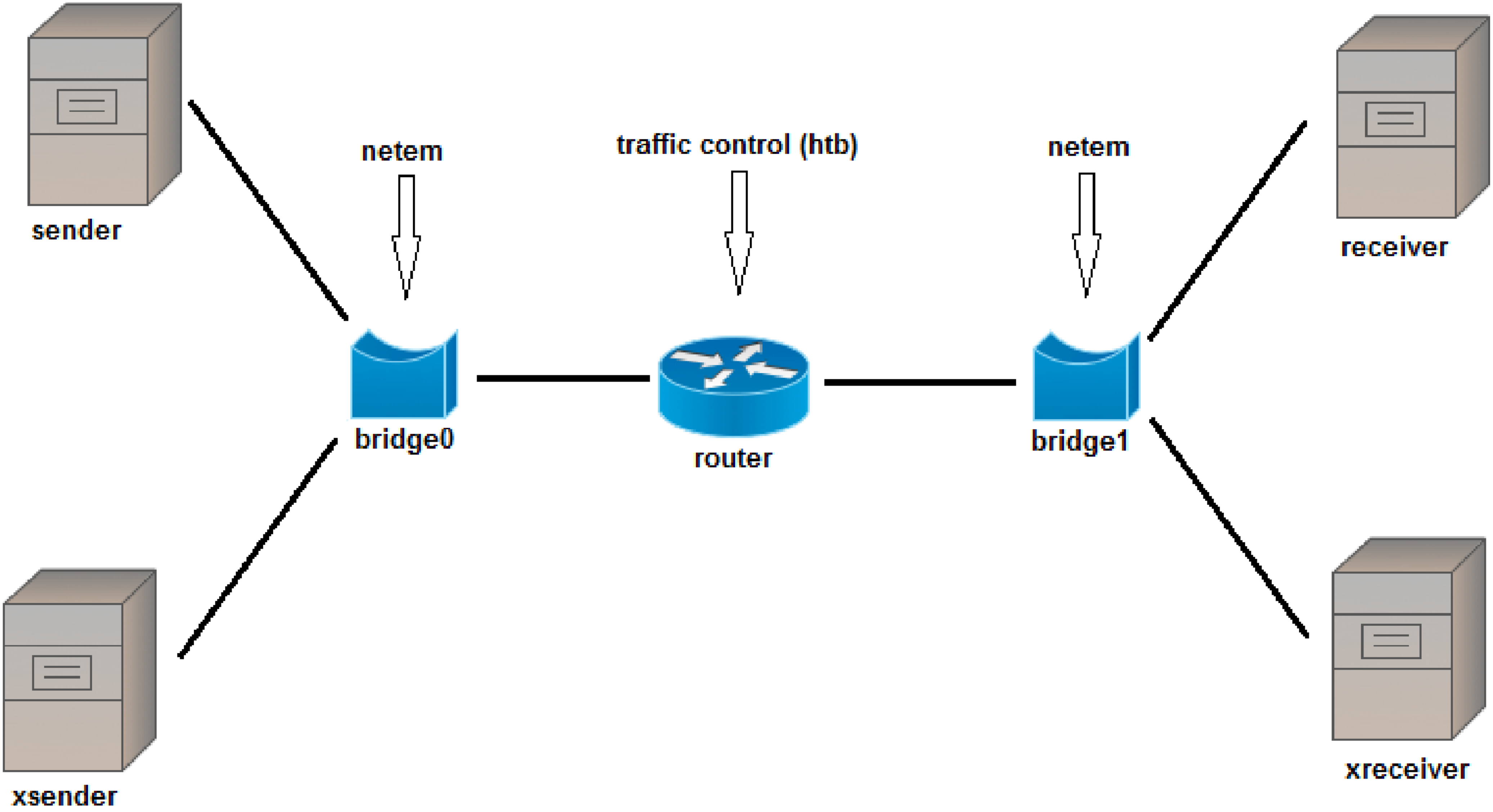
- Compare Linux version of CDG against Linux default CUBIC CC
- We use throughput share as our metric for LBE behaviour

  - How does tuning G affect CDG’s LBE behaviour?

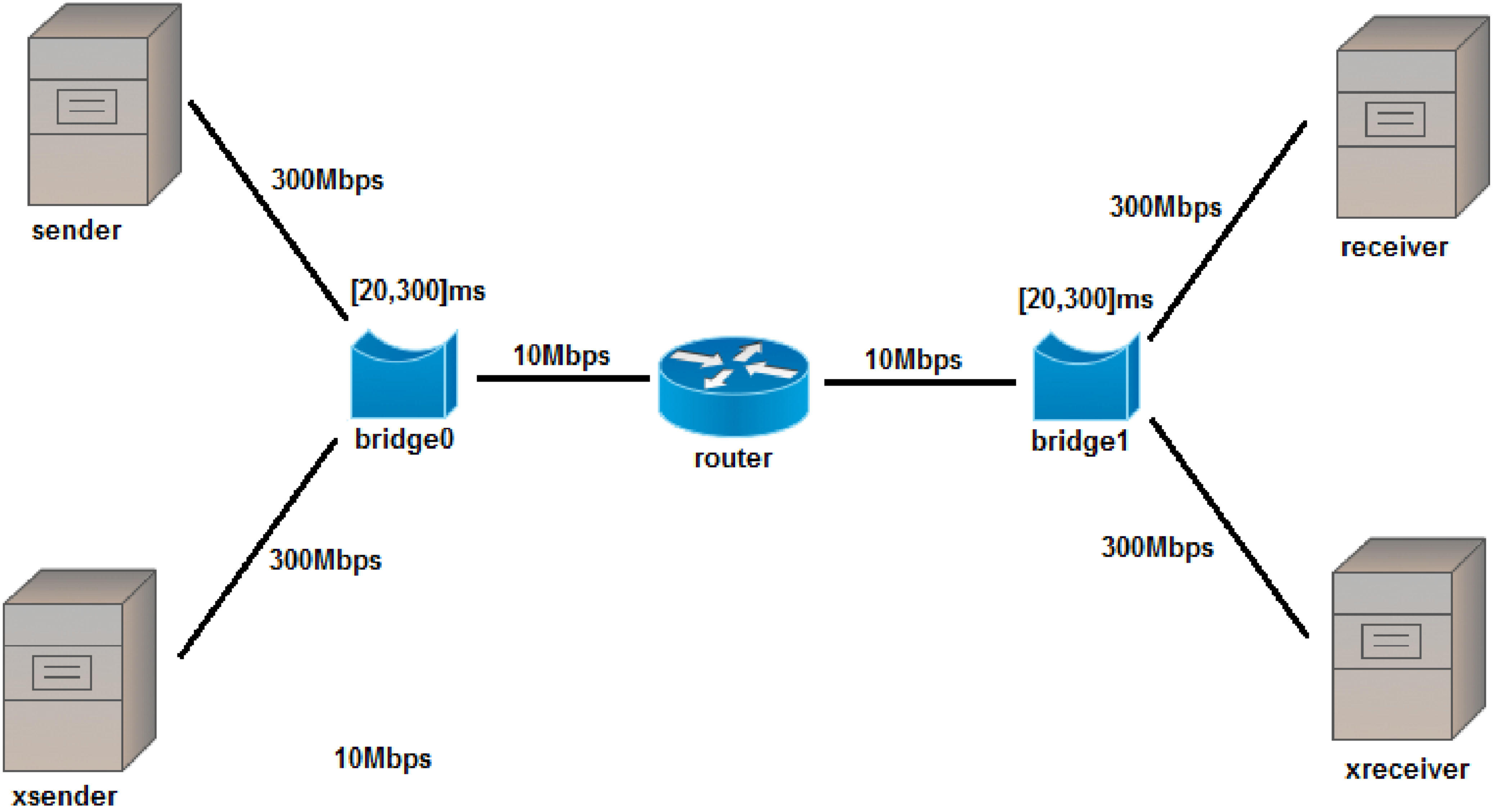
  - How does RTT affect the range to which CDG can be tuned?
- We turn off CDG’s co-existence heuristics to evaluate the LBE characteristics
- CUBIC use default parameters
- Parameter space:

  - 20 values of G, ranging from 0.1 to 120
  - 7 different values of RTT, ranging from 20-300ms

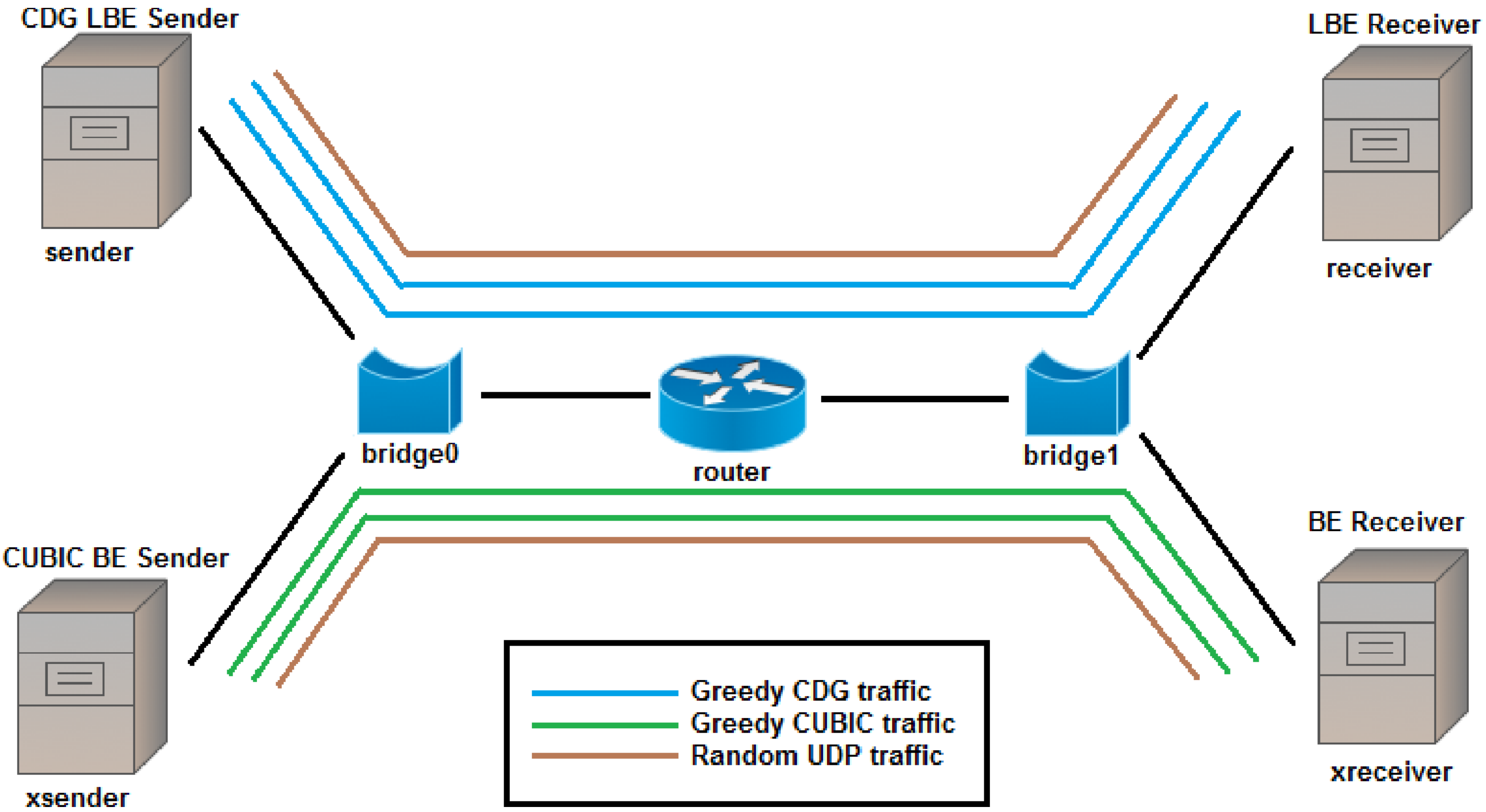
Experimental set-up



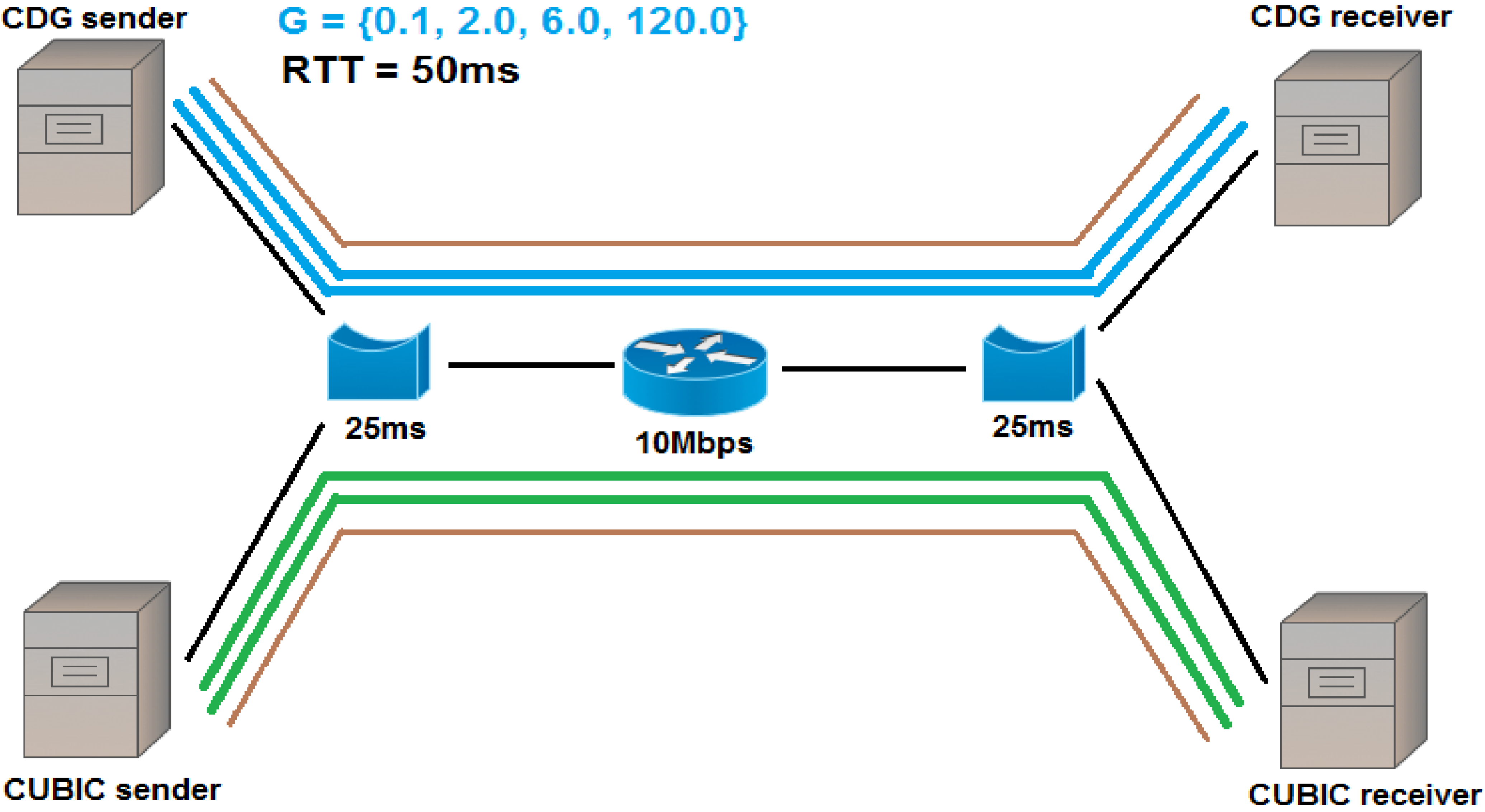
Experimental set-up

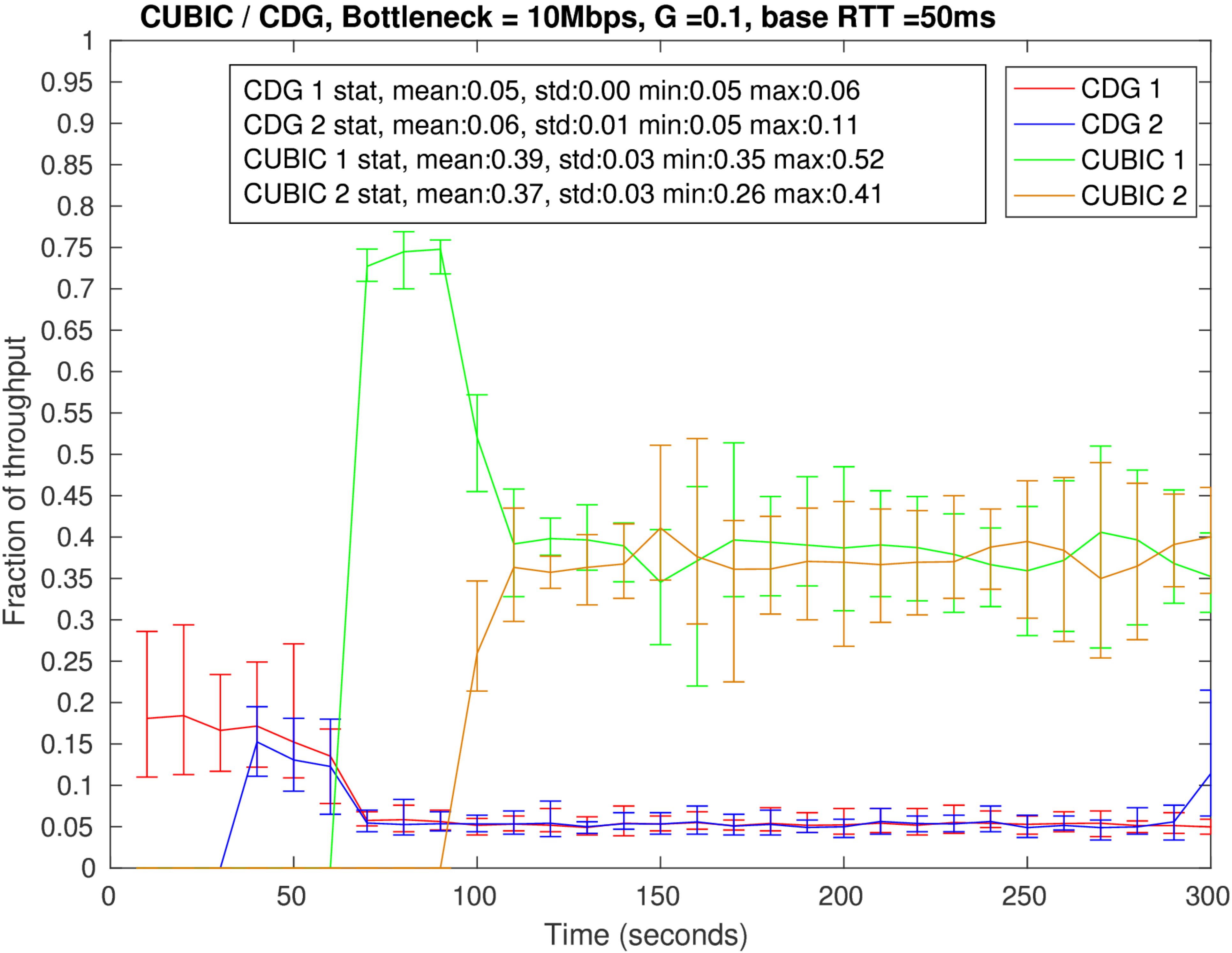


Experimental set-up

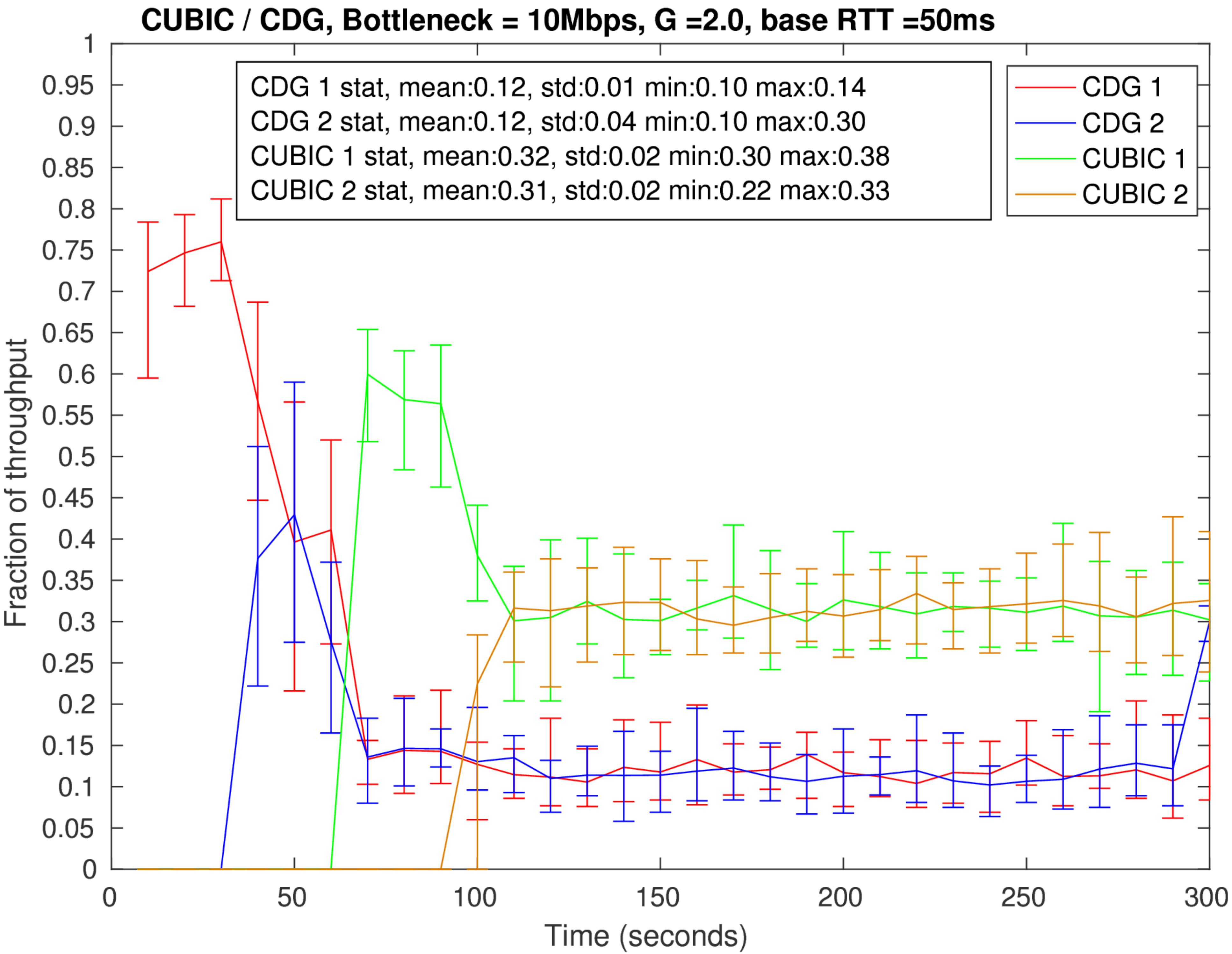


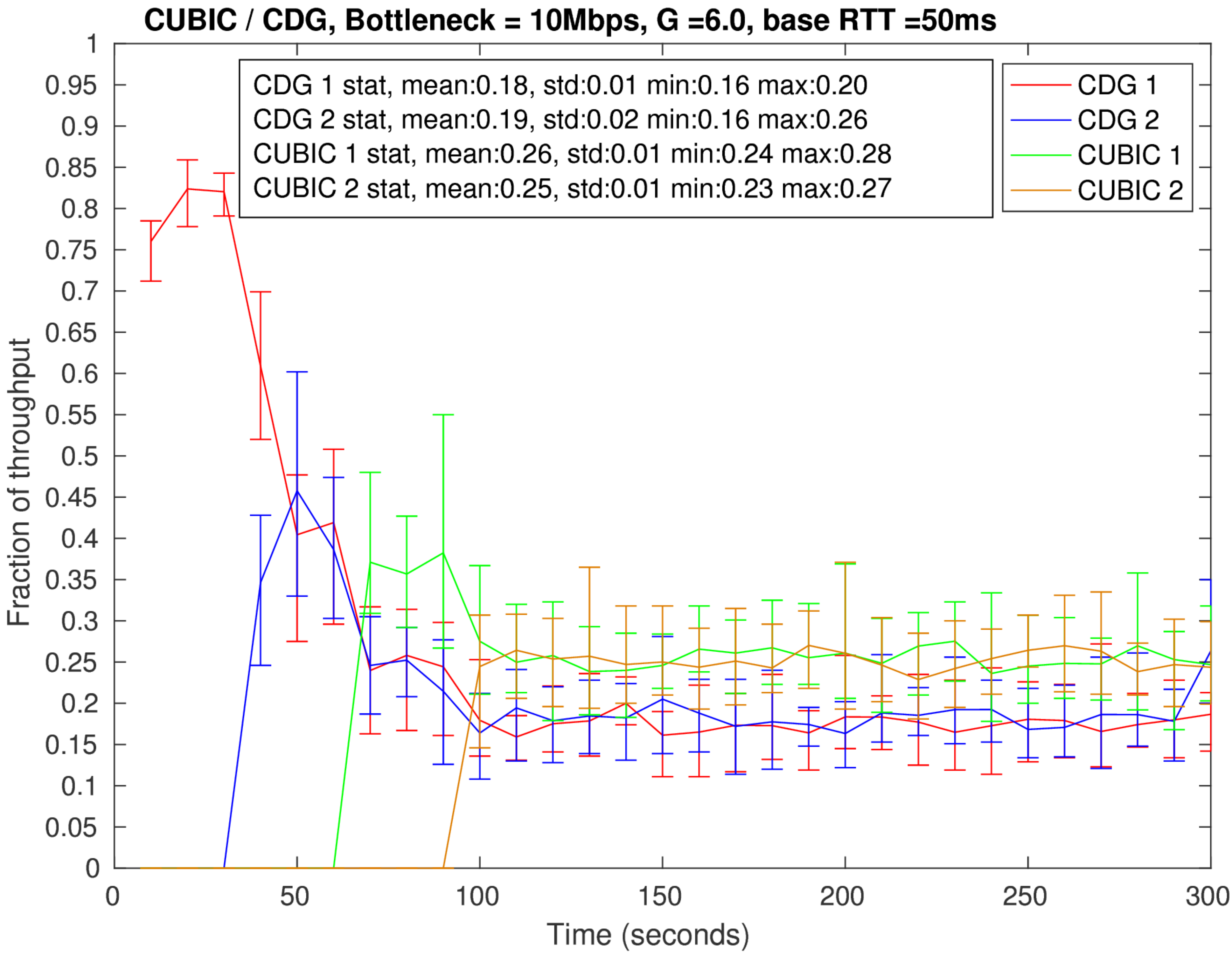
Experimental set-up

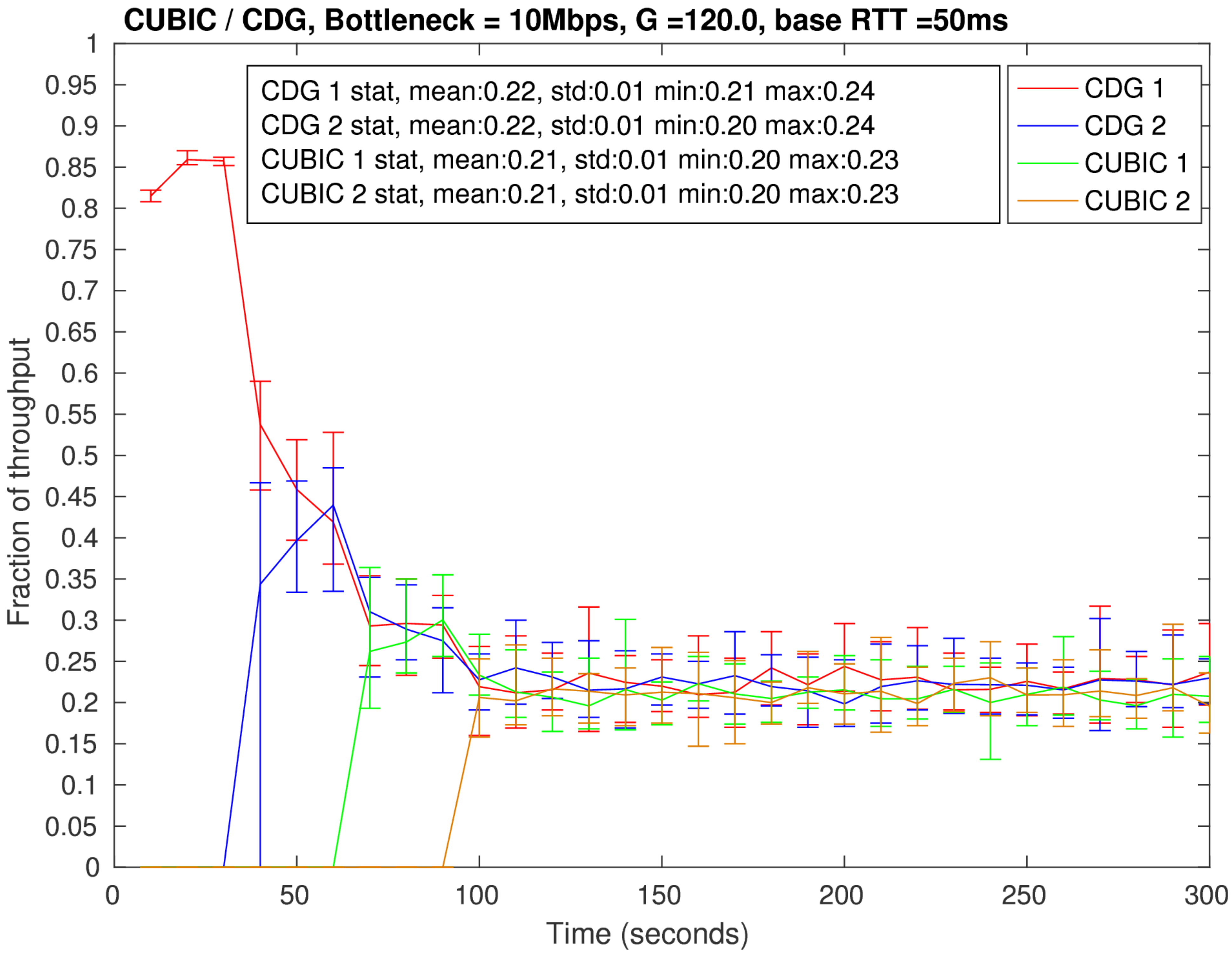




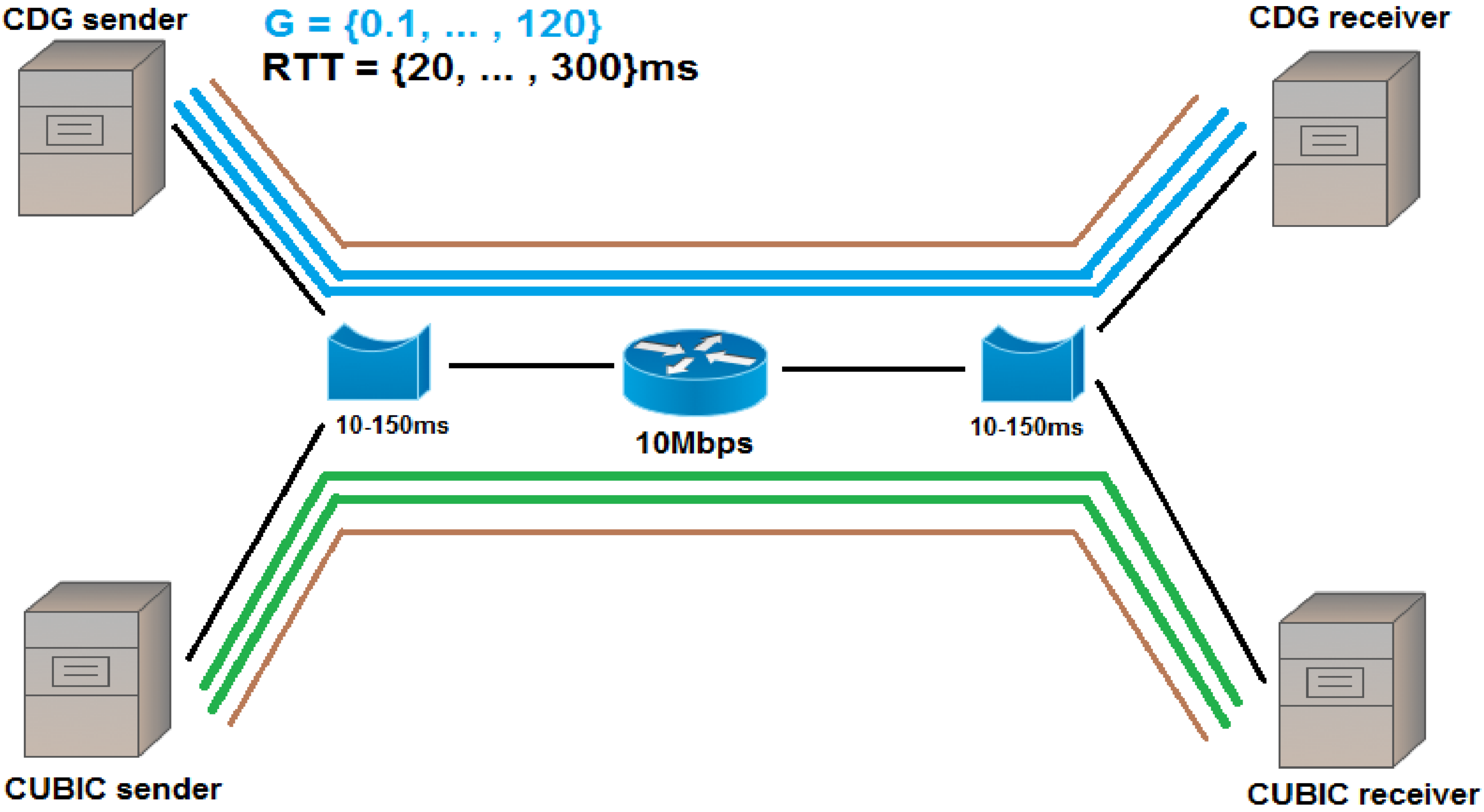
CDG 1 stat, mean:0.05, std:0.00 min:0.05 max:0.06
CDG 2 stat, mean:0.06, std:0.01 min:0.05 max:0.11
CUBIC 1 stat, mean:0.39, std:0.03 min:0.35 max:0.52
CUBIC 2 stat, mean:0.37, std:0.03 min:0.26 max:0.41

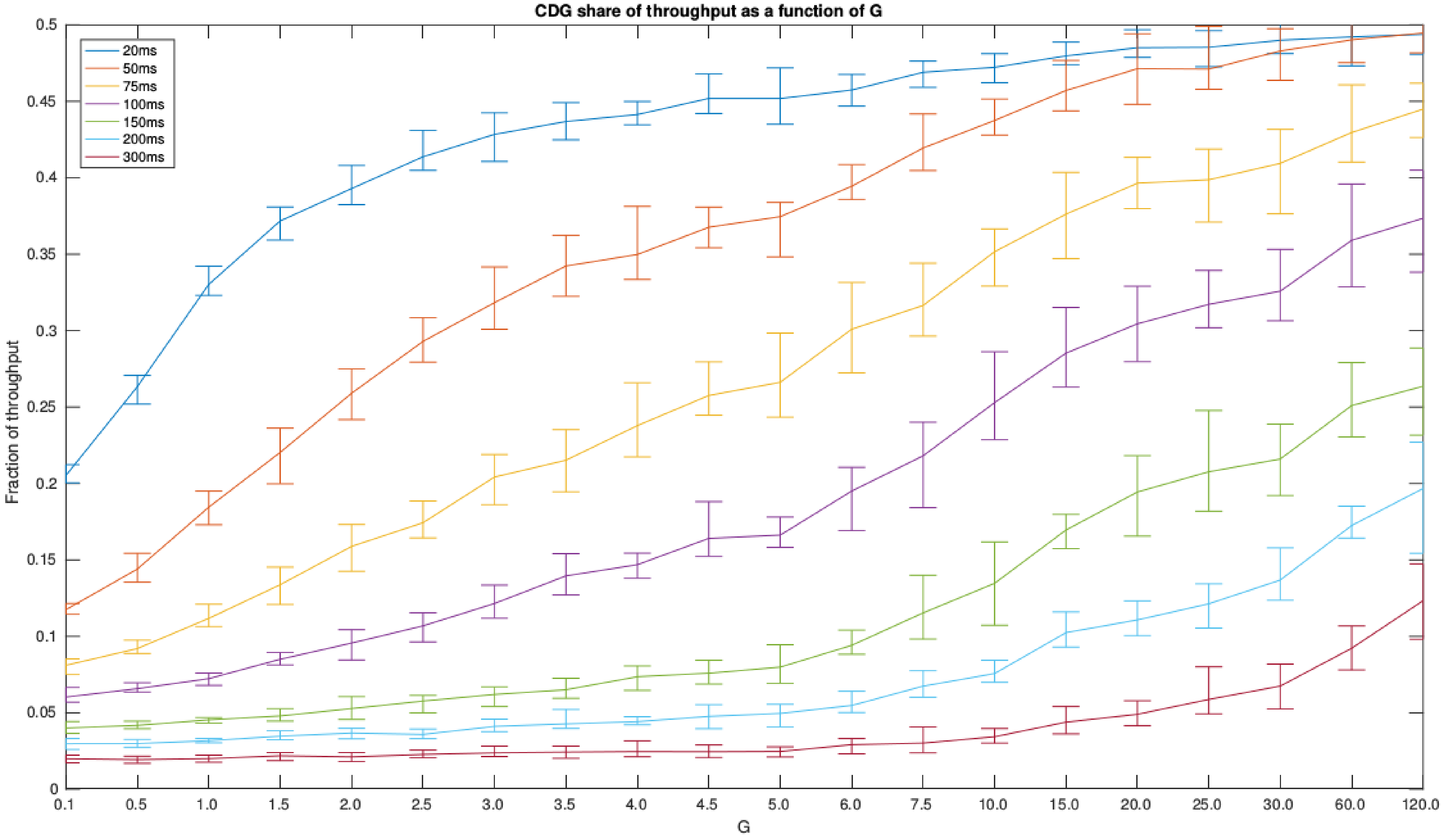

CUBIC / CDG, Bottleneck = 10Mbps, G =2.0, base RTT =50ms
CDG 1 stat, mean:0.12, std:0.01 min:0.10 max:0.14
CDG 2 stat, mean:0.12, std:0.04 min:0.10 max:0.30
CUBIC 1 stat, mean:0.32, std:0.02 min:0.30 max:0.38
CUBIC 2 stat, mean:0.31, std:0.02 min:0.22 max:0.33
CDG 1
CDG 2
CUBIC 1
CUBIC 2
Fraction of throughput
1
0.95
0.9
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
50
100
150
200
250
300
Time (seconds)


CUBIC / CDG, Bottleneck = 10Mbps, G =6.0, base RTT =50ms
CDG 1 stat, mean:0.18, std:0.01 min:0.16 max:0.20
CDG 2 stat, mean:0.19, std:0.02 min:0.16 max:0.26
CUBIC 1 stat, mean:0.26, std:0.01 min:0.24 max:0.28
CUBIC 2 stat, mean:0.25, std:0.01 min:0.23 max:0.27
CDG 1
CDG 2
CUBIC 1
CUBIC 2
Fraction of throughput
1
0.95
0.9
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
50
100
150
200
250
300
Time (seconds)


CUBIC / CDG, Bottleneck = 10Mbps, G =120.0, base RTT =50ms
CDG 1 stat, mean:0.22, std:0.01 min:0.21 max:0.24
CDG 2 stat, mean:0.22, std:0.01 min:0.20 max:0.24
CUBIC 1 stat, mean:0.21, std:0.01 min:0.20 max:0.23
CUBIC 2 stat, mean:0.21, std:0.01 min:0.20 max:0.23
CDG 1
CDG 2
CUBIC 1
CUBIC 2
Fraction of throughput
1
0.95
0.9
0.85
0.8
0.75
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0
50
100
150
200
250
300
Time (seconds)

Experimental set-up




CDG share of throughput as a function of G
Fraction of throughput
G
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5
0.1
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
6.0
7.5
10.0
15.0
20.0
25.0
30.0
60.0
120.0
20ms
50ms
75ms
100ms
150ms
200ms
300ms

Conclusion and future work

- CDG is a delay-based CC which could be a candidate DA-LBE because:
  - Has a tunable G scaling parameter
  - Does not suffer from latecomers advantage
  - Is readily available in Linux and FreeBSD
- Experiments indicate CDG can achieve a tuning range of its aggressiveness against CUBIC between 20-100 % of a what CUBIC flow under similar conditions would achieve when RTT is 20ms.
- Experiments are limited in scope, we only compare against CUBIC (e.g. comparison with Reno could yield different results) and we assume FIFO drop tail queues. Future work may investigate CDG in more heterogeneous environments.
- Future work could investigate whether enabled CDG experimental parameter *alpha_inc*, allowing cwnd to increase by more than 1MSS pr RTT, could widen CDG’s tuning range.
- LBE-mode CDG could be included in NEAT as part of the DA-LBE framework

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 644334 (NEAT). The views expressed are solely those of the authors.