`ndncatchunks` **Performance Issues**
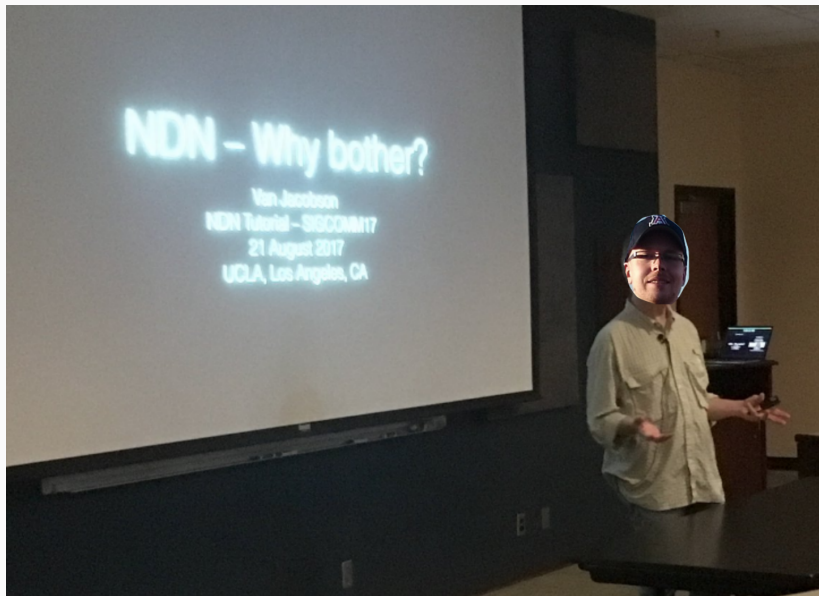
8th NDN Hackathon

Klaus Schneider, Saurab Dulal

March 10, 2019

Why bother about `catchunks/putchunks`?

1. Catchunks = First application new people use

Why bother about `catchunks`/`putchunks`?

1. Catchunks = First application new people use
2. Many larger applications built on ndnchunks

Why bother about `catchunks`/`putchunks`?

1. Catchunks = First application new people use
2. Many larger applications built on ndnchunks
3. Improvements trickle down into SegmentFetcher

Some Reported Performance Issues

1. Performance. on lossy, high-delay WiFi links

Some Reported Performance Issues

1. Performance. on lossy, high-delay WiFi links
2. Window decrease > retransmissions?

Some Reported Performance Issues

1. Performance. on lossy, high-delay WiFi links
2. Window decrease $>$ retransmissions?
3. Catchunks exceeds maximum retries

Some Reported Performance Issues

1. Performance. on lossy, high-delay WiFi links
2. Window decrease > retransmissions?
3. Catchunks exceeds maximum retries
4. Too many/too little cong. marks? (Unix, UDP)

Experiment Setup

Host — Router (VM) — Server (VM)

Experiment Setup

Host — Router (VM) — Server (VM)

- VMs: Virtualbox $+$ mini-ndn
- Traffic shaping (tc netem) at router!
- Catchunks vs. Iperf3 (TCP)

Experiment Setup

Host — Router (VM) — Server (VM)

- VMs: Virtualbox + mini-ndn
- Traffic shaping (tc netem) at router!
- Catchunks vs. Iperf3 (TCP)

Variables:

1. Bandwidth
2. Delay
3. Jitter
4. Buffer Queue Size (qdisc)
5. Link Loss

Not the Problem: Delays < 150ms

BW=50Mbit, queueSize=300

RTT	Catchunks (Mbps)	Iperf (Mbps)
2ms	46.2	48.4
50ms	45.3	47.4
100ms	30.2	32.4

Not the Problem: Jitter

BW=50Mbit, queueSize=300

RTT	Jitter	Catchunks (Mbps)	Iperf (Mbps)
10ms	1ms	45.2	48.1
20ms	2ms	43.3	45.4
100ms	20ms	24.7	37.3

$\Rightarrow$ Some difference, but not very large! (x1.5)

Not the Problem: Packet Loss

BW=50Mbit, queueSize=300, delay=20ms

Loss	Catchunks (Mbps)	Iperf (Mbps)
.1%	38.4	38.3
1.0%	11.8	10.1
5.0%	3.5	3.5

Not the Problem: Delay + Jitter + Loss

BW=50Mbps, qSize=300, delay=60ms, jitter=20ms, loss=1%

Not the Problem: Delay + Jitter + Loss

BW=50Mbps, qSize=300, delay=60ms, jitter=20ms, loss=1%

Catchunks: 3.94 Mbps
Iperf3: 7.50 Mbps

$\Rightarrow$ Higher difference, but still not very large! (x1.9)

The Problem: NFD Performance (1)

No Traffic Shaping

```
Time elapsed: 9620.52 milliseconds
Total size: 104858kB, 23832 segments
Goodput: 87.194912 Mbit/s
Total # of lost/retx segments: 829 (caused 40 window decr)
Packet loss rate: 3.36158%, cong marks: 10
RTT min/avg/max = 0.833/16.764/125.612 ms
```

The Problem: NFD Performance (1)

No Traffic Shaping

```
Time elapsed: 9620.52 milliseconds
Total size: 104858kB, 23832 segments
Goodput: 87.194912 Mbit/s
Total # of lost/retx segments: 829 (caused 40 window decr)
Packet loss rate: 3.36158%, cong marks: 10
RTT min/avg/max = 0.833/16.764/125.612 ms

IPERF:
[ ID] Interval      Transfer     Bandwidth  Retr  Cwnd
[  4] 0.00-0.55s   100 MBytes   1.53 Gbps   92    348 KBytes
```

The Problem: NFD Performance (1)

No Traffic Shaping

```
Time elapsed: 9620.52 milliseconds
Total size: 104858kB, 23832 segments
Goodput: 87.194912 Mbit/s
Total # of lost/retx segments: 829 (caused 40 window decr)
Packet loss rate: 3.36158%, cong marks: 10
RTT min/avg/max = 0.833/16.764/125.612 ms

IPERF:
[ ID] Interval     Transfer    Bandwidth   Retr  Cwnd
[ 4] 0.00-0.55s   100 MBytes   1.53 Gbps   92    348 KBytes
```

CPU is the limiting factor: Router: 96%, Server: 80%
$\Rightarrow$ **NFD,** buffer size, cong. marks, window adaptation?

The Problem: Buffer Queue Size (2)

BW=50Mbit, delay=20ms

Q (Pkts)	Catchunks (Mbps)	Iperf (Mbps)
20	5.7	31.0
50	15.0	46.3
100	37.5	47.8
300	46.8	48.0
1000	47.0	48.2

Large difference: **5.4x lower throughput!**

The Problem: Buffer Queue Size (2)

BW=50Mbit, delay=20ms

Q (Pkts)	Catchunks (Mbps)	Iperf (Mbps)
20	5.7	31.0
50	15.0	46.3
100	37.5	47.8
300	46.8	48.0
1000	47.0	48.2

Large difference: **5.4x lower throughput!**

Improves slightly with smaller chunk size (1.3KB)

5.7 Mbps $\Rightarrow$ 7.5 Mbps. **???**

The Problem: Delay > 200ms (3)

50MB file, BW=50Mbit, queueSize=1000

RTT	Catchunks (Mbps)	Iperf (Mbps)
100ms	11.8	44.1
150ms	12.4	44.6
200ms	1.4	32.4
300ms	0.9	22.5
400ms	2.2	16.6

Large difference: **25x lower throughput!**

The Problem: Delay > 200ms (3)

50MB file, BW=50Mbit, queueSize=1000

RTT	Catchunks (Mbps)	Iperf (Mbps)
100ms	11.8	44.1
150ms	12.4	44.6
200ms	1.4	32.4
300ms	0.9	22.5
400ms	2.2	16.6

Large difference: **25x lower throughput!**

What's special about 200ms? ⇒ **minRTO=200ms!**

Hackathon Improvements: Better Statistics (1)

Measure spurious retransmissions!

```
All segments have been received.
Time elapsed: 78861.7 milliseconds
Total # of segments received: 11916
Total size: 52428.8kB
Goodput: 5.318554 Mbit/s

RTO Timeouts: 245 (caused 22 window decreases)
Retx segments: 49, skipped: 196

Packet loss rate: 0.409528%
Total # of received congestion marks: 1
RTT min/avg/max = 201.598/207.656/261.004 ms
```
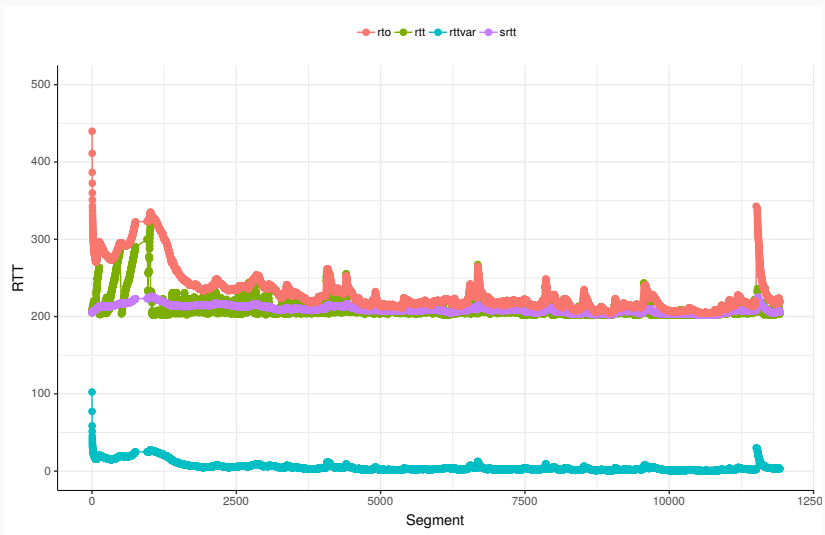
Hackathon Improvements: Better Statistics (1)

Measure spurious retransmissions!

```
All segments have been received.
Time elapsed: 78861.7 milliseconds
Total # of segments received: 11916
Total size: 52428.8kB
Goodput: 5.318554 Mbit/s

RTO Timeouts: 245 (caused 22 window decreases)
Retx segments: 49, skipped: 196

Packet loss rate: 0.409528%
Total # of received congestion marks: 1
RTT min/avg/max = 201.598/207.656/261.004 ms
```
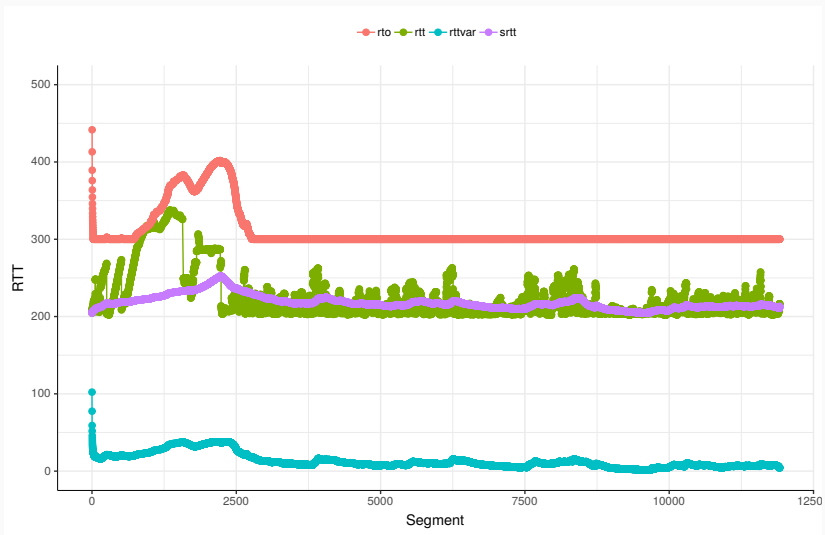
Explains why sometimes **window decrease** > **retx!**

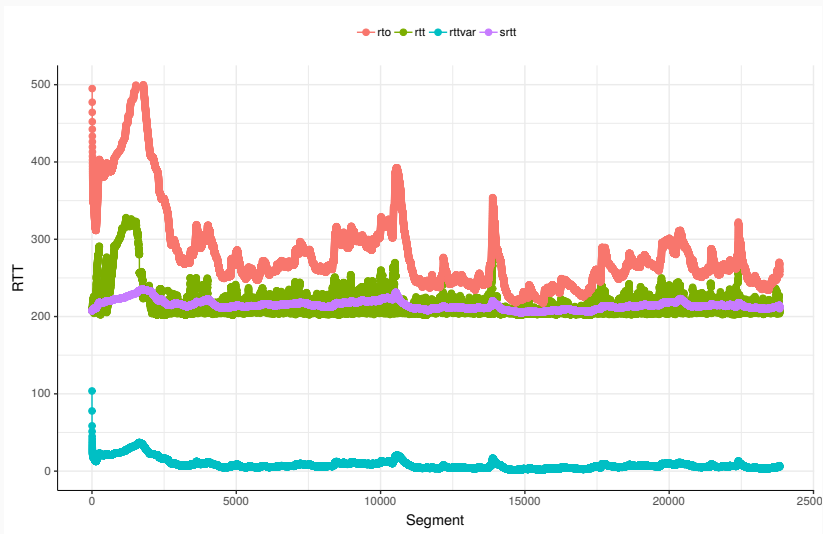# Hackathon Improvements: Increase RTO (2)

$$RTO = sRTT + k * varRTT$$

Increase minRTO to 300ms

Hackathon Improvements: Increase RTO (2)

Increase **k=8.** TP: 5.6 Mbps $\Rightarrow$ **28.6 Mbps** (TCP: 34Mbps)

Delay=400ms, 100MB file

Scen	TP (Mbps)	cwnd dec.	spur. rtx
AIMD, k=4	2.3	61	308
CUBIC, k=4	8.8	27	260
CUBIC, k=6	12.7	7	123
CUBIC, k=8	14.6	7	5
TCP	16.1	-	-

Small Improvements & Open Problems

1. Increase retry limit from 3 to **15!**

Small Improvements & Open Problems

1. Increase retry limit from 3 to **15!**
2. Look into **small buffer size** issue!
    - Timeouts: > 1000 in NDN vs 80 in TCP
    - Increasing k & CUBIC doesn't help

Small Improvements & Open Problems

1. Increase retry limit from 3 to **15!**
2. Look into **small buffer size** issue!
   - Timeouts: > 1000 in NDN vs 80 in TCP
   - Increasing k & CUBIC doesn't help
3. Tune congestion marks (UDP $+$ Unix sockets)

Small Improvements & Open Problems

1. Increase retry limit from 3 to **15!**
2. Look into **small buffer size** issue!
   - Timeouts: > 1000 in NDN vs 80 in TCP
   - Increasing k & CUBIC doesn't help
3. Tune congestion marks (UDP + Unix sockets)
4. Test with Mini-NDN WiFi

Any Questions?

Klaus Schneider, Saurab Dulal