An Improvement of ECN to Enhance TCP Fairness Performance

draft-sun-tcpm-ecn-improvement-00

Marcus Sun

marcus.sun@huawei.com

Jing Cui (presenter)

IETF97-Seoul

Background

❑ RFC 3168 allows E2E notification of network congestion:

- CE can be marked by any of the network devices in the path
- TCP receiver echoes CE to sender by ACK
- Sender adjusts SWND according to CWND & ECN labeled packets ratio

❑ CANNOT accurately reflect network congestion status:

- If multiple nodes exceed threshold value ⟹

Sender received congestion status ≥ Actual link congestion(the worst node status)

Other ECN Limitations – 1/2

- ❑ Lack of good solution for light load in network:
  - Current ECN only handles with network congestion
  - If light loaded, lack of rapid notification to TCP sender
  - Unable to rapidly adjust SWND for better utilization of idled bandwidth

Other ECN Limitations - 2/2

❑ TCP fairness problem:

- In Current ECN, different flows on a same path may attain discrepant link congestion status:
  - Packet transmission order depends on flows' own sending rate
  - Amounts of packets sent in 1 RTT vary ➔Proportions of ECN labelled packets vary

- Unfairness between flows' SWND adjustment ratio:
  - Especially for flows in same business scenarios/protocols

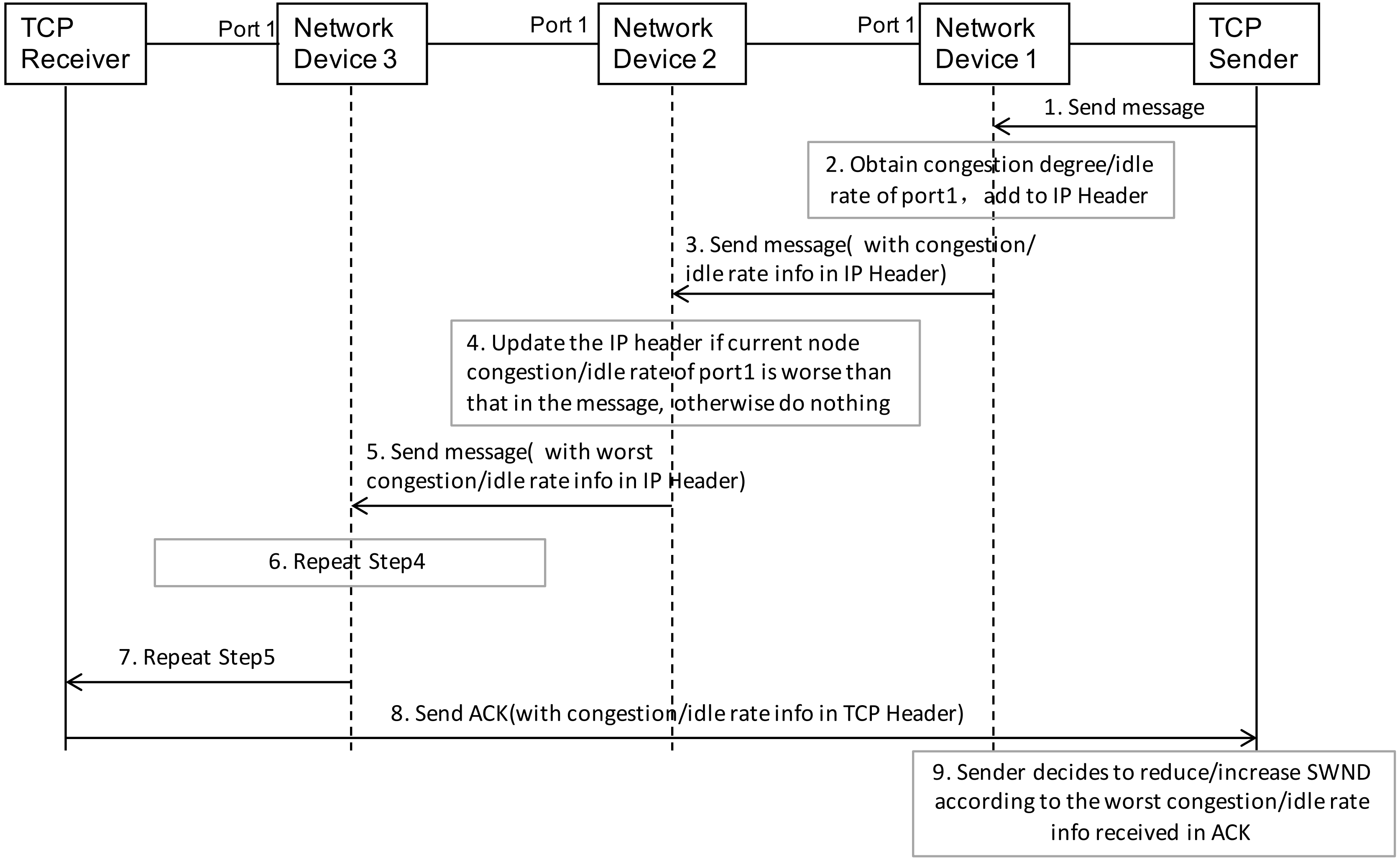
Example:
Assuming that 3 receivers are requesting the same sports live show stream in 4K resolution,
If the 3 flows attain different link congestion status, flow rates will vary although they are in the same contents/protocols.

Main Goal

- ❑ Optimize the ECN scheme, which can:
  - Reflect the worst congested node status for more accurate congestion control
  - Fully utilize the link idle rate in light load network situation
  - Achieve more fairness for different streams

Improvement of ECN

The worst case(congestion or idle rate) of network devices is notified to TCP sender



1. Send message
2. Obtain congestion degree/idle rate of port1， add to IP Header
3. Send message(with congestion/ idle rate info in IP Header)
4. Update the IP header if current node congestion/idle rate of port1 is worse than that in the message, otherwise do nothing
5. Send message(with worst congestion/idle rate info in IP Header)
6. Repeat Step4
7. Repeat Step5
8. Send ACK(with congestion/idle rate info in TCP Header)
9. Sender decides to reduce/increase SWND according to the worst congestion/idle rate info received in ACK

❑ Congestion degree:

- Time it takes to complete message transmission in the link cache
- Example:
  - Cache size: 20MB, Link Bandwidth: 1Gbps
  - 20 x 8 Mb/1Gbps = 160/1024 = 0.16s

❑ Link idle rate:

- Opposite to link usage
- Example:
  - 1Gbps link with 600Mbps traffic
  - Link usage 60%, Link idle rate 40%

Send Window Adjusting Method

❑Option 1: Using the Worst Congestion Degree to Adjust SWND

- TCP sender adjusts the decrease rate of the window according to the worst congestion degree in the received TCP ACK and the current SWND

❑ Option 2: Using the Worst Idle Rate to Adjust SWND

- TCP sender adjusts the increase in window size based on the worst idle rate in the received TCP ACK and the current SWND

Example: Assuming that,

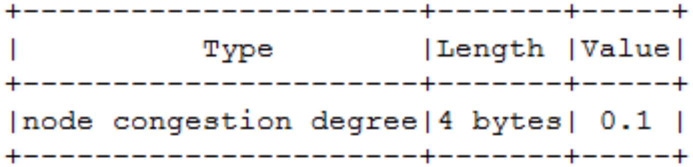
The worst idle rate in TCP ACK: 40%
The current window of Flow1: 1000
The current window of Flow2: 200
The current link utilization (total flow rate): 1-40% = 60%
The **flow rate can be increased by** 40% / 60% = 66.67%

For Flow 1: the window should be increased by 1000 x 66.67% = 667
For Flow 2: the window should be increased by 200 x 66.67% = 133

TCP/IP Option Extension

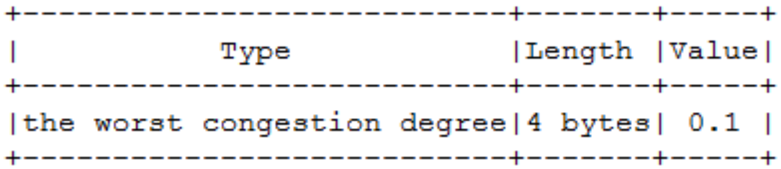
❑Congestion Degree Extend

- Congestion degree carried in IP packets can be achieved by extending the IP option



```
+----------------------+-------+-----+
|         Type         |Length |Value|
+----------------------+-------+-----+
|node congestion degree|4 bytes| 0.1 |
+----------------------+-------+-----+
```

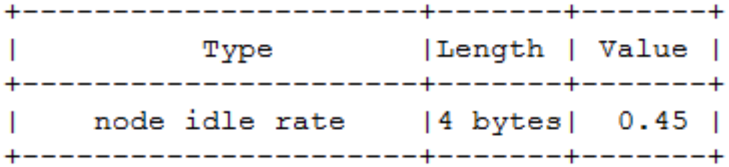
- The worst congestion degree carried by TCP ACK can be extended by TCP option



```
+---------------------------+-------+-----+
|           Type            |Length |Value|
+---------------------------+-------+-----+
|the worst congestion degree|4 bytes| 0.1 |
+---------------------------+-------+-----+
```

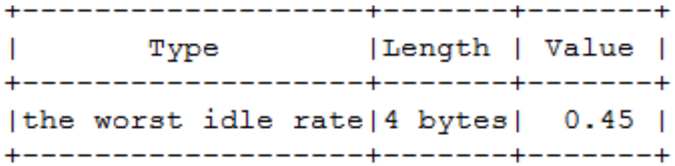
❑Idle Rate IP Extend

- Idle rate carried in IP packets can be achieved by extending the IP option



```
+----------------------+-------+-------+
|         Type         |Length | Value |
+----------------------+-------+-------+
|    node idle rate    |4 bytes|  0.45 |
+----------------------+-------+-------+
```

- The worst idle rate carried by TCP ACK can be extended by TCP option



```
+-------------------+-------+-------+
|       Type        |Length | Value |
+-------------------+-------+-------+
|the worst idle rate|4 bytes|  0.45 |
+-------------------+-------+-------+
```

TCP Fairness Enhancement

❑ Each message is carrying the SAME worst node/port status (congestion/idle rate) in the same path

❑ TCP sender is no longer calculating the ratio of ECN labelled messages to modify the SWND, but adjusting window size according to the accurate worst node/port status

❑ Since each TCP flow attains the same worst node/port status, TCP sender give them same proportion on SWND adjustment, which indicates that their TCP fairness is guaranteed.

Next Step

❑ Read and comment, please – thx for useful comments so far

❑ More experiments & verification:

- Give some more verification supports in next version
- Methods of more rapid notification to sender
- Alternative method to present the worst link node status

❑ Etc.