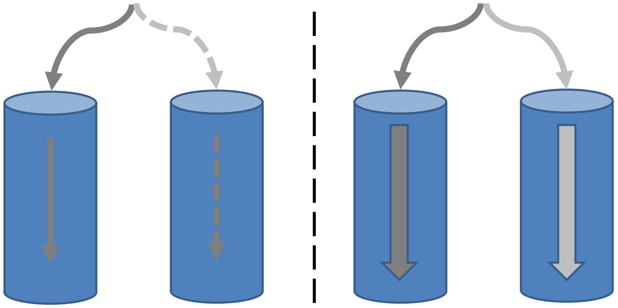
Multi-path TCP in Data Centers

Trade-off between reliability and utilization

Static Data Centers



10G

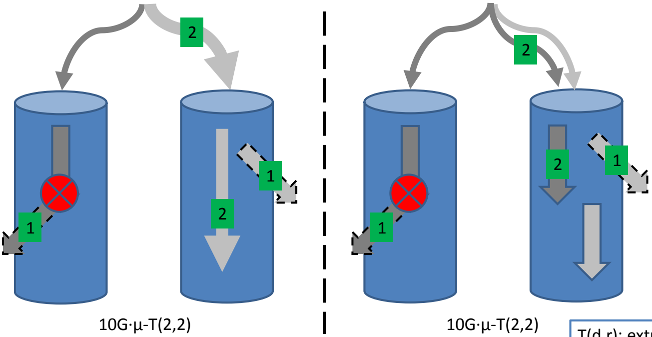
Single-path:
- **low utilization**
- high reliability

20G·μ

Multi-path:
- high utilization
- **low reliability**

μ: efficient throughput considering ACK, coding.

Static Data Centers



Multi-path failures:
1. Drop two subflows;
2. Retransmit initial flow.

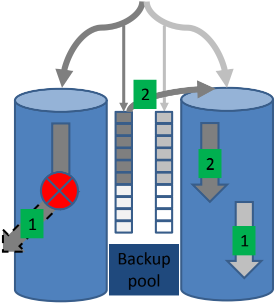
Multi-path failures:
1. Drop two subflows;
2. Retransmit two subflows;
3. Decoding after transmit.

T(d,r): extra overhead of drops and retransmit.

Multi-path TCP in Data Centers

High reliability through flows backup

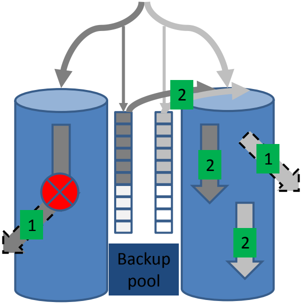
Static Data Centers



10G·µ-T(1,1)

Multi-path failures by backup:

1. Drop one subflow;
2. Retransmit one subflow;
3. Decoding after transmit.



10G·µ-T(2,2)

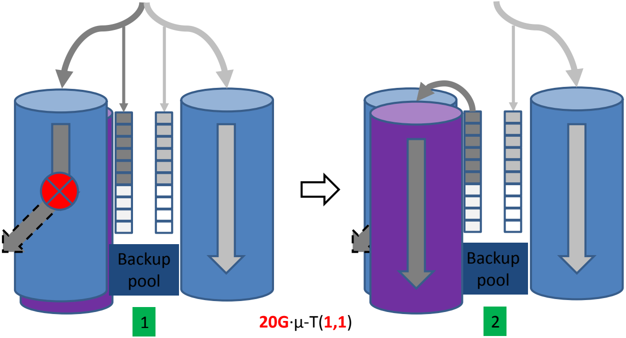
Multi-path failures by backup:

1. Drop two subflows;
2. Retransmit two subflows;
3. Decoding **as transmitting**.

Multi-path TCP in Data Centers

High efficiency through flexible switching

Flexible Data Centers



20G·μ-T(**1,1**)

Multi-path failures by backup:

1. Drop **one subflow**;
2. Retransmit **one subflow**;
3. Decoding **as transmitting**.

1. Split flow(s) into subflows
2. Congestion control by block ACK
3. Add premix to subflows
4. Transmit subflows
5. Decoding according to premix and delay (failures occur)

Failures handling technologies

- **Flow backup** reduces the overhead of drops and retransmission: T(2,2) $\rightarrow$ T(1,1).
- **Flexible switching** improves the capacity and reduces the delay: 10G $\rightarrow$ 20G, decoding after transmit $\rightarrow$ as transmitting.

Multi-path TCP technologies

- **Block ACK** reduces extra overhead: ACK/subflow $\rightarrow$ ACK/block
-

These are much work on multi-path TCP including protocol design and congestion control. But the multi-path routing and scheduling (topology control) in flexible data centers has not been explored, especially on failures handling (including congestion control) and flexible switching.