# Application-aware Data Center Network (APDN) Use Cases and Requirements

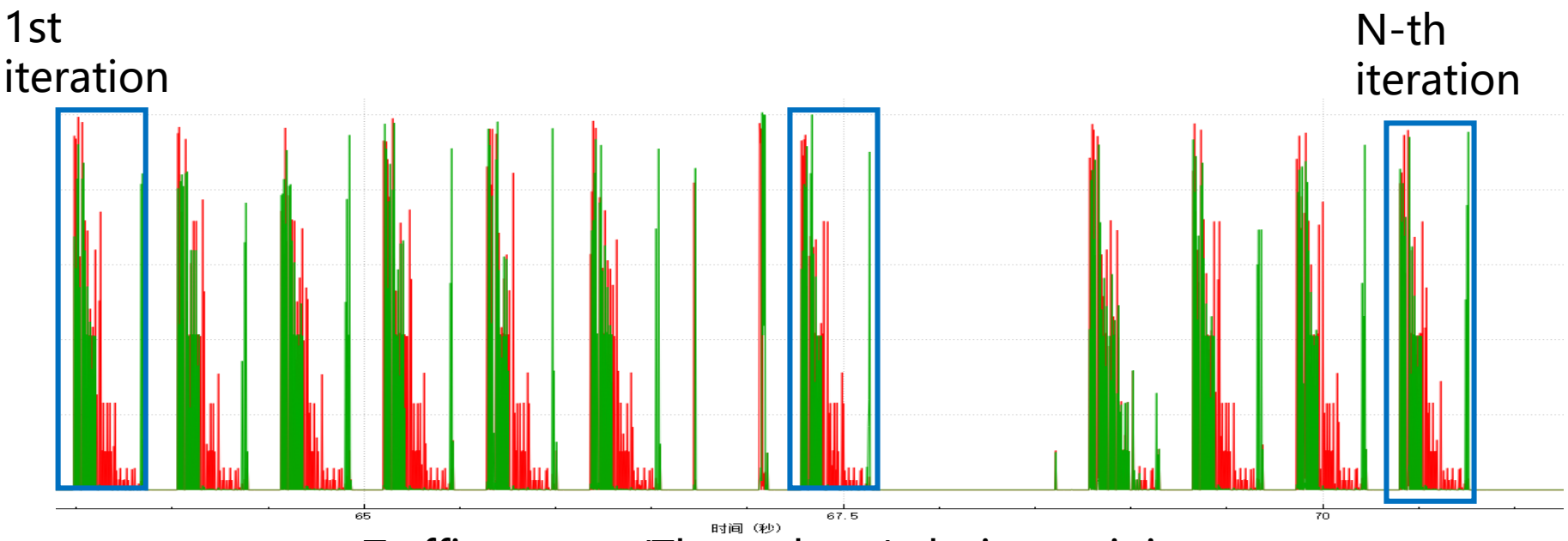
https://datatracker.ietf.org/doc/draft-wh-rtgwg-application-aware-dc-network/

Hiabo Wang, Hongyi Huang (Huawei)

RTGWG, 118

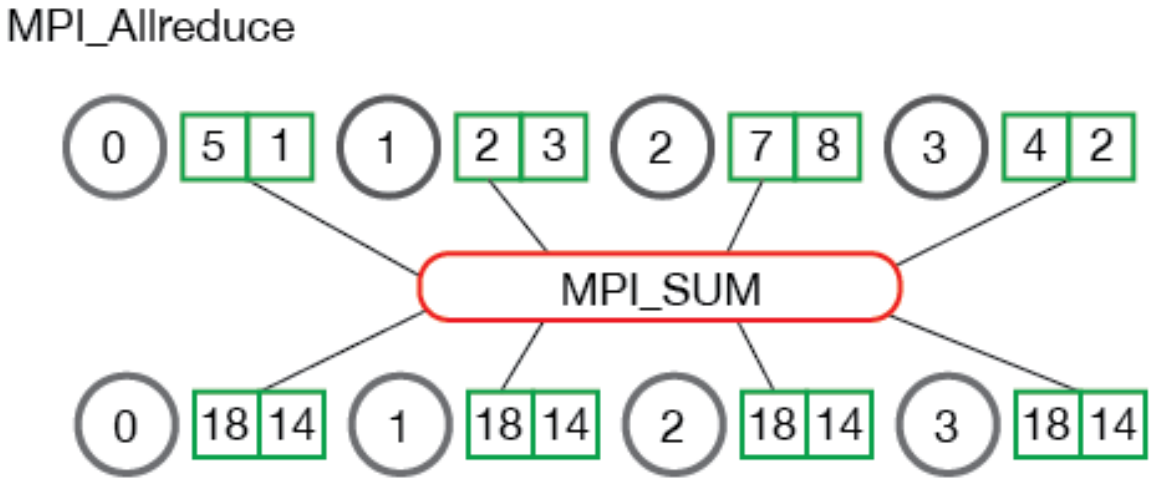
# Distributed AI Model Training

- Large AI models(e.g., Large Language Models) require distributed training among thousands of accelerators(e.g., GPUs), which generates **synchronized, periodic and large flows** to exchange intermediate results(i.e., update gradients) between accelerators before a next iteration.




Traffic pattern(Throughput) during training

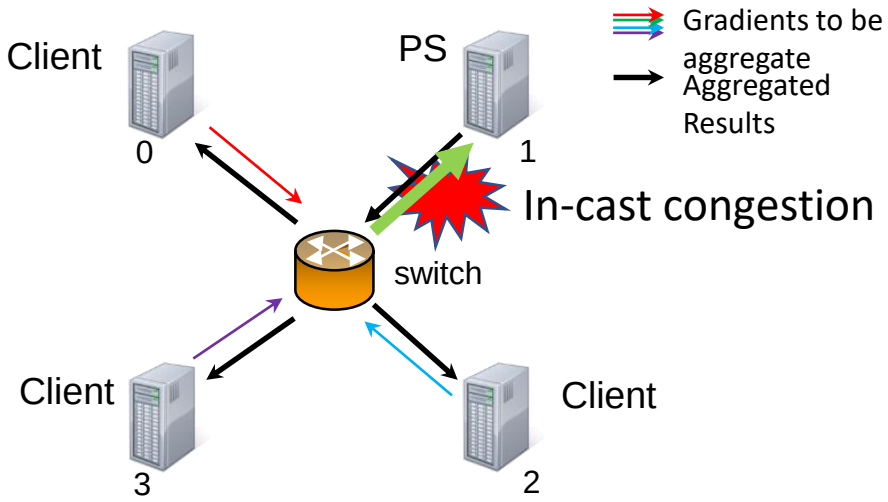
# UC1: In-network computing for distributed machine learning training




- As one kind of MPI(Message Passing Interface), AllReduce is widely used in each iteration during the process of distributed AI model training.
- AllReduce will reduce the values from multi-processors (MPI_SUM) and distribute the results to all processors finally.

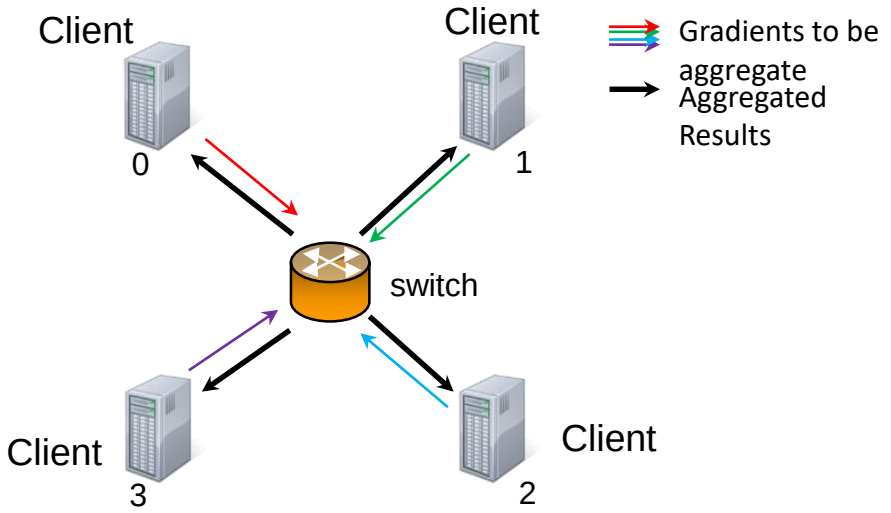
# UC1: In-network computing for distributed machine learning training

**AS-IS**




Under the **Parameter Server(PS)** architecture, when multiple clients send a large amount of gradient data to the same server simultaneously for reducing, it is prone to induce in-cast (many-to-one) congestion from the perspective of server.

**TO-BE**




**In-network computing (INC)** offloads the behavior of the server (MPI_SUM) to the switch capable of line-rate processing, eliminating the in-cast congestion.

# UC1: In-network computing for distributed machine learning training

- To enable INC, current implementations (e.g., ATP[atp], NetReduce[netreduce]) require the switches
  - to parse upper-layer protocol
  - and understand application-specific logic that is dedicated to certain application

- Gap analysis:
  - Different transports/applications
  - Difficult to fetch INC info with encryption of app content applied

# UC1: In-network computing for distributed machine learning training

- Potential Solution:
  - APN can be explored to transmit information about the requested INC operations and information about the corresponding data segments to perform INC.

- APDN Requirements:

1. APN MUST carry application identifier to distinguish different INC tasks.
2. APN MUST support to carry various formats and length of application data (such as gradients in this use case) to apply INC and the expected INC operations.
3. APN SHOULD be able to carry other application-aware information, not compromising the reliability of end-to-end transport.
4. APN MUST be able to carry complete INC results and record the computation status in the data packets.

# Other Use Cases

- UC2: Fine-grained packet scheduling for load balancing
- UC3: Refined congestion control that requires feedback of accurate congestion information

https://datatracker.ietf.org/doc/draft-wh-rtgwg-application-aware-dc-network/

# Next Steps

- Solicit comments and refine drafts.
- Welcome cooperation.
- Possible side meeting/BOF plan for IETF

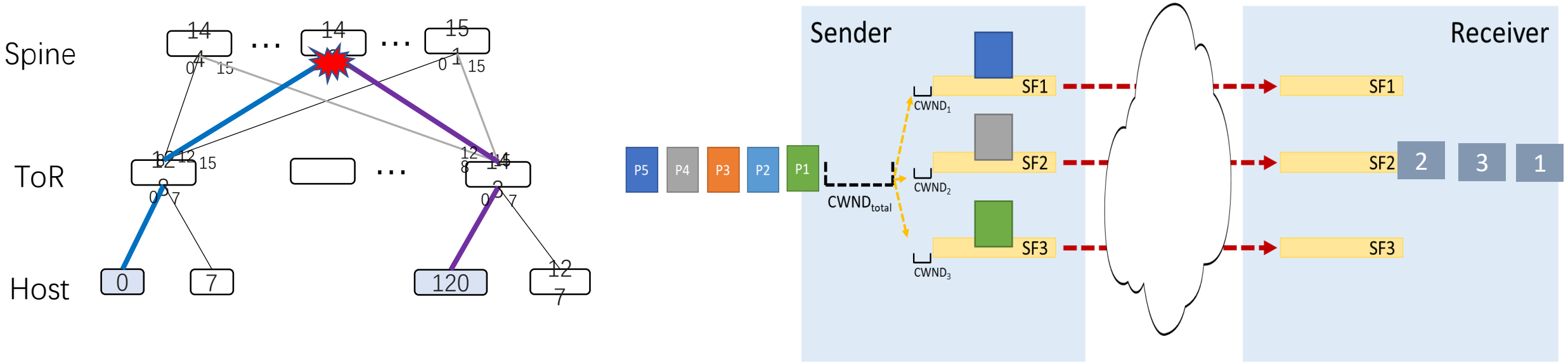
draft-li-rtgwg-apn-app-side-framework

draft-peng-rtgwg-apn-for-media-service

draft-wh-rtgwg-application-aware-dc-network

Thank you

# UC2: Fine-grained packet scheduling for load balancing



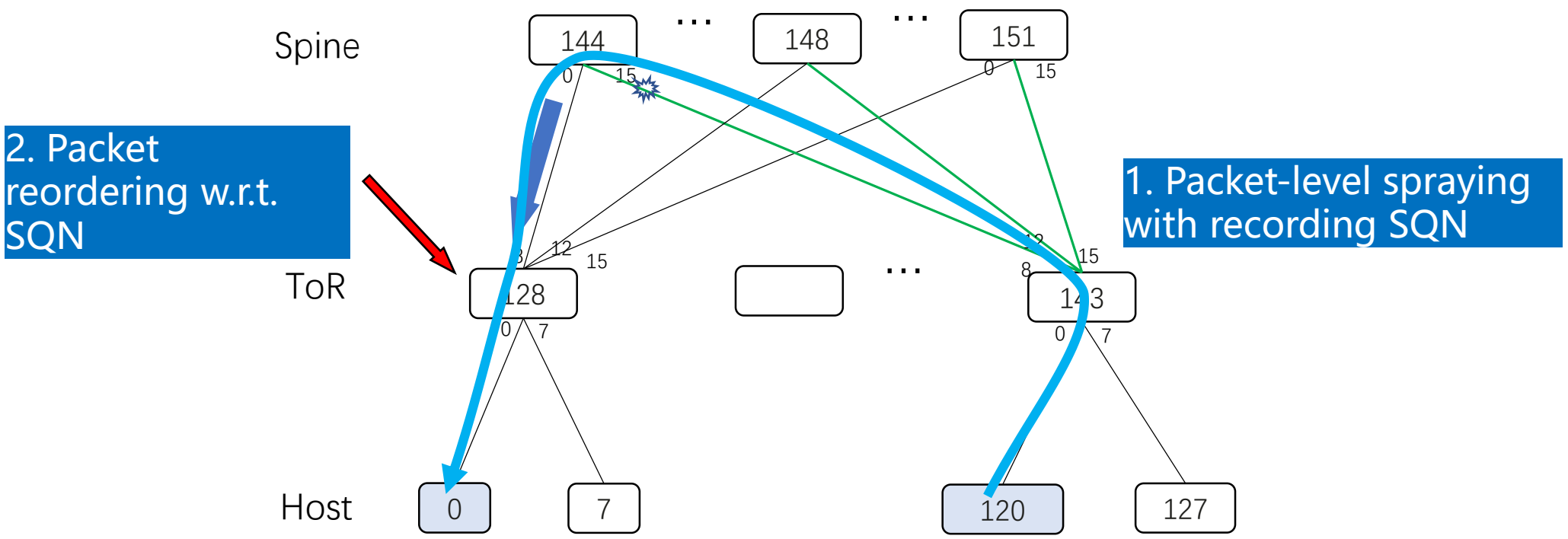

**Traditional per-flow ECMP**: Synchronized large flows may be distributed to the same path/switches and incur congestion.

➔**Fine-grained per-packet ECMP**: leading to packet disorder due to multi-pathing.

➔Experiments[1] show that disorder can greatly decrease the performance of applications.

[1] https://www.linkedin.com/pulse/spray-validation-dmitry-shokarev/

# UC2: Fine-grained packet scheduling for load balancing




- One of the solutions is to re-organize the packets in order in the exit ToR switch, without modifying and affecting the end-hosts.

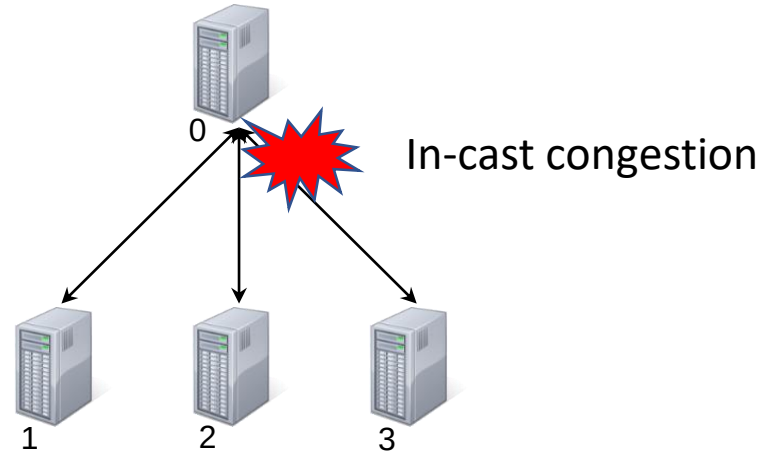
# UC2: Fine-grained packet scheduling for load balancing

- Gap Analysis: Only Transport/application carries sequence number of packets.
  - Difference transports/applications.
  - Each transport/application (flow) requires separate reordering queue.
  - Transport SQN is not supposed to be modified by the network.

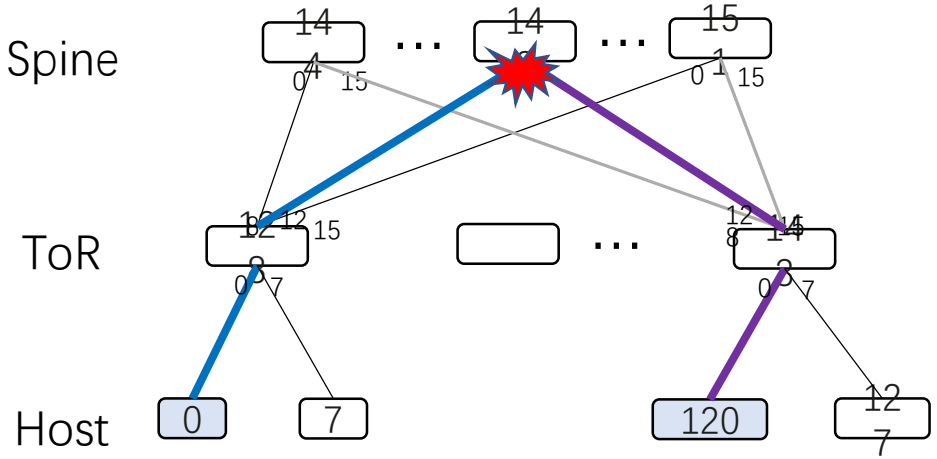
- APDN Requirements:
  1. APN SHOULD encapsulate each packet with SQN besides APN ID for reordering.
  2. The SQN in APN MUST NOT be modified inside the multi-pathing domain and could be cleared from APN at the egress device.
  3. APN SHOULD be able to carry necessary queue information (i.e., the sorting queue ID) usable for fine-grained reordering process.

# UC3: Refined congestion control that requires feedback of accurate congestion information

- AI Data center requires congestion control, due to
  - In-cast congestion (e.g., AllReduce, All2All traffic)

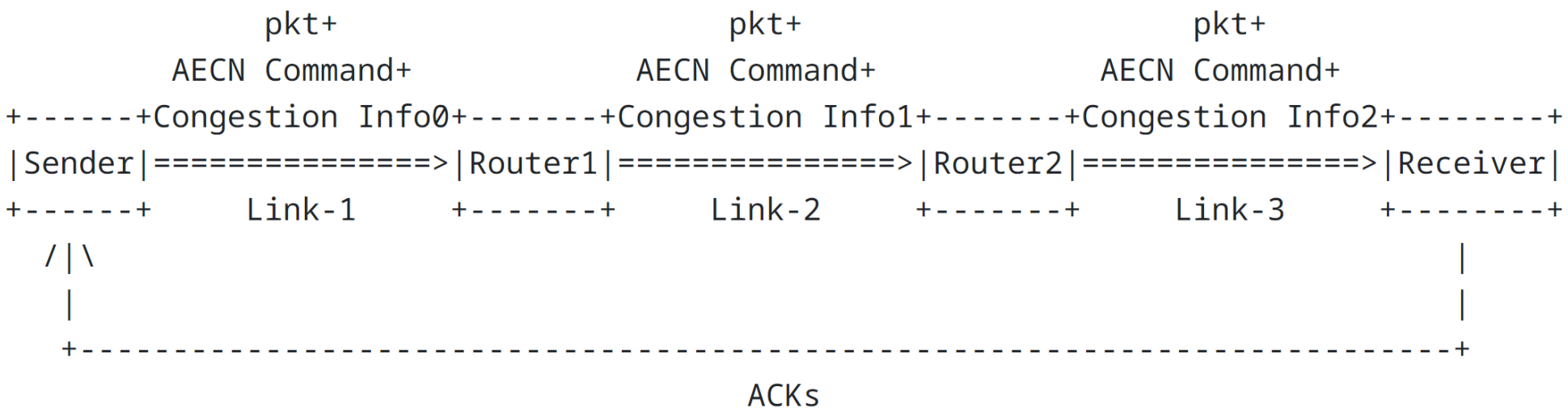



  - Unbalanced load distribution among paths



# UC3: Refined congestion control that requires feedback of accurate congestion information

- ECN [RFC3168] uses only 1-bit marker unable to transmit more congestion information.
- Newly CC algorithms (e.g., AECN[draft-shi-ccwg-advanced-ecn]) encourages to collect/update the congestion info hop by hop to help locate the congestion point and support fine-grained control.



# UC3: Refined congestion control that requires feedback of accurate congestion information

- Potential solution: APN for application side can be used to carry application information to determine the type of information to be collected.

- Requirements:
  1. APN **MUST** allow the data sender to express its intention about which measurement it wants to collect and the condition when it should be recorded or updated.
  2. APN **MUST** allow network nodes to record/update necessary measurement results, if the nodes decide to do so, and transmit these results to the data receiver.