Large-Scale Deterministic Network

draft-qiang-detnet-large-scale-detnet-02

Cristian Qiang qiangli3@huawei.com, Bryan Liu liubingyang@huawei.com
Toerless Eckert tte+ietf@cs.fau.de (Huawei USA)
Liang Geng gengliang@chinamobile.com, Lei Wang wangleiyjy@chinamobile.com

Background

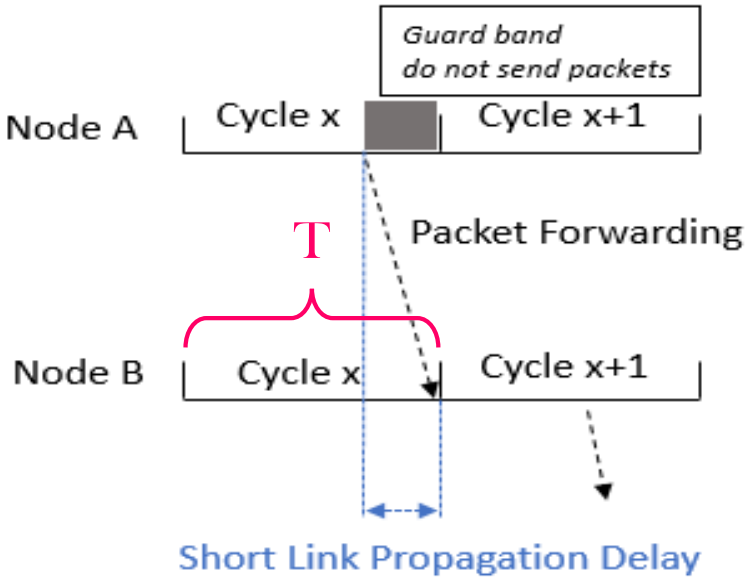
- Charter for DetNet WG: provide bounds on latency, loss, and packet delay variation (jitter), and high reliability
- TSN techniques are equally applicable to DetNet [draft-ietf-detnet-architecture-09]

> While these techniques are currently embedded in Ethernet [IEEE802.3-2018] and bridging standards, we can note that they are all, except perhaps for packet preemption, equally applicable to other media than Ethernet, and to routers as well as bridges. Other media may have its own methods, see, e.g.,

- For Jitter, TSN has two main solutions
  - ✓IEEE 802.1 Qbv (per-flow gate control)
  - ✓IEEE 802.1 Qch (aggregated cyclic queuing and forwarding) ➡ ***More scalable for large-scale Network with a great amount of DetNet flows***

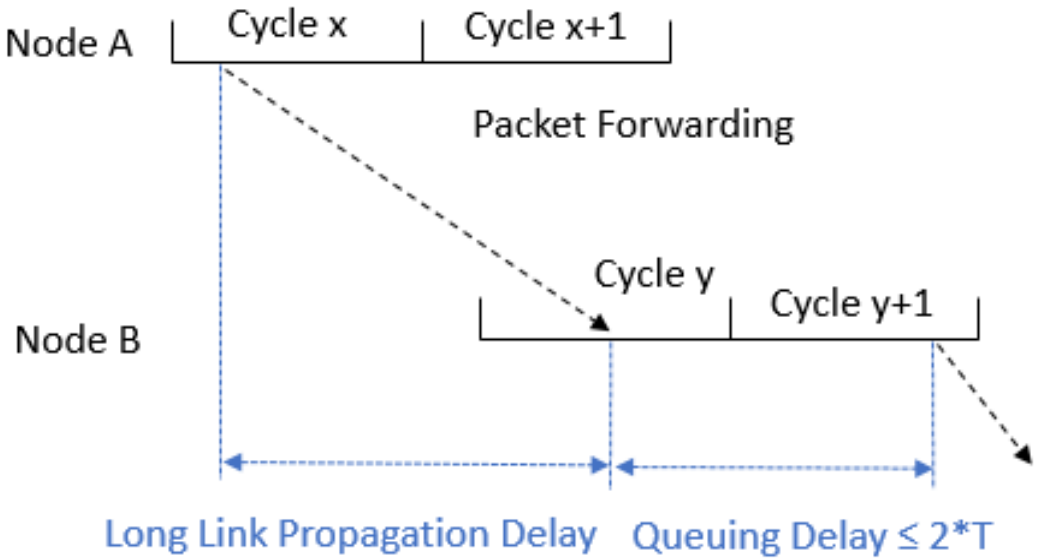
How Qch Works for Long Links

Traditional Qch



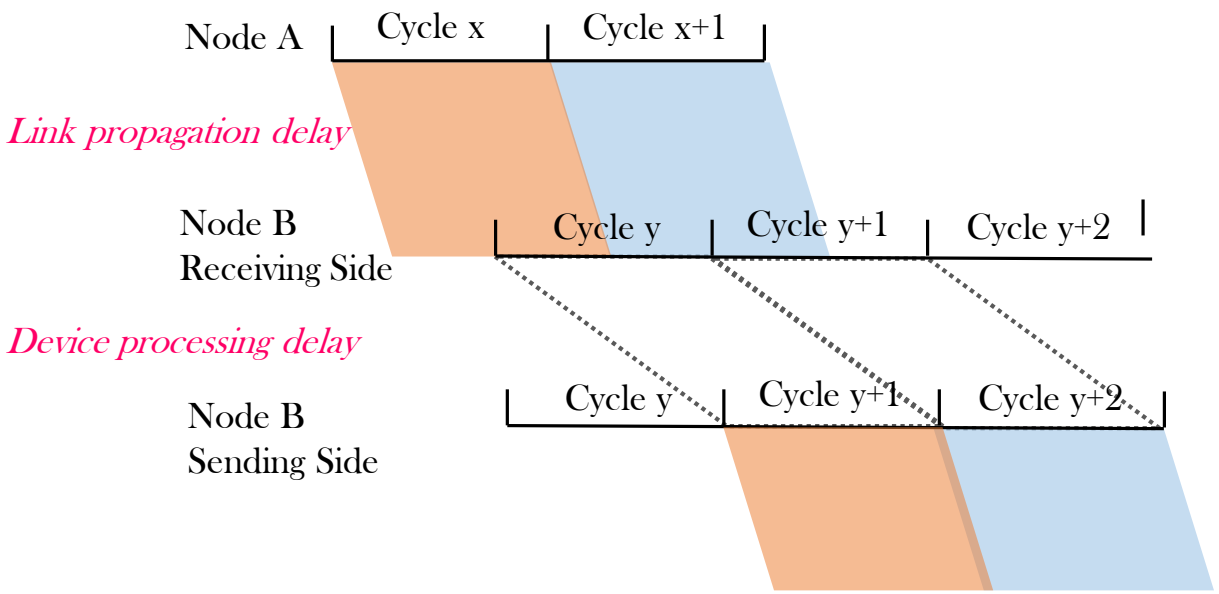
Link propagation delay is required to be smaller than T, T couldn't be too big since End-to-End Jitter's upper bounder is 2*T. Therefore link propagation delay couldn't be very high.

Extension for Long Link



Each pair of neighboring nodes has a stable cycle mapping relationship (e.g., <A, B> has cycle x→ cycle y+1). Packets forwarding follows this cycle mapping relationship.

Cycle Identifiers in Packet Headers are Required



- Cycle y may receive packets sent from two cycles (x and x+1) due to loose time-synchronization, hence needs two receiving queues

Two receiving queues	cycle x → cycle y+1
	cycle x+1 → cycle y+2
One sending queue	cycle x-1 → cycle y

- Three possible behaviors，minimal 2 bits for distinguish
  - ✓ enqueue receiving queue 1
  - ✓ enqueue receiving queue 2
  - ✓ dequque

In order to provide bounded jitter, need to specify each packet's behavior (queue/cycle), that requires to encapsulate extra info into packet

Data Plane Encapsulation Options for Cycle Identifiers

- DetNet forwarding plane documents already indicate that DetNet encap elements (MPLS/IP) can support such operations
- dp-sol-mpls draft: can carry cycle identifiers through T-lable, better to decide and document on preferred header element
- dp-sol-ip draft: IP draft seems missing some parts compared with MPLS draft

draft-ietf-detnet-dp-sol-mpls-01

6. A method of carrying queuing and forwarding indication.

MPLS-SR [I-D.ietf-spring-segment-routing-mpls]). The LSP (T-Label) label and/or the S-Label may be used to indicate the queue processing as well as the forwarding parameters.

o Zero or more MPLS transport LSP label(s) (T-label) used to direct the packet along the label switched path (LSP) to the next peer node along the path. When Penultimate Hop Popping is in use there may be no label T-label in the protocol stack on the final hop.

draft-ietf-detnet-dp-sol-ip-01

6.3. DetNet IP Traffic Treatment Procedures

Implementations if this document MUST ensure that a DetNet flow receives the traffic treatment that is provisioned for it via management and control functions, e.g., via [YANG-REF-TBD]. General information on DetNet service can be found in [I-D.ietf-detnet-flow-information-model]. Typical mechanisms used to provide different treatment to different flows includes the allocation of system resources (such as queues and buffers) and provisioning or related parameters (such as shaping, and policing). Support can also be provided via an underlying network technology such as MPLS [I-D.ietf-detnet-dp-sol-mpls] and IEEE802.1 TSN Section 7. Other than in the TSN case, the specific mechanisms used

MPLS-based Encapsulation (for T-Lable)

- ✓ TLV of SRv6
- ✓ TC of MPLS Labels (S/T) (used to be EXP)
- ✓ TC of MPLS-SR SIDs (used to be EXP)
- ✓ *Three labels/adjacency of SIDs for MPLS-SR*
- ✓ Etc.

IP-based Encapsulation Extension ??

- ✓ Native IP Header
- ✓ IPv6 Extension Header
- ✓ Overlay encapsulation
- ✓ Etc.

Thank you!

Prototype Demo at 2nd Floor Coffee Break Now!