P802.1Qcz interworking with other data center technologies

Jesus Escudero-Sahuquillo, Pedro Javier Garcia, Francisco J. Quiles, Jose Duato

IEEE 802.1 Plenary Meeting, San Diego, CA, USA
July 8, 2018
Executive summary

Existing congestion control mechanisms work end-to-end. We need complementary mechanisms that react quickly when transient congestion appears, also preventing HoL blocking from degrading performance.
Outline

• Why congestion isolation is needed?
• Analysis of congestion scenarios
• Limitations of current technologies
• Congestion Isolation in DCNs
• Conclusions
Outline

• Why congestion isolation is needed?
• Analysis of congestion scenarios
• Limitations of current technologies
• Congestion Isolation in DCNs
• Conclusions
Why congestion isolation is needed?

• Todays Datacenters (DCNs) require a **flexible fabric** for carrying in a convergent way traffic from different types of applications, storage of control.

• Fabric design for DCNs must minimize or eliminate packet loss, provide **high throughput** and maintain low latency.

• These goals are crucial for applications of OLDI, Deep Learning, NVMe over Fabrics and the Cloudified Central Offices.

• However, **congestion** threatens these goals.

Why congestion isolation is needed?

- HoL-blocking dramatically degrades the network performance (e.g. PFC has not enough granularity and there is no congested flow identification).

- Classical e2e congestion control for lossless networks is difficult to tune, reacts slowly, and may introduce oscillations and instability [1].

---

[1] Jesús Escudero-Sahuquillo, Ernst Gunnar Gran, Pedro Javier García, Jose Flich, Tor Skeie, Olav Lysne, Francisco J. Quiles, José Duato: Combining Congested-Flow Isolation and Injection Throttling in HPC Interconnection Networks. ICPP 2011: 662-672
Why congestion isolation is needed?

• We need a congestion isolation (CI) mechanism that **reacts quickly** when transient congestion situations appear, preventing network performance degradation caused by the HoL blocking.

• We want a CI mechanism that **complements other technologies** available in the DCNs, so that CI improves their performance, while the others reduce the CI complexity.
Outline

• Why congestion isolation is needed?
• Analysis of congestion scenarios
• Limitations of current technologies
• Congestion Isolation in DCNs
• Conclusions
Analysis of congestion

The Origin of Congestion

- Some packets simultaneously request the same output port within a switch.
- A packet can be forwarded while the other(s) wait(s), since transference speed is determined by the output link.
Analysis of congestion
The Origin of Congestion

• **Persistent contention** during time.
• Buffers containing blocked packets **fill up** at ingress and egress port, and congestion appears.
Analysis of congestion

The Origin of Congestion

- In lossless networks, congestion propagates quickly due to buffers backpressure. Congestion trees grow up in this way.
Analysis of congestion

The Origin of Congestion

- Different congestion trees dynamics makes more complex the congestion management [Garcia et al. 05].

Analysis of congestion

Congestion trees dynamics

• In general, the switch where congestion originates could be located at some initial or intermediate stage or be directly connected to end nodes.
Analysis of congestion

In-Network Congestion

• It usually occurs when congestion is light (i.e. it exceeds available link bandwidth by a small integer factor at most).

• There are two basic scenarios:
  1. A few nodes injecting traffic at full rate towards the same destination.
  2. Many nodes injecting traffic at low rates towards the same destination.

• Egress ports of in-network congested switches work at full capacity and may contend with other flows for upstream switches, moving the root of congestion upwards.
Analysis of congestion

In-Network Congestion

• **Traditional approaches**: spread traffic flows across the multiple paths in order to balance the load and hopefully avoid congestion (load balancing).

• **Problems:**
  1. Spreading traffic do not take into account whether the selected path is congested, generating collisions of traffic flows in paths already congested.
  2. The nature of flows matters: elephant flows increase the chance of creating in-network congestion.
  3. Traditional load balancing (e.g. ECMP) do not work where incast congestion appear.
Analysis of congestion

Incast Congestion

- Many nodes start to send packets at full rate towards the same destination, almost at the same time (e.g. OLDI services)

- **Incast congestion** occurs at the ToR switch where the node that multiple parties are synchronizing with is connected, and **grows from ToR switches to downstream switches.**

- In CLOS networks many small congestion trees concurrently appear at first-stage switches, later merging at second-stage switches and forming several larger congestion trees.
Analysis of congestion

Incast Congestion

• Traditional approach: The DCN network equipment simply reacts to incast using ECN + PFC and smart buffer management in an attempt to minimize packet loss.

• Problems:
  1. Large DCN networks have more hops, increasing the closed-loop reaction time of ECN.
  2. More traffic in flight makes it difficult for ECN to react until sudden traffic bursts.
  3. PFC generates HoL blocking in upstream switches.
  4. ECN may be triggered at sources not contributing to congestion.

Congestion affects sources that do not cause congestion

Non-congested flows advance at the same speed as congested ones.
Analysis of congestion

Proposed Technologies

• A small number of long duration elephant flows can align in such a way to create queuing delays for the larger number of short but critical mice flows.

• Traditional load balancing (i.e. ECMP) and ECN + PFC are not enough when in-network and incast congestion appear in DCN networks.

• **In-network congestion** can be reduced by suitably:
  • Dimensioning network bisection bandwidth.
  • Applying clever buffers organization.
  • Using some form of load-aware traffic balancing at the sources.

• **Incast congestion** can be alleviated by using destination scheduling.

• Proposed technologies have also limitations when congestion appears.

Outline

• Why congestion isolation is needed?
• Analysis of congestion scenarios
• Limitations of current technologies
• Congestion Isolation in DCNs
• Conclusions
Limitations of current technologies
Load balancing

• Technique to **avoid in-network congestion**.
• Ineffective approaches can actually do the opposite.
• Load balancing **selects a path by hashing the flow identity fields** in the routed packet such that all packets from a particular flow traverse the same path.
  • Equal Cost Multi-Path (ECMP) routing: **Flow granularity is a problem** that may generate elephant flows to traverse and occupy a route in the network for a long period of time.
• ECN mechanism may reduce injection rate of elephant flows, but during the **closed-loop transient period** they may interfere with mice flows, slowing down their advance.
Limitations of current technologies

Load-aware packet-level balancing

• To overcome these issues, several ideas focus on reducing granularity of flows to make better load balancing decisions, based on measuring the congested paths.
• The granularity of load balancing has trade-offs between the uniformity of the distribution and complexity associated with assuring data is delivered in its original order.
• They require some form of signalling congestion to the sources.
• Balancing congested packets through alternative routes may end up moving congestion roots near to end nodes, transforming in-network congestion in incast congestion.

Limitations of current technologies

Load-aware packet-level balancing

Load-aware packet-balancing decision based on congestion indicators

Root of Congestion

Alternative path to destination

Packets Reordering

In-network congestion in a 3-tier CLOS
Limitations of current technologies

Destination scheduling

• The network can assist in eliminating packet loss at the destination by scheduling traffic delivery when it would otherwise be lost.
  • Traditional TCP assess the bandwidth and resource availability by measuring feedback through acks (it works with light load)

• Once incast congestion appears at the destination, delays increase and buffers overflow, throughput is lost and latency rises.
  • Traditional TCP cannot react quick enough to handle incast

• Solution: Requesting data from the source at a rate that it can be consumed without loss.
Limitations of current technologies

Load-aware destination scheduling

• Sources request (send) directly a small amount of unscheduled data to their destinations.

• Destinations schedule a grant response, by means of ACKs, when resources are available to receive the entire transfer.

• There is a RTT request-grant delay that may increase during incast situations.

• Solution: Sources monitor the level of congestion in the network (light, moderate and high) and schedule data injection according to the level of congestion.
Limitations of current technologies

Combined load-aware destination scheduling and balancing

• It is possible to combine destination scheduling and load balancing, depending on whether incast or in-network congestion is monitored.

• Sources measure if the congestion is light, moderate or high, applying different injection rates.

• The idea is to work in load-aware balancing mode until incast congestion appear. When this happens, the network switches to destination-scheduling mode.

• The frequency use of PFC and ECN is reduced
Incast congestion in a 3-tier CLOS

Limitations of current technologies

**Destination scheduling**

Sources inject a moderate amount of data and apply load balancing

Sources schedule big amount of data

Light load

Sources inject a small amount of data (granted by destination)

High load

Moderate load

ECN-marked packets received a certain rate

Root of Congestion

Sources inject a moderate amount of data and apply load balancing
Limitations of current technologies

Consequences

• These technologies may work together to eliminate loss in the cloud data center network [1].

• Load-balancing and destination scheduling are end-to-end solutions incurring in the RTT delays when congestion appear.

• However, there is no time for loss in the network due to congestion and congestion trees grow very quickly.

• Transient congestion may still produce HoL blocking that leads to increase latency, lower throughput and buffers overflow, significantly degrading performance.

• Even using these mechanisms, we still need something to deal with HOL Blocking locally and fast.
Outline

• Why congestion isolation is needed?
• Analysis of congestion scenarios
• Limitations of current technologies
• Congestion Isolation in DCNs
• Conclusions
Congestion Isolation in DCNs

Motivation

• CI is needed to react locally and very fast to immediately eliminate HoL blocking.

• Previous technologies reduce the use of PFC and ECN, but their closed- and open-loop approach cause delays still happening.

• Congestion trees appear suddenly, are difficult to predict (even worse when load balancing is applied) and grow quickly.

• CI can be applied in combination to the previous technologies, improving their behavior.
Congestion Isolation in DCNs

Improvements on current technologies

• CI works well when combined with other CC mechanisms (e.g. e2e congestion control) [1].

• Load balancing makes it difficult to predict when and where congestion points arise.

• Destination Scheduling has RTT-delays that may make feedback information obsolete by the time it reaches the sources.

• CI will complement these technologies working together, making them behave better.

[1] Jesús Escudero-Sahuquillo, Ernst Gunnar Gran, Pedro Javier García, Jose Flich, Tor Skeie, Olav Lysne, Francisco J. Quiles, José Duato: Combining Congested-Flow Isolation and Injection Throttling in HPC Interconnection Networks. ICPP 2011: 662-672
Congestion Isolation in DCNs

Improvements on current technologies

• **Load balancing:**
  • CI complements load balancing as local and fast congested flows isolation reduces the HoL blocking probabilities when load balancing is applied throughout the entire network.
  • **Better decisions for load balancing** can be made once the congested flows are isolated.

• **Destination scheduling:**
  • Transient periods where grants are sent from destinations to sources can be complemented with CI.
  • Fast and local isolation of congested flows reduce RTT-delays of grants.
Congestion Isolation in DCNs
Current technologies also improve CI

- Do the others complement CI? Yes, they make possible to keep the CI required resources low.

- CI require additional resources to keep track of congestion trees at switches.

- If the number of congestion spots grows, switches may end up running out of resources to keep track of them.

- Load balancing and destination scheduling strategies will drain congestion trees faster than using PFC+ECN.

- They will complement (and improve) the CI behavior.

Outline

• Why congestion isolation is needed?
• Analysis of congestion scenarios
• Limitations of current technologies
• Congestion Isolation in DCNs
• Conclusions
Conclusions

• There is a lot of work done in DCNs to deal with congestion and HoL blocking.

• Existing solutions work end-to-end, so that transient congestion may still spoil network performance.

• CI provides a fast reaction to congestion and HoL blocking.

• In fact, CI can work in cooperation with other approaches proposed to deal with congestion, improving their behavior.

• In addition, the proposed approaches can also work in cooperation with CI, increasing its benefits.

• It is very interesting to explore the synergy of all these techniques working together.
P802.1Qcz interworking with other data center technologies

Jesus Escudero-Sahuquillo, Pedro Javier Garcia, Francisco J. Quiles, Jose Duato

IEEE 802.1 Plenary Meeting, San Diego, CA, USA
July 8, 2018
Backup slides
Analysis of congestion
In-Network Congestion transition to Incast

- Consider an already formed congestion tree whose root is located at some intermediate switch egress port, which is connected to a downstream switch.

- Assume another flow reaching that downstream switch through a different ingress port and destined to the same node as the flows in the existing tree.
Analysis of congestion

In-Network Congestion transition to Incast

- If the **aggregated bandwidth required by the root of the congestion tree and the additional flow exceed link bandwidth**, congestion will be detected at some egress port of downstream switch.
- The existing congestion tree is merged with a new branch, moving the root of the congestion tree to an egress port of downstream switch A.