TSVWG Y. Zhuang
Internet-Draft R. Huang
Intended status: Informational Huawei Technologies Co., Ltd.
Expires: January 5, 2020 July 4, 2019

An Open Congestion Control Architecture with network cooperation for RDMA Fabric
draft-zhh-tsvwg-open-architecture-00

Abstract

This document describes an open congestion control architecture with network cooperation (including network proactive and passive control) for high performance RDMA fabric to provide low latency and high throughput for datacenter applications such as the AI computing.

Table of Contents

1. Introduction

Traditionally, RDMA (Remote Direct Memory Access) is running over the closed and expensive InfiniBand (IB) [IB] networks. However, due to the limitation of network scalability and high costs of IB, RDMA traffic is moving to IP/Ethernet as its underlay networks for better scale and low cost. Supporting RDMA over IP/Ethernet using lower price NICs and Switches with reduced latency is important for low latency and high throughput datacenter applications such as AI Computing.

As such, the datacenter networks (DCNs) nowadays is not only providing traffic transmission for tenants using TCP/IP network protocol stack, but also is required to provide RDMA traffic for High Performance Computing (HPC) and distributed storage accessing applications which requires low latency and high throughput. With that said, there are more stringent requirements for basic performance of DCN.

[Requirement] discusses major problems of current RDMA fabric technologies and the requirements for better performance. Also, [HPC] presents the problems of current RDMA fabric from a cloud operators' perspectives.Based on that, this document proposes an open

congestion control architecture of hosts and networks with network cooperation (including network proactive and passive control) for the high performance RDMA fabric to provide better congestion control.

The scalability and compatibility of congestion control under the proposed architecture are also discussed in order to provide incremental upgrade of the current RDMA technologies.

Discussions of new congestion control algorithms and improved active queue management (AQM) are out of scope for this document.

2. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Abbreviations

IB - InfinitBand

HPC - High Performance Computing

ECN - Explicit Congestion Notification

AI/HPC - Artificial Intelligence/High-Performance computing

RDMA - Remote Direct Memory Access

NIC - Network Interface Card

AQM - Active Queue Management

NP - Notification Point

CP - Congestion Point

RP - Reaction Point

4. Design Principle for high performance RDMA fabric

Based on the [Requirement] and [HPC], the architecture design should follow some principles:

- Can adopt enhancements to provide better performance than existing technologies, such as better latency, convergence and handling of packet loss.

- Can support both RoCEv2 and iWARP [RFC5040] as RDMA transports.

- Can support mixture of RDMA traffics and normal TCP traffics.

- Can provide better interoperability between vendors while keep flexibility.

- Do not modify or provide limited modification to RDMA data plane.

- Be compatible with legacy devices.

- Be easy to deploy new congestion control algorithms.

5. Architecture Overview

The architecture is shown in Figure 1. It composes of hosts (i.e. sender/receiver NICs) and network nodes (i.e. switches).

```
  Sender(RP)                                           Receiver(NP)

''''''''''''''''''''''''''
''''''''''''''''''''''''''
'   +---+  +---+         '                             '   +---+  +---+
'
'   |CC1|  |CC1| ...     '                             '   |CC1|  |CC1| ...
'
'   +-*-+  +-*-+         '                             '   +-*-+  +-*-+
'
'     *      *           '                             '     *      *
'
'+----*------*---------+ '                             ' +----*------*---------
+'
'|  Congestion control | '      Switch(CP and NP)     ' |  Congestion control
|'
'|  Engine             | '                             ' |  Engine
|'
'+---------------------+ '      ''''''''''''''''''     ' +---------------------
+'
'+--------++-----------+ '      ' +-----------+ '      ' +-----------++--------
+'
'|rate-co ||net-control|<-------- |net-control| '      ' |net-control||rate-co
|'
'|ntrol;  ||           | '      ' |           | '      ' |           ||ntrol;
|'
'|loss-re |+-----------+ '      ' +-----------+ '      ' +-----------+|loss-re
|'
'|covery  |+-----------+ '      '               '      ' +-----------+|covery
|'
'|        ||nic-control|<........              <........|nic-control||
|'
'|        ||           | '      '               '      ' |           ||
|'
'+--------++-----------+ '      '               '      ' +-----------++--------
+'
'+---------------------+ '      '               '      ' +---------------------
+'
'|        data         |========>              ======> |        data
|'
'|                     | '      '               '      ' |
|'
'+---------------------+ '      '               '      ' +---------------------
+'
'                        '      '               '      '
'
''''''''''''''''''''''''''      ''''''''''''''''''
''''''''''''''''''''''''''
```

```
<-------- Net2Nic control channel   ========> RDMA stream
<........ Nic2Nic control channel   ********  System APIs
```

Figure 1. The open congestion control architecture with network cooperation

Sender and Receiver are both NICs. Within the architecture, the NICs are proposed to introduce two new interfaces: 1) an interface for the operators to install/manage congestion control algorithms which can share the local transmit function blocks such as rate control and loss recovery to facilitate the deployment of new congestion control algorithms and the management of different algorithms while regardless of the detailed hardware implementation; 2) an interface for net-control module inside network nodes (e.g. switches) to signal back to senders, and further incorporate the collected information into the local transmit control.

For the interface to network nodes, we introduce a new NET to NIC control channel, in which the control message is initiated and sent by the net-control module inside a switch instead of the receiver. Since most congestion happens on network nodes, the switch noted as congestion point (CP) in Fig.1 should be the point aware of the ongoing or expected congestion. The advantage of doing so, is to provide more accurate congestion information and how to prevent or resolve the congestion based on traffic traversing and resources allocated on the network nodes directly.

The NIC to NIC control channel signaled by dotted link presents a logical channel for legacy NIC to NIC control notification. It can be for example CNP message for RoCEv2 or flags/fields in TCP headers for iWARP. The RDMA data streams is indicated by bold line and works as it is. However, some extensions might be needed to implement the new interfaces which is out of the scope for this document.

5.1. Roles and Functionalities

5.1.1. Sender NIC

As the reaction point (RP) of the architecture, the sender NIC can deploy/manage the congestion control algorithms based on system configurations or the negotiation with remote NICs. When congestion happens, it accordingly adjusts its sending rate based on the used congestion control algorithm and signaled feedbacks from both the network nodes and/or the receiver's NIC.

5.1.2. Switch

Switch is the congestion point, which detects the network congestion based on some metrics, such as queue length or measured latency on the path or traffic patterns it might have learnt.

For a legacy switch with ECN enabled, it can mark CE in the IP header of RDMA traffics when congestion exists to notify the receivers. When the condition is getting worse, it either uses PFC or discard the packet based on some AQM policies. For legacy switches without ECN, it discards packets when congestion happens.

For a switch with net-control module, called a net-control switch here, it can act as the notification point (NP) which can initiate the control message and delivery it through the NET to NIC control channel back to the sender, which adjusts its sending rate accordingly. Net-control switches can be deployed in any places of a DCN fabric, e.g., TOR or spine.

5.1.3. Receiver NIC

Receiver NIC might negotiate with the sender NIC on the congestion control capability. It is also the notification point (NP). Based on the ECN mark or lost packets, it discovers congestion and send congestion information back to the sender through NIC to NIC control

channel to adjust sending rate. In RoCEv2, the CNP message is used for the NIC to NIC control.

5.2. Interfaces

The architecture introduces two interfaces on NICs and one interface on the network node for the open control as shown in Figure 2. As for the NIC, one interface is for deploying/managing different congestion controls while the other is to communicate with the network control module on switches. For the switch, the proposed interface is dedicated for control of network congestions back to the senders.

```
          Sender(RP)

''''''''''''''''''''''''''
'   +---+  +---+         '
'   |CC1|  |CC1| ...     '
'   +-*-+  +-*-+         '
'     *      *           '
'+----*------*---------+ '      Switch(CP and NP)
'|  Congestion control | '
'|  Engine             | '      ''''''''''''''''''
'+---------------------+ '      ' +-----------+ '
'+--------++-----------+ '      ' |net-control| '
'|rate-co ||net-control|<-------- |net-control| '
'|ntrol;  ||           | '      ' |           | '
'|loss-re |+-----------+ '      ' +-----------+ '
'|covery  |+-----------+ '      '               '
'|        ||nic-control| '      '
'|        ||           | '      '               '
'+--------++-----------+ '      '               '
'+---------------------+ '      '               '
'|        data         | '      '
'|                     | '      '               '
'+---------------------+ '      '               '
'                        '      '               '
''''''''''''''''''''''''''      ''''''''''''''''''

 <-------- Net2Nic interface   ********  system CC interface
```

Figure 2. Imported NIC interfaces and network interface

5.2.1. NIC interfaces

To cope with various scenarios and facilitate the deployment of new congestion control algorithms, it would be good if NICs will be able to deploy congestion controls and further manage and configure them in a common way. The idea to provide a system CC interface is that the cloud operators can deploy/manage congestion control algorithms on NICs based on the traffic patterns as well as the network resources. Then the NICs might negotiate the congestion control capability with each other.

The function blocks within in the NIC are logic components, not indicating any specific implementation. A congestion control engine acts as a platform to provide a system CC interface to deploy different CCs and then map to local actions and communicate with local function blocks to provide congestion control operations.

Ideally, local functions related to congestion controls will be implemented as function blocks and interact with each other through internal interfaces to achieve the final congestion controls. As such, CCs can share common local operations and it would be easy for developers to develop and deploy new CCs regardless of detailed local implementations. The design of the CC Engine and local function blocks are out of scope for this document. An example of the design and implantation can be found in [HotCocoa] .

For now, the local function blocks can include rate-control and loss-recovery, as well as two blocks to deal with congestion control information from the interface to NIC control and the interface to NET control respectively.

The other proposed interface of the NIC is to the NET control (Net2Nic control channel), which is used to collect congestion information from the network nodes to be further incorporated to the congestion control of sender NICs.

5.2.2. Network interface

To achieve more accurate congestion control and ways to prevent or resolve the congestion based on traffic traversing, as indicated in Figure 2, the net-control switch will provide a network interface (Net2Nic interface), by which net-control module inside the node can signal back to the senders.

The definition of Net2Nic control channel messages and processes are out of scope for this document. It relies on the design of net-control module which is responsible for dealing with network congestions and exposing what precise information to the sender.

6. Compatibility Consideration

6.1. Negotiate the congestion control capability

The host might negotiate their supported congestion control capability during the session setup phase.

However, it should use the existing way of congestion control as default to provide compatibility with legacy devices.

The net-control switches should be capable of both legacy control and NET to NIC control. The capability negotiation between NICs and Switches can be considered either some in-band ECN-like negotiations or out-of-band individual message negotiations.

6.2. Co-exist with current NIC to NIC control channel

In this architecture, NET to NIC control channel can co-exist with NIC to NIC control channel. It can be an additional control channel for better congestion control.

Once the NET-to-NIC channel of a sender is enabled on a switch, it will signal the congestion information back to the sender through this channel. While for hosts without NET control, the switch works the same as the legacy switches when congestion happens.

For receivers that detect the congestion based on lost packets, packets marked CE due to congestion on legacy network nodes, or the exhaustion of local resources, they can still notify the senders according to the congestion control algorithms. The senders evaluate the messages based on its local polices, e.g., if it receives a message from the net-control interface prior to the message from the receiver in certain period, it may decide to make decision based on the net-control message; While if there's no net-control message received, the sender may react according to the message from the receiver.

Please note that NET to NIC control channel SHOULD be implemented as an option rather than a mandatory feature.

7. Security Considerations

TBD

8. Manageability Consideration

TBD

9. IANA Considerations

No IANA action

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

10.2. Informative References

[HotCocoa] Arashloo, M. T., Ghobadi, M., Rexford, J., and D. Walker, "HotCocoa: Hardward Congestion Control Abstractions", 11 2017, <https://www.cs.princeton.edu/~jrex/papers/hotcocoa17.pdf>.

[HPC] Cardona, O., "Towards Hyperscale High Performance Computing with RDMA", 6 2019, <https://pc.nanog.org/static/published/meetings/NANOG76/1999/20190612_Cardona_Towards_Hyperscale_High_v1.pdf>.

[IB] "Infiniband Trade Association. InfiniBandTM Architecture Specification Volume 1 and Volume 2.", <https://cw.infinibandta.org/document/dl/7781>.

[Requirement] Chen, F., Sun, W., Yu, X., and R. Even, "Data Center Congestion Management requirements", 6 2019, <https://datatracker.ietf.org/doc/html/draft-yueven-tsvwg-dccm-requirements>.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, DOI 10.17487/RFC3168, September 2001, <https://www.rfc-editor.org/info/rfc3168>.

[RFC5040] Recio, R., Metzler, B., Culley, P., Hilland, J., and D. Garcia, "A Remote Direct Memory Access Protocol Specification", RFC 5040, DOI 10.17487/RFC5040, October 2007, <https://www.rfc-editor.org/info/rfc5040>.

Authors' Addresses

Yan Zhuang
Huawei Technologies Co., Ltd.

Email: zhuangyan.zhuang@huawei.com

Rachel Huang
Huawei Technologies Co., Ltd.

Email: rachel.huang@huawei.com