Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 20 November 2025

X. Xu
China Mobile
S. Hegde
Juniper
Z. He
Broadcom
J. Wang
Centec
H. Huang
Huawei
Q. Zhang
H3C
H. Wu
Ruijie Networks
Y. Liu
Y. Xia
Tencent
P. Wang
Baidu
T. Li
IEIT SYSTEMS

19 May 2025

Fully Adaptive Routing Ethernet using BGP
draft-xu-idr-fare-03

Abstract

Large language models (LLMs) like ChatGPT have become increasingly popular in recent years due to their impressive performance in various natural language processing tasks. These models are built by training deep neural networks on massive amounts of text data, as well as visual and video data, often consisting of billions or even trillions of parameters. However, the training process for these models can be extremely resource-intensive, requiring the deployment of thousands or even tens of thousands of GPUs in a single AI training cluster. Therefore, three-stage or even five-stage CLOS networks are commonly adopted for AI networks. The non-blocking nature of the network become increasingly critical for large-scale AI models. Therefore, adaptive routing is necessary to dynamically distribute the traffic to the same destination over multiple equal-cost paths, based on the network capacity and even congestion information along those paths.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 20 November 2025.

Copyright Notice

Table of Contents

1. Introduction

Large language models (LLMs) like ChatGPT have become increasingly popular in recent years due to their impressive performance in various natural language processing tasks. These models are built by training deep neural networks on massive amounts of text data, as well as visual and video data, often consisting of billions or even trillions of parameters. However, the training process for these models can be extremely resource-intensive, requiring the deployment of thousands or even tens of thousands of GPUs in a single AI training cluster. Therefore, three-stage or even five-stage CLOS networks are commonly adopted for AI networks. Furthermore, In rail-optimized CLOS network topologies with standard GPU servers (HB domain of eight GPUs), the Nth GPUs of each server in a group of servers are connected to the Nth leaf switch, which provides higher bandwidth and non-blocking connectivity between the GPUs in the same rail. In rail-optimized network topology, most traffic between GPU servers would traverse the intra-rail networks rather than the inter-rail networks. In addition, whether in rail-optimal or rail-free networks, collective communication job schedulers always opt to schedule jobs with network topology awareness to minimize the amount of traffic going to the upper layers of the network.

The non-blocking nature of the network, particularly at the lower layers, is essential for large-scale AI training clusters. AI workloads are usually very bandwidth-hungry and often generate several large data flows simultaneously. If traditional hash-based ECMP load balancing is used without optimization, it can lead to serious congestion and high latency in the network when multiple large data flows are directed to the same link. This congestion can result in longer-than-expected model training times, as job completion time depends on worst-case performance. Therefore, adaptive routing is necessary to dynamically distribute traffic to the same destination across multiple equal-cost paths, taking into account network capacity and even congestion along these paths. In essence, adaptive routing is a capacity- and even congestion-aware dynamic path selection algorithm.

Furthermore, to reduce the congestion risk to the maximum extent, the routing should be more granular if possible. Flow-granular adaptive routing still has a certain statistical possibility of congestion. Therefore, packet-granular adaptive routing is more desirable although packet spray would cause out-of-order delivery issues. A flexible reordering mechanism must be put in place（e.g., egress ToRs

or the receiving servers). Recent optimizations for RoCE and newly invented transport protocols as alternatives to RoCE no longer require handling out-of-order delivery at the network layer. Instead, the message processing layer is used to address it.

To enable adaptive routing, no matter whether flow-granular or packet-granular adaptive routing, it is necessary to propagate network topology information, including link capacity and path capacity across the CLOS network. Therefore, it seems straightforward to use link-state protocols such as OSPF or ISIS as the underlay routing protocol in the CLOS network, instead of BGP. How to leverage OSPF or ISIS to achieve adaptive routing has been described in [I-D.xu-lsr-fare]. However, some data center network operators have been used to the use of BGP as the underlay routing protocol of data center networks [RFC7938]. Therefore, there does exist a need to leverage BGP to achieve adaptive routing as well.

Hence, this document defines a new extended community referred to as Path Bandwidth Extended Community, and describes how to use this extended community to carry end-to-end path bandwidth within the data center fabric so as to achieve adaptive routing.

Note that while adaptive routing, especially at the packet-granular level can help reduce congestion between switches in the network, thereby achieving a non-blocking fabric, it does not address the incast congestion issue which is commonly experienced in last-hop switches that are connected to the receivers in many-to-one communication patterns. Therefore, a congestion control mechanism is always necessary between the sending and receiving servers to mitigate such congestion.

1.1. Comparison with Related Works

[I-D.ietf-idr-link-bandwidth] outlines a method for implementing weighted ECMP load-balancing based on the bandwidth of the EXTERNAL (DMZ) link, which is conveyed in the non-transitive link bandwidth extended community. However, it is not feasible to enable adaptive routing directly using the non-transitive link bandwidth extended community due to the following constraints mentioned in [I-D.ietf-idr-link-bandwidth]. "No more than one link bandwidth extended community SHALL be attached to a route. Additionally, if a route is received with a link bandwidth extended community and the BGP speaker sets itself as next-hop while announcing that route to other peers, the link bandwidth extended community should be removed. The extended community is optional non-transitive."

[I-D.ietf-bess-ebgp-dmz] removes the previous restriction that the EXTERNAL (DMZ) link bandwidth extended community could not be sent across AS boundaries. Additionally, when receiving multiple equal-cost BGP paths towards the external network (e.g., the WAN), the best path among them will be advertised to eBGP peers with the transitive link bandwidth extended community filled with the cumulative bandwidth of the multiple external links. Since the approach as described in this document is based on the assumption that "The total BW available towards WAN is significantly lower than the total BW within the fabric," the internal path bandwidth within the fabric is not taken into account when performing weighted ECMP load-balancing.

[I-D.ietf-bess-evpn-unequal-lb] describes an EVPN-dedicated extended community and an EVPN link-bandwidth sub-type of the above EVPN-dedicated extended community for EVPN weighted ECMP load-balancing. Additionally, the document defines different ways to express the link bandwidth.

The three previous documents explain how to use the extended community to carry the bandwidth of the external links towards the outside of the fabric (such as WAN, services bound to anycast address, or multi-homed VPN sites) for weighted ECMP load-balancing. In contrast, this document explains how to use the extended community to carry the end-to-end path bandwidth within the data center fabric for weighted ECMP load-balancing.

2. Terminology

This memo makes use of the terms defined in [RFC4360].

3. Path Bandwidth Extended Community

The Path Bandwidth Extended Community is used to indicate the minimum bandwidth of the path towards the destination. It is a new IPv4 Address Specific Extended Community that can be transitive or non-transitive.

The value of the high-order octet of this extended type is either 0x01 or 0x41. The low-order octet of this extended type is TBD.

The Value field consists of two sub-fields:

- Global Administrator sub-field: This sub-field contains the router ID of the advertising router that appends the path bandwidth extended community or updates the path bandwidth value of the existing path bandwidth extended community.

Local Administrator sub-field: This sub-field contains the path bandwidth value in IEEE floating point format with units of Gigabytes per second (GB/s).

4. Solution Description

4.1. Adaptive Routing in 3-stage CLOS

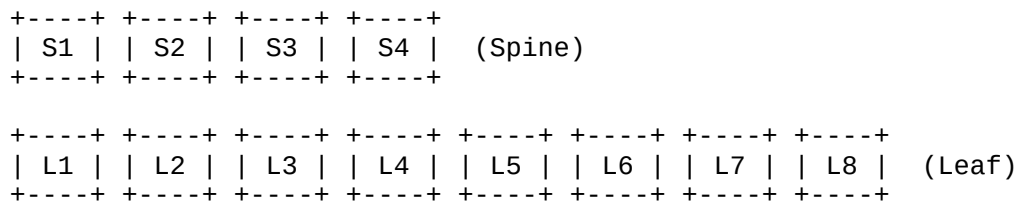

Figure 1

(Note that the diagram above does not include the connections between nodes. However, it can be assumed that leaf nodes are connected to every spine node in the above CLOS topology.)

In a three-stage CLOS network as shown in Figure 1, also known as a leaf-spine network, each leaf node would establish eBGP sessions with all spine nodes.

All nodes are enabled for adaptive routing.

When a leaf node, such as L1, advertises the route to a specific IP prefix that it originates, it will attach a transitive path bandwidth extended community filled with a maximum bandwidth value.

Upon receiving the above advertisement, a spine node, such as S1, SHOULD determine the minimum value between the bandwidth of the link towards the advertising node (e.g., L1) and the value of the path bandwidth extended community carried in the received route, and then update the path bandwidth extended community with the above minimum value before readvertising that route to remote eBGP peers. Once S1 receives multiple equal-cost routes for a given prefix from multiple leaf nodes (e.g., L1 and L2 in the server multi-homing scenario), for each route, it SHOULD determine the minimum value between the bandwidth of the link towards the advertising node and the value of the path bandwidth extended community carried in the received route, and then use that minimum bandwidth value as a weight value for that route when performing weighted ECMP load-balancing. When

readvertising the route for that prefix to remote eBGP peers further, the path bandwidth extended community would be updated with the sum of the minimum bandwidth value of each route.

When a leaf node, such as L8, receives multiple equal-cost routes for that prefix from spine nodes (e.g., S1, S2, S3 and S4), for each route, it will determine the minimum value between the bandwidth of the link towards the advertising node and the value of the path bandwidth extended community carried in the received route, and then use that minimum bandwidth value as a weight value for that route when performing weighted ECMP load-balancing.

Note that weighted ECMP load-balancing according to path bandwidth SHOULD NOT be performed unless all equal-cost routes for a given prefix carry path bandwidth extended community.

4.2. Adaptive Routing in 5-stage CLOS

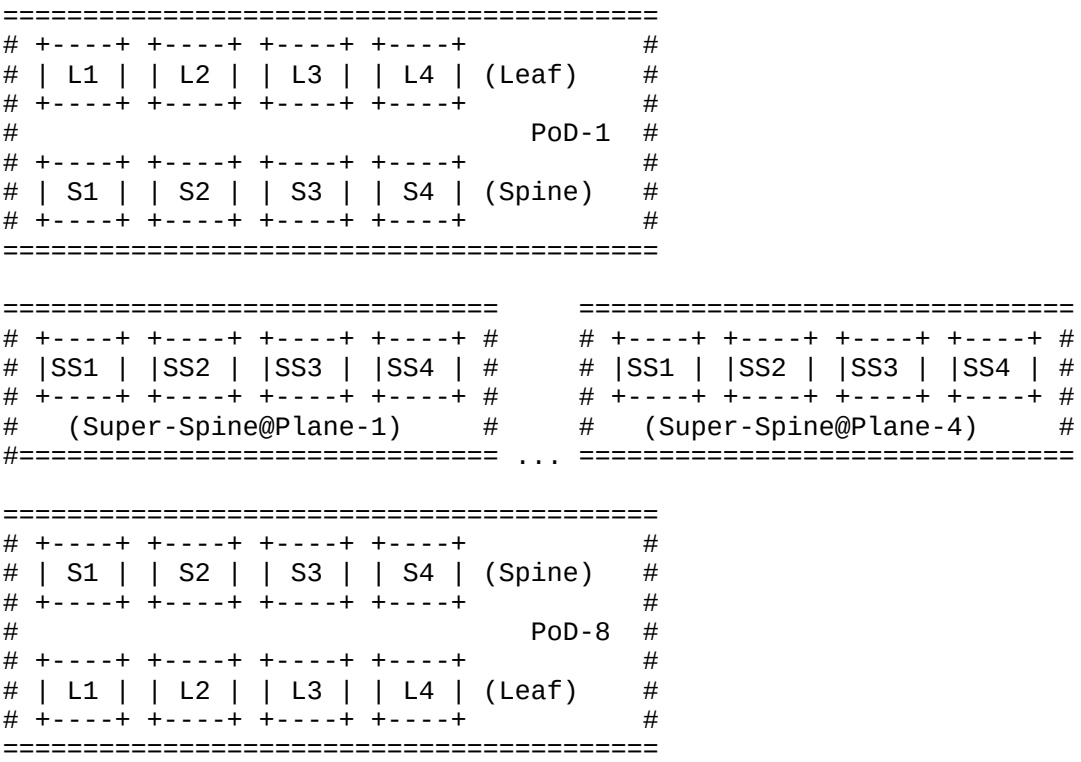

Figure 2

(Note that the diagram above does not include the connections between nodes. However, it can be assumed that the leaf nodes in a given PoD are connected to every spine node in that PoD. Similarly, each spine node (e.g., S1) is connected to all super-spine nodes in the corresponding PoD-interconnect plane (e.g., Plane-1).)

For a five-stage CLOS network as illustrated in Figure 2, each leaf node would establish eBGP sessions with all spine nodes of the same PoD while each spine node would establish eBGP sessions with all super-spine nodes in the corresponding PoD-interconnect plane.

When a given leaf node, such as L1@PoD-1, advertises the route for a specific IP prefix that it originates, it will attach a transitive path bandwidth extended community filled with a maximum bandwidth value.

Upon receiving the above route advertisement, a spine node, such as S1@PoD-1, will determine the minimum value between the bandwidth of the link towards the advertising node (e.g., L1@PoD-1) and the value of the path bandwidth extended community carried in the route, and then update the path bandwidth extended community with the above minimum value before advertising that route to its peers. Once S1@PoD-1 receives multiple equal-cost routes for a given prefix from multiple leaf nodes (e.g., L1 and L2@PoD-1 in the server multi-homing scenario), for each route, it will determine the minimum value between the bandwidth of the link towards the advertising node and the value of the path bandwidth extended community carried in the route, and then use that minimum bandwidth value as a weight value for that route when performing weighted ECMP load-balancing. When advertising the route for that prefix to remote peers further, the path bandwidth extended community would be updated with the sum of the bandwidth value of each received route.

When a given super-spine node, such as SS1@Plane-1, receives the above route advertised from S1@PoD-1, it will not update the transitive path bandwidth extended community when advertising that route to its peers. Additionally, it COULD optionally attach another path bandwidth extended community which is non-transitive to indicate the bandwidth of the link towards the advertising router of the received route (i.e., S1@PoD-1).

When a given spine node in another PoD, such as S1@PoD-8, receives multiple equal-cost routes for a given prefix from super-spine nodes in Plane-1 (e.g., SS1, SS2, SS3 and SS4@Plane-1), once each route contains a non-transitive path bandwidth extended community, for each route, it will determine the minimum value between the bandwidth of the link towards the advertising node and the bandwidth value of the non-transitive path bandwidth extended community carried in the

route, and then use that minimum value as a weight value for that route when performing weighted ECMP load-balancing. Otherwise, it would perform ECMP load-balancing by default.

When advertising that route to its peers, it will not update the value of the transitive path bandwidth extended community by default (Note that the transitive path bandwidth extended community of those multiple equal-cost routes carry the same value that was set by S1@PoD-1). In the case where each route contains a non-transitive path bandwidth extended community, the above spine node COULD optionally update the value of the transitive path bandwidth extended community with the total bandwidth value of all paths towards the next-next hop (e.g., the paths towards S1@PoD-1 via SS1, SS2, SS3 and SS4@Plane-1) if the latter is smaller than the former.

When a given leaf node in PoD-8, such as L1@PoD-8, receives multiple equal-cost routes for that prefix from multiple spine nodes (e.g., S1, S2, S3 and S4@PoD-8), for each route, it will determine the minimum value between the bandwidth of the link towards the advertising node and the value of the path bandwidth extended community carried in the route, and then use that minimum value as a weight value for that route when performing weighted ECMP load-balancing.

Note that weighted ECMP load-balancing according to path bandwidth SHOULD NOT be performed unless all equal-cost routes for a given prefix carry path bandwidth extended community.

5. Acknowledgements

TBD.

6. IANA Considerations

TBD.

7. Security Considerations

TBD.

8. References

8.1. 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>.

[RFC4360] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended Communities Attribute", RFC 4360, DOI 10.17487/RFC4360, February 2006, <https://www.rfc-editor.org/info/rfc4360>.

8.2. Informative References

[I-D.ietf-bess-ebgp-dmz]
Satya, M. R., Vayner, A., Gattani, A., Kini, A., Tantsura, J., and R. Das, "Cumulative DMZ Link Bandwidth and load-balancing", Work in Progress, Internet-Draft, draft-ietf-bess-ebgp-dmz-06, 2 January 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-bess-ebgp-dmz-06>.

[I-D.ietf-bess-evpn-unequal-lb]
Malhotra, N., Sajassi, A., Rabadan, J., Drake, J., Lingala, A. R., and S. Thoria, "Weighted Multi-Path Procedures for EVPN Multi-Homing", Work in Progress, Internet-Draft, draft-ietf-bess-evpn-unequal-lb-25, 13 May 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-bess-evpn-unequal-lb-25>.

[I-D.ietf-idr-link-bandwidth]
Mohapatra, P., Fernando, R., Das, R., Satya, M. R., Mishra, M. P., and R. J. Szarecki, "BGP Link Bandwidth Extended Community", Work in Progress, Internet-Draft, draft-ietf-idr-link-bandwidth-11, 3 March 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-idr-link-bandwidth-11>.

[I-D.xu-lsr-fare]
Xu, X., Hegde, S., He, Z., Wang, J., Huang, H., Zhang, Q., Wu, H., Liu, Y., Xia, Y., and P. Wang, "Fully Adaptive Routing Ethernet using LSR", Work in Progress, Internet-Draft, draft-xu-lsr-fare-03, 1 September 2024, <https://datatracker.ietf.org/doc/html/draft-xu-lsr-fare-03>.

[RFC7938] Lapukhov, P., Premji, A., and J. Mitchell, Ed., "Use of BGP for Routing in Large-Scale Data Centers", RFC 7938, DOI 10.17487/RFC7938, August 2016, <https://www.rfc-editor.org/info/rfc7938>.

Authors' Addresses

Xiaohu Xu
China Mobile
Email: xuxiaohu_ietf@hotmail.com

Shraddha Hegde
Juniper
Email: shraddha@juniper.net

Zongying He
Broadcom
Email: zongying.he@broadcom.com

Junjie Wang
Centec
Email: wangjj@centec.com

Hongyi Huang
Huawei
Email: hongyi.huang@huawei.com

Qingliang Zhang
H3C
Email: zhangqingliang@h3c.com

Hang Wu
Ruijie Networks
Email: wuhang@ruijie.com.cn

Yadong Liu
Tencent
Email: zeepliu@tencent.com

Yinben Xia
Tencent
Email: forestxia@tencent.com

Peilong Wang
Baidu
Email: wangpeilong01@baidu.com

Tiezheng Li
IEIT SYSTEMS
Email: litiezheng@ieisystem.com