Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: 21 November 2025

X. Xu
China Mobile
Z. He
Broadcom
H. Wang
Moore Threads
T. Zhou
Resnics Technology
Y. Yang
Centec
Y. Xia
Tencent
P. Wang
Baidu
Y. Zhuang
Huawei Technologies
F. Yang
Cloudnine Information Technologies
C. Li
Metanet Networking Technology
X. Wang
Ruijie Networks

20 May 2025

Fully Adaptive Routing Ethernet in Scale-Up Networks
draft-xu-rtgwg-fare-in-sun-01

Abstract

The Mixture of Experts (MoE) has become a dominant paradigm in transformer-based artificial intelligence (AI) large language models (LLMs). It is widely adopted in both distributed training and distributed inference. Furthermore, the disaggregation of the prefill and decode phases is highly beneficial and is considered a best practice for distributed inference models; however, this approach depends on highly efficient Key-Value (KV) cache synchronization. To enable efficient expert parallelization and KV cache synchronization across dozens or even hundreds of Graphics Processing Units (GPUs) in MoE architectures, an ultra-high-throughput, ultra-low-latency AI scale-up network (SUN) that can efficiently distribute data across all network planes is critical. This document describes how to extend the Weighted Equal-Cost Multi-Path (WECMP) load-balancing mechanism, referred to as Fully Adaptive Routing Ethernet (FARE), which was originally designed for scale-out networks, to scale-up networks.

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 21 November 2025.

Copyright Notice

Table of Contents

1. Introduction 3
2. Terminology 4
3. Solution Description 4
  3.1. Per-Flow Weighted Load Balancing 5
  3.2. Per-Packet Weighted Load Balancing 6
4. Acknowledgements 6
5. IANA Considerations 6
6. Security Considerations 6
7. References 6
  7.1. Normative References 6
  7.2. Informative References 6

Authors' Addresses 7

1. Introduction

The Mixture of Experts (MoE) has become a dominant paradigm in transformer-based artificial intelligence (AI) large language models (LLMs). It is widely adopted in both distributed training and distributed inference. Furthermore, the disaggregation of the prefill and decode phases is highly beneficial and is considered a best practice for distributed inference models; however, this approach depends on highly efficient Key-Value (KV) cache synchronization.

To enable efficient expert parallelization and KV cache synchronization across dozens or even hundreds of Graphics Processing Units (GPUs) in MoE architectures, an ultra-high-throughput, ultra-low-latency AI scale-up network (SUN) is indispensable. This network serves as the interconnection fabric, allowing GPUs to function as a unified super GPU, referred to as a SuperPod. The scale-up network is fundamental for efficiently transporting substantial volumes of communication traffic within the SuperPod. It includes 1) all-to-all traffic for Expert Parallelism (EP) communication, enabling experts running on GPU servers to exchange information seamlessly, and 2) all-reduce traffic for Tensor Parallelism (TP) communication, ensuring consistent tensor values across GPUs during training and inference.

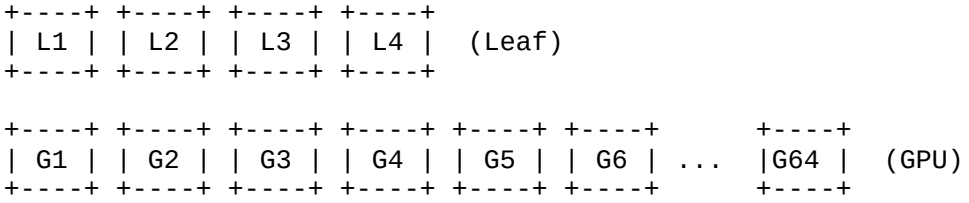

Figure 1

(Note that the diagram above does not include the connections between GPUs and leaf switches. However, it can be assumed that GPUs are connected to every leaf switch in the above one-tier scale-up network topology.)

As shown in Figure 1, it's a 64-GPU SuperPoD that consists of 64 GPUs and four leaf switches with high radix (e.g., 128 400G QSFP112 ports). To achieve inter-GPU bandwidths of several terabits per second (Tbps) or higher, each GPU is typically equipped with multiple

scale-up network ports (e.g., four 800 Gbps OSFP ports). Each port connects to a separate scale-up leaf switch via Y-cables, forming four distinct network planes.

In such multi-plane scale-up networks, achieving ultra-high bandwidth and ultra-low latency requires two key strategies. First, efficiently distributing data across all network planes is critical. For instance, if an 800G port on a GPU fails, traffic destined for that GPU over the faulty plane must immediately cease. If one 400G sub-cable of a given 800G Y-cable malfunctions, halving the bandwidth of the affected plane, traffic on that plane between the relevant GPU pair should be proportionally reduced. Second, incast traffic patterns inherent to all-to-all communication may cause congestion on the egress ports of a last-hop switch; therefore, a more efficient congestion management mechanism is required.

This document describes how to extend the Weighted Equal-Cost Multi-Path (WECMP) load-balancing mechanism, referred to as Fully Adaptive Routing Ethernet (FARE) in [I-D.xu-idr-fare], which was originally designed for scale-out netowrks, to scale-up networks.

2. Terminology

This memo makes use of the terms defined in [RFC2119].

3. Solution Description

Each pair of GPUs establishes multiple Remote Direct Memory Access (RDMA) Queue Pairs (QPs) for data transmission by using the loopback addresses of the GPUs. Note that upper-layer adaptations can enable memory semantic operations (load/store/atomic) based on RDMA message semantics. However, implementation details are beyond the scope of this document.

By acting as Border Gateway Protocol (BGP) speakers, GPU servers exchange BGP routes with connected switches of different planes (e.g., advertising the reachability of their loopback addresses). This allows servers to obtain route reachability and available path bandwidth information for each destination GPU, enabling WECMP load balancing across multiple planes.

Of course, some data-plane health check mechanisms running directly between GPUs, spanning each network plane, could be leveraged to speed up route convergence, especially in cases where a network plane is broken.

3.1. Per-Flow Weighted Load Balancing

For per-flow weighted load balancing, a minimum of one QP per sub-port must be established across each network plane between a given pair of GPUs. Each QP utilizes a unique UDP source port to differentiate traffic flows. For example, if a physical port is divided into m sub-ports and there are n distinct network planes (where n 1), at least m × n QPs must be instantiated— one QP per sub-port per plane— to ensure proper flow distribution across all available paths. Consequently, the traffic between each pair of GPUs is balanced across all available network planes (a.k.a., QPs bound to those network planes) according to the path bandwidth values associated with those network planes. In addition, the traffic distributed to a given network plane (a.k.a., QPs bound to that network plane) is further evenly distributed at the QP granularity across available links connected to that network plane by the source GPU server.

GPU servers could utilize a connection tracking table— a technique commonly used in Server Load Balancer (SLB) systems— to implement per-flow weighted load balancing. When the path bandwidth of a route via a specific network plane to a destination GPU degrades— due to events such as network plane failures or partial link outages— existing Queue Pairs (QPs) traversing unaffected planes retain their established forwarding paths. Meanwhile, the source GPU must release all or a subset of QPs associated with the affected network plane, adjusting their usage in strict accordance with updated weight values that reflect the reduced capacity. Conversely, when path bandwidth via a previously degraded network plane recovers— such as after failed links or planes are restored— the source GPU reinstates all or a subset of QPs traversing that plane. This reestablishment is performed in alignment with the revised weight values, which now reflect the increased available bandwidth, ensuring optimal traffic distribution across all operational network paths.

The expiration timer for connection tracking entries can be configured based on the traffic characteristics of collective communications, such as periodic burst patterns. For example, entries corresponding to QP can expire during the interval between consecutive bursts. This ensures that each batch of data transferred between GPU pairs is distributed according to the current weight values of available paths.

The switch within each network plane should perform per-flow load balancing as well to ensure ordered packet delivery for all QPs.

3.2. Per-Packet Weighted Load Balancing

For per-packet weighted load balancing, all QPs established between a pair of GPUs must support disordered packet delivery (e.g., via the Direct Data Placement mechanism as described in [RFC7306] .) Similarly, the traffic between each pair of GPUs is balanced across all available network planes according to the path bandwidth values associated with those network planes. In this mode, a single QP per network plane between a given GPU pair suffices, with packets sprayed evenly across all available links connected to that network plane by the source GPU server.

The switch within each network plane could perform per-packet load balancing since disordered packet delivery is acceptable for all QPs.

4. Acknowledgements

TBD.

5. IANA Considerations

TBD.

6. Security Considerations

TBD.

7. References

7.1. 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>.

7.2. Informative References

[I-D.xu-idr-fare] Xu, X., Hegde, S., He, Z., Wang, J., Huang, H., Zhang, Q., Wu, H., Liu, Y., Xia, Y., Wang, P., and Tiezheng, "Fully Adaptive Routing Ethernet using BGP", Work in Progress, Internet-Draft, draft-xu-idr-fare-02, 1 September 2024, <https://datatracker.ietf.org/doc/html/draft-xu-idr-fare-02>.

[RFC7306] Shah, H., Marti, F., Noureddine, W., Eiriksson, A., and R. Sharp, "Remote Direct Memory Access (RDMA) Protocol Extensions", RFC 7306, DOI 10.17487/RFC7306, June 2014, <https://www.rfc-editor.org/info/rfc7306>.

Authors' Addresses

Xiaohu Xu
China Mobile
Email: xuxiaohu_ietf@hotmail.com

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

Hua Wang
Moore Threads
Email: wh@mthreads.com

Tianyou Zhou
Resnics Technology
Email: tzhou@resnics.com

Yongtao Yang
Centec
Email: yangyt@centec.com

Yinben Xia
Tencent
Email: forestxia@tencent.com

Peilong Wang
Baidu
Email: wangpeilong01@baidu.com

Yan Zhuang
Huawei Technologies
Email: zhuangyan.zhuang@huawei.com

Fajie Yang
Cloudnine Information Technologies
Email: yangfajie@cloudnineinfo.com

Chao Li
Metanet Networking Technology
Email: lichao22@ieisystem.com

Wang Xiaojun
Ruijie Networks
Email: wxj@ruijie.com.cn