Building AI Network Fabrics

Peter Lundqvist

peter@arista.com

Agenda

- Al Basics
- Deep Learning
- Networking requirements
- Architectures for AI Network Fabrics

Let's begin with a definition



"Artificial intelligence (AI) is the intelligence of machines or software, as opposed to the intelligence of humans or animals" https://en.wikipedia.org/wiki/Artificial_intelligence

"Artificial Intelligence (AI) refers to the simulation of human intelligence in machines, allowing them to perform tasks that typically require human intelligence" ChatGPT3

Machine Learning Complexity



Deep Learning Complexity



Agenda

- Al Basics
- Deep Learning
- Networking requirements
- Architectures for AI Network Fabrics

Al pipeline with Deep Learning



Back to Maths... Derivate and Gradient Descent

Derivative of a function



In neural networks:

The **backward-pass** is computing and updating the weights with the gradient descent method



Gradient Descent - Find the parameters

The previous graph was just a function of loss for one parameter

 Introduce just one more parameter you get a 3dimensional graph like this, on which you apply the gradient descent algorithms to find the local minimums

Real world models use billions or even trillion of input data points using 1000+ GPUs

- ChatGPT3 uses 96 layers and is trained on input dataset of 175 billion parameters
- ChatGTP4 uses a trillion parameters latest version
- Llama2 2 trillion parameters





Why does the BGP marker field usually starts with FFFFF...

\$

ΡE

The BGP (Border Gateway Protocol) marker field does not usually start with "FFFFFF." Instead, the marker field in BGP messages is a 16-byte field that can contain any value. The marker field serves as a synchronization mechanism and padding in BGP messages. It is used to ensure that both peers in a BGP session are in sync and that messages are correctly aligned.

Here's how the BGP marker field works:

- 1. **Synchronization**: When a BGP session is established between two routers, both routers exchange Open messages to establish the connection. The marker field is included in these messages. By comparing the marker field in the received Open message with the expected value, the receiving router can verify that both routers are synchronized before processing the rest of the message. If the marker fields do not match, it indicates a synchronization error, and the session is terminated.
- 2. **Padding**: The marker field also serves as padding in BGP messages. BGP messages are variable in length, and the marker field helps ensure that the rest of the message's fields are correctly aligned on 32-bit boundaries. This alignment is essential for efficient processing of BGP messages.

The "FFFFFF" pattern you mentioned may appear in some BGP messages, but it is not a requirement. The marker field can contain any 16-byte value, and the choice of value is up to the implementation. The "FFFFFF" pattern might be used for simplicity or as a default value in some implementations, but it is not a standard requirement of the protocol.

In summary, the BGP marker field is used for synchronization and padding purposes

S Regenerate

enecific value it contains can vary between implementations and is not mandated to start

This is a sample of the outcome from good training...

Сb

₿ 🏠

Default (GPT-3.5)

Djurgården. These teams are associated with football (soccer) in Sweden. Bajen is a common nickname for Hammarby IF, and Djurgården refers to Djurgårdens IF. Matches between these two Stockholm-based teams are often intense and have a strong rivalry.

If you have more specific information or a question related to a recent match or event involving these teams, please provide additional details, and I'd be happy to assist further.

bajen beat djurgarden

\$

PE

It sounds like you're indicating that Hammarby IF (often referred to as "Bajen") defeated Djurgårdens IF in a recent football (soccer) match. If that's the case, congratulations to Hammarby IF on their victory! Rivalry matches like these often carry a lot of significance for fans and can be quite exciting. If you have any specific details or questions about the match or related topics, feel free to ask, and I'll do my best to provide information or insights.

"The harder i practice, the luckier i get"

ſ٦

ሰ

 \Box

Neural Networking Operations

 AllReduce : Collect elements from all nodes, apply a reduction operator (e.g. sum), then distribute reduction to all nodes



AllGather : Collect elements from all nodes and distribute them to all other nodes



Many operational ways are used during model training

• ALLReduce, ALLGather, All-To-All...

How to distribute the gradient descent calculation?



Data parallelism allows feeding different GPUs with different parts of the data and process the data in parallel

• After each GPU processed its data, it shares the result with all the other GPUs

That thing with "GPU" vs. CPU and AI Workloads



4-16 cores	100s or 1000s of cores
Great for Serial Processing	Great for Parallel Processing
Process Interactive tasks	Break Jobs in separate tasks to process them simultaneously

Agenda

- Al Basics
- Deep Learning
- Networking requirements
- Architectures for AI Network Fabrics

Now to Networking for AI: What do you need ?

A fast, lossless network

- Graceful handling of large/bursty synchronized flows
- Fast and reliable transfer from host to network (RDMA)

A network with consistent latency

 Tail latency is likely to impact job completion time significantly

A network without collision

 Distribute equally low-entropy flows along all physical paths

Visibility and telemetry

• To identify bottlenecks in the network or application



RDMA (Remote Direct Memory Access)

- Most AI network traffic is GPU-to-GPU
- Normally network packets are written to kernel buffers and then copied by the CPU to application's buffers
- The CPU becomes a major bottleneck
- RDMA allows applications to directly read or write to another system's GPU memory, avoiding the OS network stack and any CPU processing



RDMA uses Queue Pairs (QPs) that are mapped to user-space memory regions. The NIC reads/writes directly to/from these regions.

RDMA Over Converged Ethernet (RoCE)

- Network protocol that allows RDMA over an Ethernet network
- The second version (RoCEv2) enhances the protocol with UDP/IP header
 - Operations on routed ethernet networks: default setup large datacenters
 - IP QoS : DSCP or alternatively COS/VLAN PRI
 - IP congestion control : the Explicit Congestion Notification (ECN) signal



Flow collision and traffic polarization

- Load balancing in IP routing based on "ECMP" 5-tuple etc...
 - Basically it is a Hash of fields in packet header
- Sadly AI clusters don't drive a significant distribution of parameters
 - Low level of entropy





Large flows could be polarized on the same links ...

How to avoid collision / traffic polarization ?

ECMP hashing : limited efficiency, especially with less entropy



• Dynamic Load Balancing (DLB): Smart flow distribution based on link utilization



•

- ECMP optimization
- Flows are allocated to new links based on current utilization, significantly increasing hash performance/efficiency
- Continuous reevaluation of best links with flows rebalancing

LBN - Load balancing "stitching" for AI workloads



Load Balancing Number (LBN) approach in brief GPUx <> GPUx, GPUy <> GPUy etc...

- Load Balancing Number assigned on each ingress interface to be the main input for hashing
- All traffic arriving on an ingress interface is effectively mapped to an egress interface between TOR & Spine

Next challenge: Incast

Many-to-1 type of traffic

- Typical in HPC environment
- Very aggressive in AI/ML environment (much more flows)
- You can't be saved by TCP, since the typical AI flow UDP based

Consequences

- Slow&Drops
- Worst case failed training session
- Highly spend \$ for nothing





Incast problem: Mitigation

- Add notification mechanism to signal pressure along path and reduce quantity of traffic at the source, Pause frames (L2) and ECN (L3)
- Deep buffering in order to reduce the networking pressure
 - Could increase the latency because you will have to "drain" the buffer



The lost last bits of the DSCP header = ECN



ECN - Operations



 ECN marking is happening in egress and start when the average queue size is above a configurable threshold



Average Queue Fill level

What can buffers solve ?

"RDMA" Incast

- Incast is a many-to-one problem
- A single host request can generate simultaneous responses from many nodes creating a microburst that overwhelms a single port



Speed Mismatch

- Mix of different ports speeds is common
- Speed mismatch can lead to buffer exhaustion, especially at the leaf layer



Same old truth... buffering capability always good !

Agenda

- Al Basics
- Deep Learning
- Networking requirements
- Architectures for AI Network Fabrics

AI Fabric



AI Fabric

Key Variables

Total # of AI NIC ports NIC SerDes Speed

Rack physical layout and fiber plant

Cost





Large AI applications (1000s of xPUs)

Single-Tiered Pizza Box



Small AI applications Up to 64 xPUs at 400Gbps



64 port 400G QSFP-DD

32 port 800G* (2x400G) OSFP or QSFP-DD

- Fixed Configuration Switch
 - 64x 400G
 - 32x 800G
- No flow collisions
 - Single-asic line-rate forwarding
- ECN and/or PFC to handle incasts
 - If Low buffers Requires tuning

Single-Tiered Chassi



Moderate AI applications Up to 576 xPUs at 400Gbps



- Modular chassis offering high port density
- No flow collisions between line card and fabric
 - · Cell-based Fabric
 - Built in overprovisioning between fabric and linecard
 - 100% Fair and Efficient Load Balancing within the chassis
- High Availability
 - Fabric, fan, power supply etc... redundancy
- ECN and/or PFC to handle incasts
- Deep buffers Requires minimal tuning

Questions/Corrections/Bashing ?