Distributed matching with local information

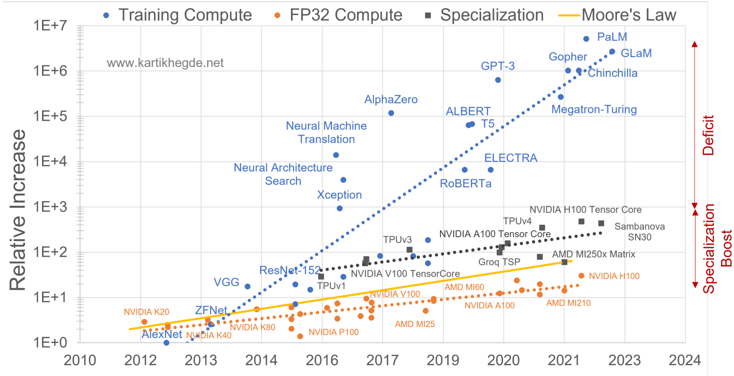
Moonmoon Mohanty, Gautham Bolar, Preetam Patil, Parimal Parag
Ayalvadi J. Ganesh, Jean-Francois Chamberland
Google Indian Networking Research Summit, Jan 05, 2026

Anusandhan National Research Foundation
CENTRE FOR NETWORKED INTELLIGENCE Indian Institute of Science
IBM
Qualcomm
intel

The growing performance deficit

Deep learning compute demand[1]

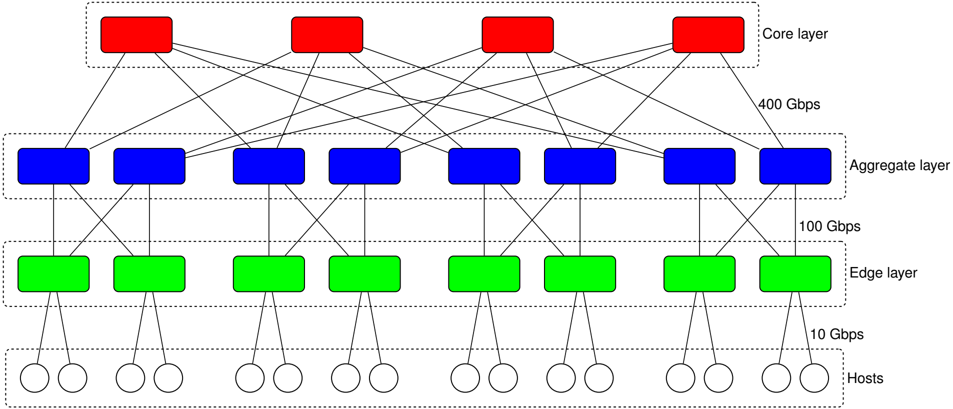


Data collected by Kartik Hegde (kvhegde2@illinois.edu). Training FLOPS for transformers is based on Narayanan et al. "Efficient large-scale language model training on gpu clusters using megatron-lm". In SC'22, for others it is calculated as `FLOPS/forward pass* #dataset * #epochs * 3.`

- End of Moore's law and Dennard scaling
- Can no longer keep adding more transistors and increasing core frequencies
- We're accustomed to increasing compute achievable only by scaling out through data center network (DCN)

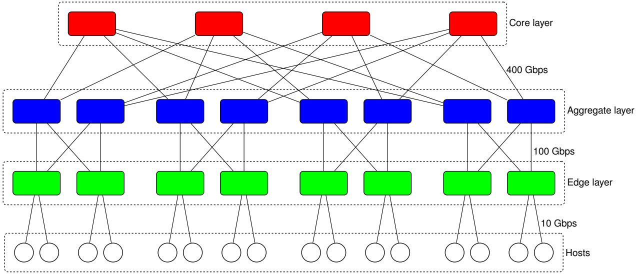
[1] Image credit: https://kartikhegde.substack.com/p/accelerating-deep-learning-in-the

DCN for compute



- DCN traffic comprises of
  - Latency sensitive compute applications (short messages)
  - Bandwidth hungry applications (long messages)
- The link from host to edge layer switch is the bottleneck
- End-hosts can be senders and receivers

Why not use TCP for networked compute?



Inadequacies of TCP for networked compute

- Not all messages are equal (TCP treats them equally)
- TCP congestion control operates at RTT (scale of microseconds in DCN)
  - Too high a latency for networked compute
  - High DCN bandwidth leads to large queue buildup and latency

Transport protocols for networked compute

- Scheduling algorithms for input-queued switches in high-speed LANs and WAN routers
  - Parallel iterative matching (PIM)[2]
  - Iterative round robin scheduling algorithm (iSLIP),[3] an improvement over PIM
- Transport protocols for networked compute
  - Priority based receiver driven Homa[4] improvement over pHost, NDP
  - Data center parallel iterative matching (dcPIM)[5]

---

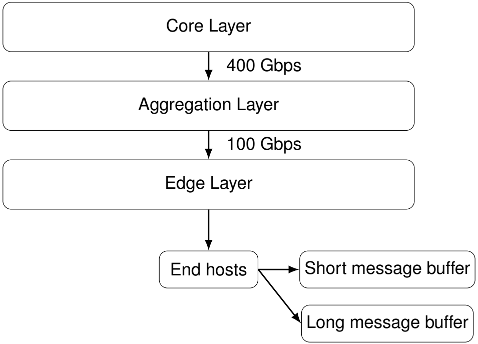
[2] Thomas E Anderson, Susan S Owicki, James B Saxe, and Charles P Thacker. High-speed switch scheduling for local-area networks. *ACM Transactions on Computer Systems, 11(4):319–352, Nov. 1993.*

[3] Nick McKeown. The iSLIP Scheduling Algorithm for Input-queued Switches. *IEEE/ACM Transactions on Networking, 7(2):188–201, Apr. 1999.*

[4] Behnam Montazeri, Yilong Li, Mohammad Alizadeh, and John Ousterhout. 2018. Homa: A Receiver-driven Low-latency Transport Protocol Using Network Priorities. *ACM SIGCOMM 2018.*

[5] Qizhe Cai, Mina Tahmasbi Arashloo, and Rachit Agarwal. dcPIM: Near-optimal proactive datacenter transport. *ACM SIGCOMM 2022.*

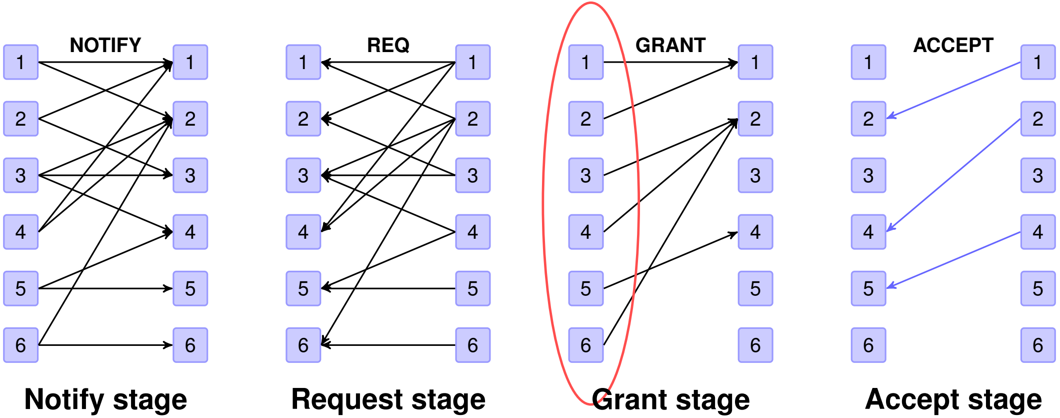
Transport protocol for networked compute



Proposals for transport protocols for compute

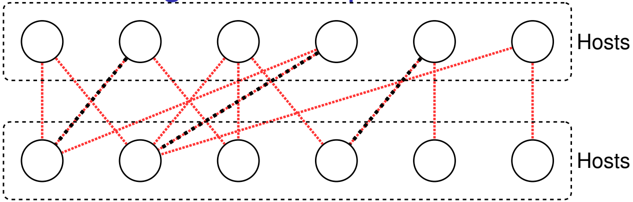
- Distinguish among control, short compute messages, and long persistent messages
- Short messages are prioritized and no congestion control for them
- Congestion control for long messages via receiver grant mechanism
- Matching of sender receiver pairs for efficient utilization of bottleneck edge links

Control message exchange during matching phase



- Bipartite matching sender receiver pairs is at the heart of the problem
- Each round consists of matching and data transmission phases
- Distributed matching with local information to keep the matching phase small

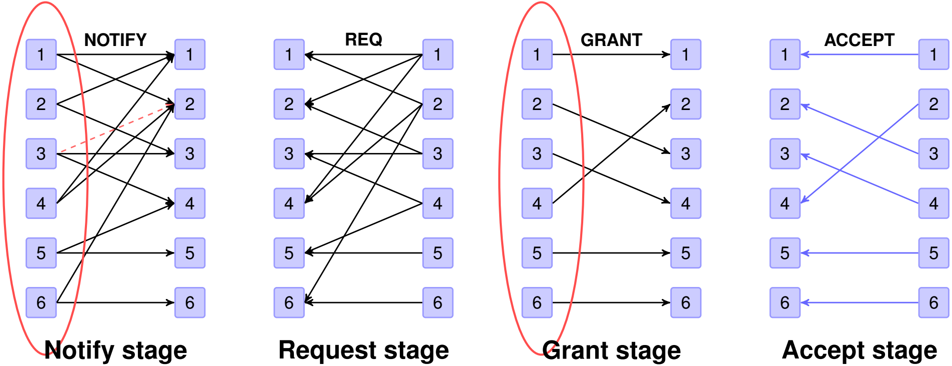
Matching algorithms using Grant requests



Related works

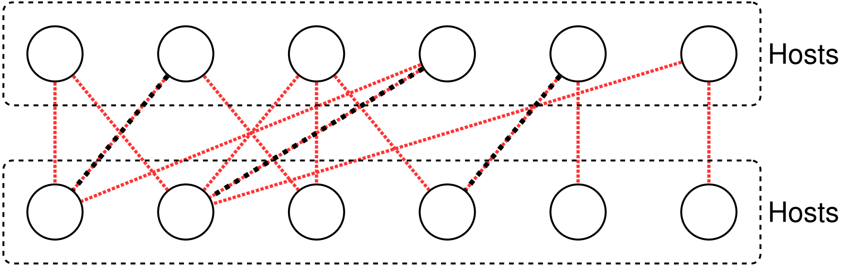
- Centralized bipartite matching
  - Hopcroft-Karp: computes maximum matchings in polynomial time
  - Hungarian: polynomial-time algorithm widely used in applications like client selection or resource allocation to maximize total weight
- Distributed matching
  - Random bipartite matching
    - Uses one neighborhood information
    - Picks one of its intended receivers uniformly at random
  - Proposal: Degree-Biased (DB) (α) random bipartite matching
    - Uses two neighborhood information
    - Biased selection of one of its receivers

DB(α) algorithm for notification and grant selection



- Random thinning of one neighborhood in the notify stage
- Uses two neighborhood information built during the request stage
- Biased selection of one of its receivers in the neighborhood in grant stage
- Selection probability of a receiver proportional to α exponent of its neighborhood size

Analysis for random graphs

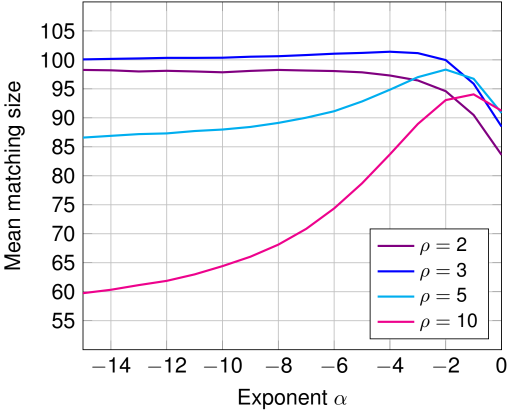


Erdős-Rényi connection

- Each sender-receiver pair is likely to be present with probability ρ/N independently
- Mean number of connections for each sender and receiver is ρ
- Closed form expression for mean matching when $\alpha \in \{0, -\infty\}$ and large N

Performance without thinning

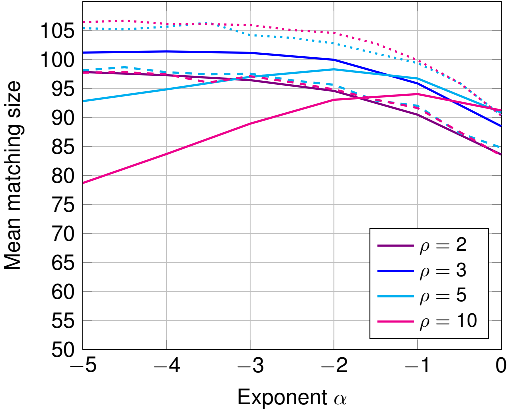
Impact of α exponent selection



- Non-trivial exponent $\alpha < 0$ maximizes the mean matching size
- Optimal exponent depends on the graph density ρ
- Biasing receiver selection in favor of lower degree ($\alpha = -\infty$) is close to optimal for sparse graphs

Performance with thinning

Impact of thinning (dotted lines)



- Independent thinning of each edge equivalent to sparser graphs
- Restriction on sender neighborhood size leads to better mean matching size
- Biasing receiver selection in favor of lower degree ($\alpha = -\infty$) optimal for sparse graphs with thinning

Conclusion

Feature of the proposed distributed matching algorithm DB(α)

- Based on two neighborhood information
- Theoretical analysis for $\alpha = -\infty$ and Erdős-Rényi random graphs in both sparse and dense regimes
- Improves the mean matching size and hence throughput for long messages
- Reduces control message overheads, improving throughput for short messages as well

Reference

M. Mohanty, G. Bolar, P. Patil, A. J. Ganesh, J. F. Chamberland, P. Parag, “Bipartite matching under communication constraints”, *arXiv:2604.10744*, Jun 2026.