A Simulation Study of Hardware Parameters for GPU-based HPC Platforms

Saptarshi Bhowmik[1], Nikhil Jain[2], Xin Yuan[1] and Abhinav Bhatele[3]

1. Florida State University, 2. NVIDIA, Inc, 3. University of Maryland

Goals

- **Study impact of (1) interconnect link bandwidth, (2) number of GPUs per node, and (3) interconnect topology in application performance through discrete-event simulations in the context of two most popular network topology Fat-Tree[1] and 1D-Dragonfly[2][4].**

Introduction

GPUs are increasingly used in High Performance Computing (HPC) platforms, resulting in the increase in per-node computational capacity and the decrease in the number of endpoints in the system. As such, it is imperative that computation and communication in the system remains balanced.Hardware architectural parameters such as the link bandwidth and the number of GPUs per node are crucial design parameters that determines this balance and thus, the overall performance of the system.In this research, we leverage the whole system simulation capability of TraceR-CODES [3] and use it to study the impact of hardware parameters using HPC workloads.

Methods

Tracer-CODES

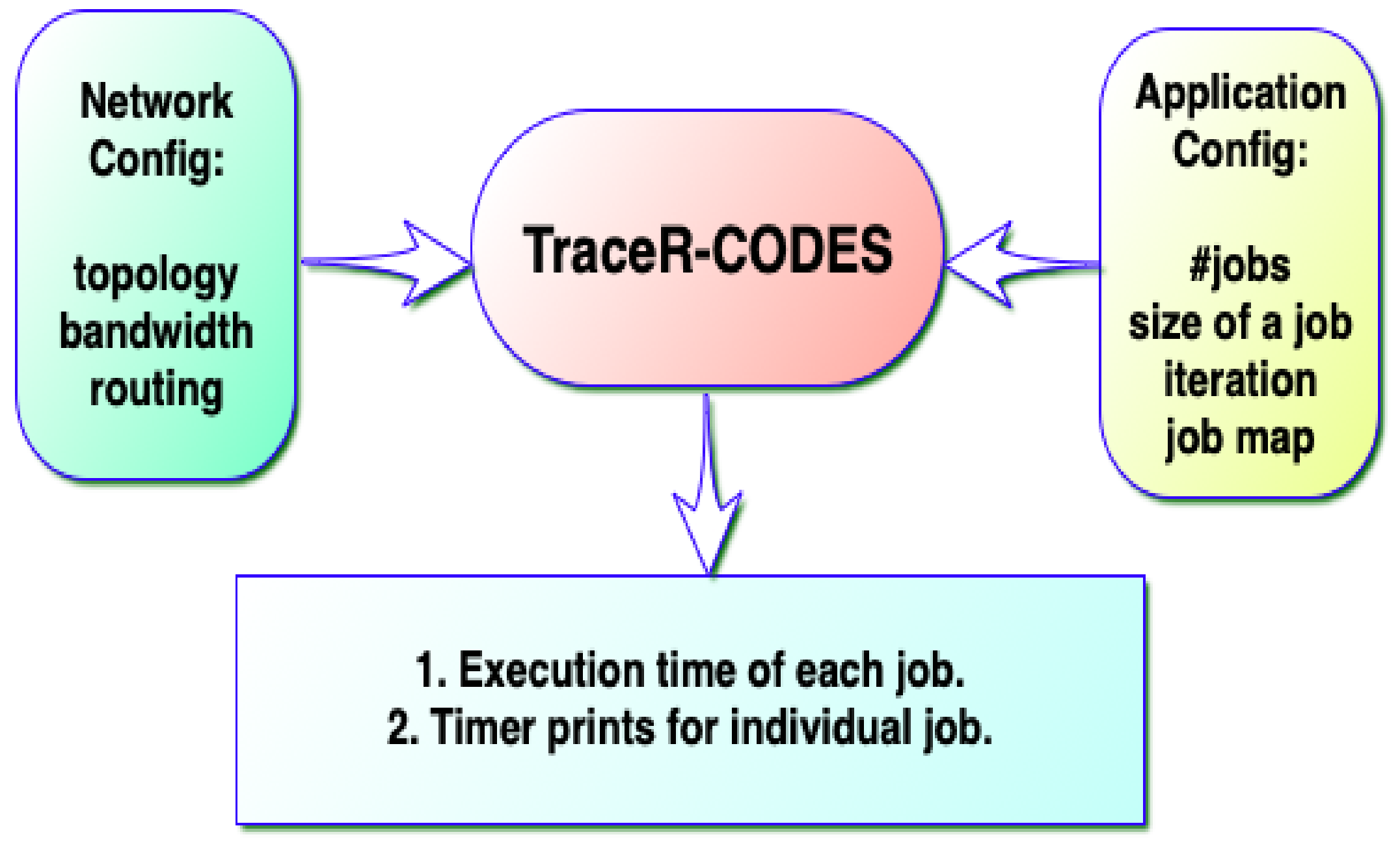

Figure: TraceR-CODES workflow

Applications

Six popular HPC Applications are used in the simulation.

Traces	Computation Intensive	Communication Intensive
Stencil4d	✗	✓
Subcomm3d	✗	✓
Kripke	✓	✗
Laghos	✓	✗
Amg	✓	✓
Sw4lite	✓	✓

Table: Profiles of Application Traces

Simulation Environment Parameters

Network Topology

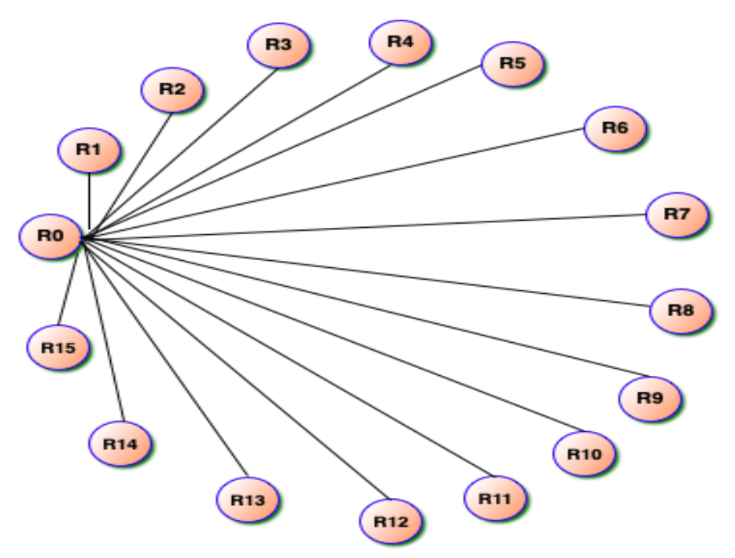

Figure: A 1D-Dragonfly Group, (Intra Group links of router R0)

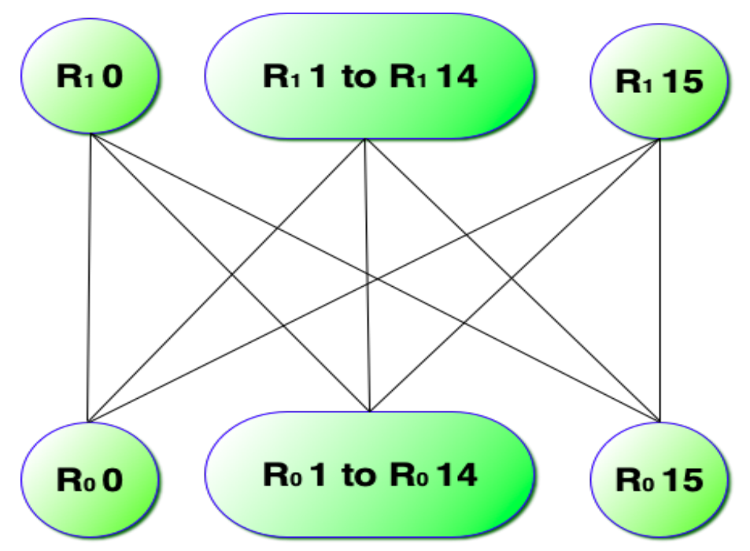

Figure: A Fat-Tree Pod

Network Sizes and GPUs per node

GPUs per node	1D-Dragonfly	Fat-Tree
1 GPU/Node	16 Groups	8 Pods
2 GPU/Node	8 Groups	4 Pods
4 GPU/Node	4 Groups	2 Pods
8 GPU/Node	2 Groups	1 Pods

Table: Network sizes for each different GPUs per node, default size is of 1 GPU per node

Bandwidth

Default Setting :

Link Bandwidth = 11.9 GB/sec
Internal Bandwidth = 23.8 GB/sec

We use 8 more bandwidth, **x/16, x/8, x/4, x/2, 2x, 4x, 8x, 16x** for our simulations.

Results

Impact of GPUs per node

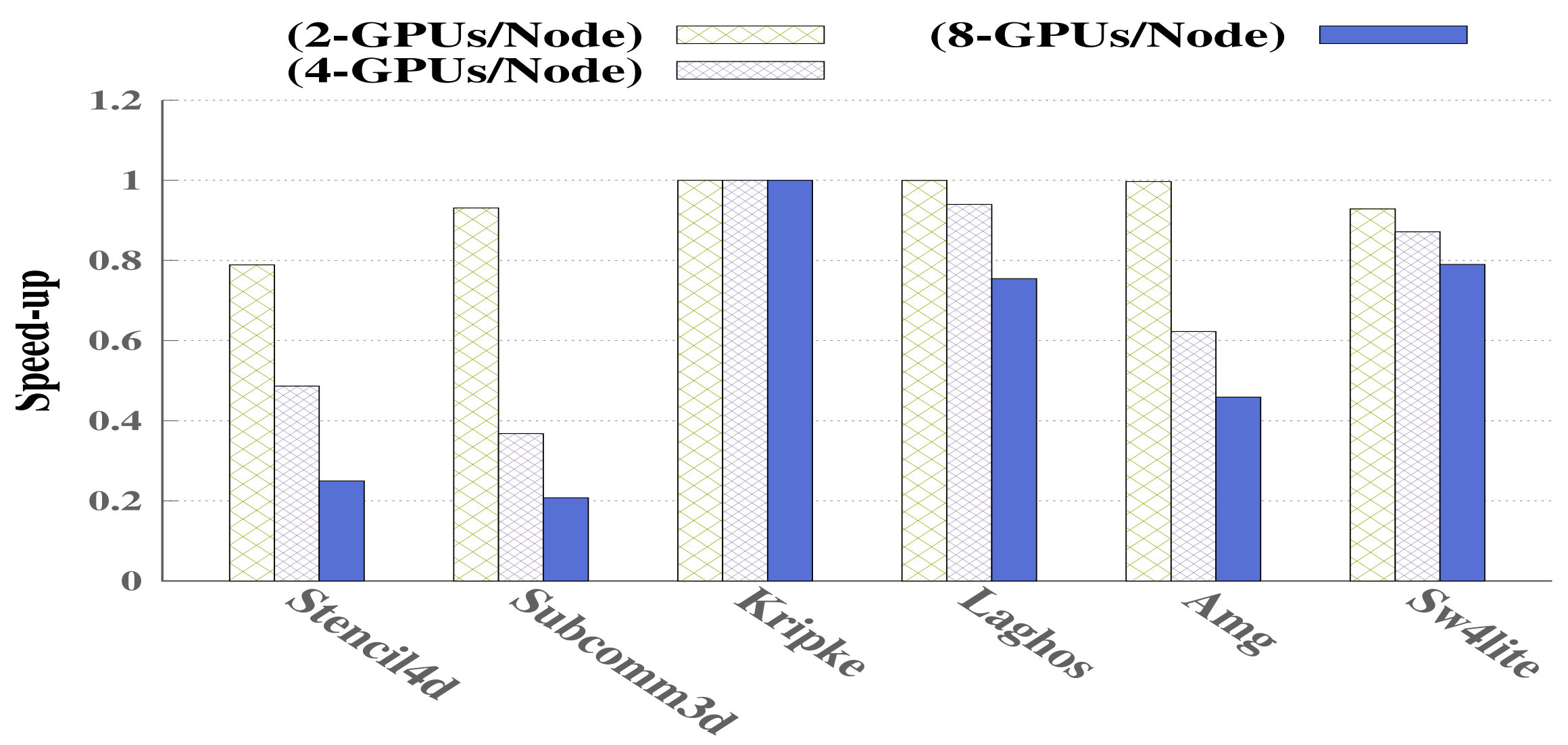

Figure: Speed-up over default setting for different GPUs per node mapping, for all Applications of 128 ranks, in Fat-Tree

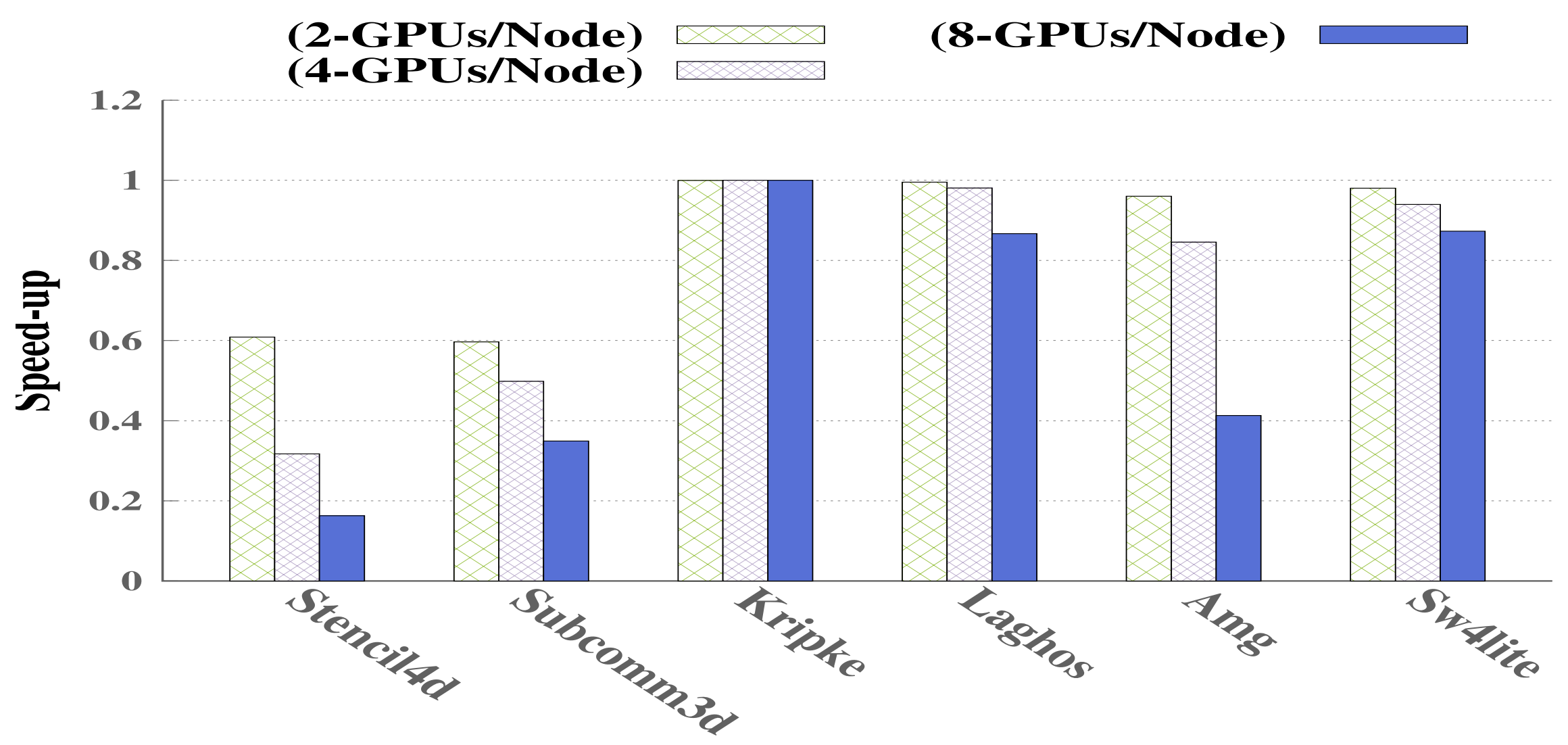

Figure: Speed-up over default setting for different GPUs per node mapping, for all Applications of 128 ranks, in 1D-Dragonfly

The performance of communication kernels Stencil4d and Subcomm3d drops significantly while the performance of computational intensive kernels Kripke and Laghos remains similar, for both 1D-Dragonfly and Fat-Tree topology.

Impact of Bandwidth

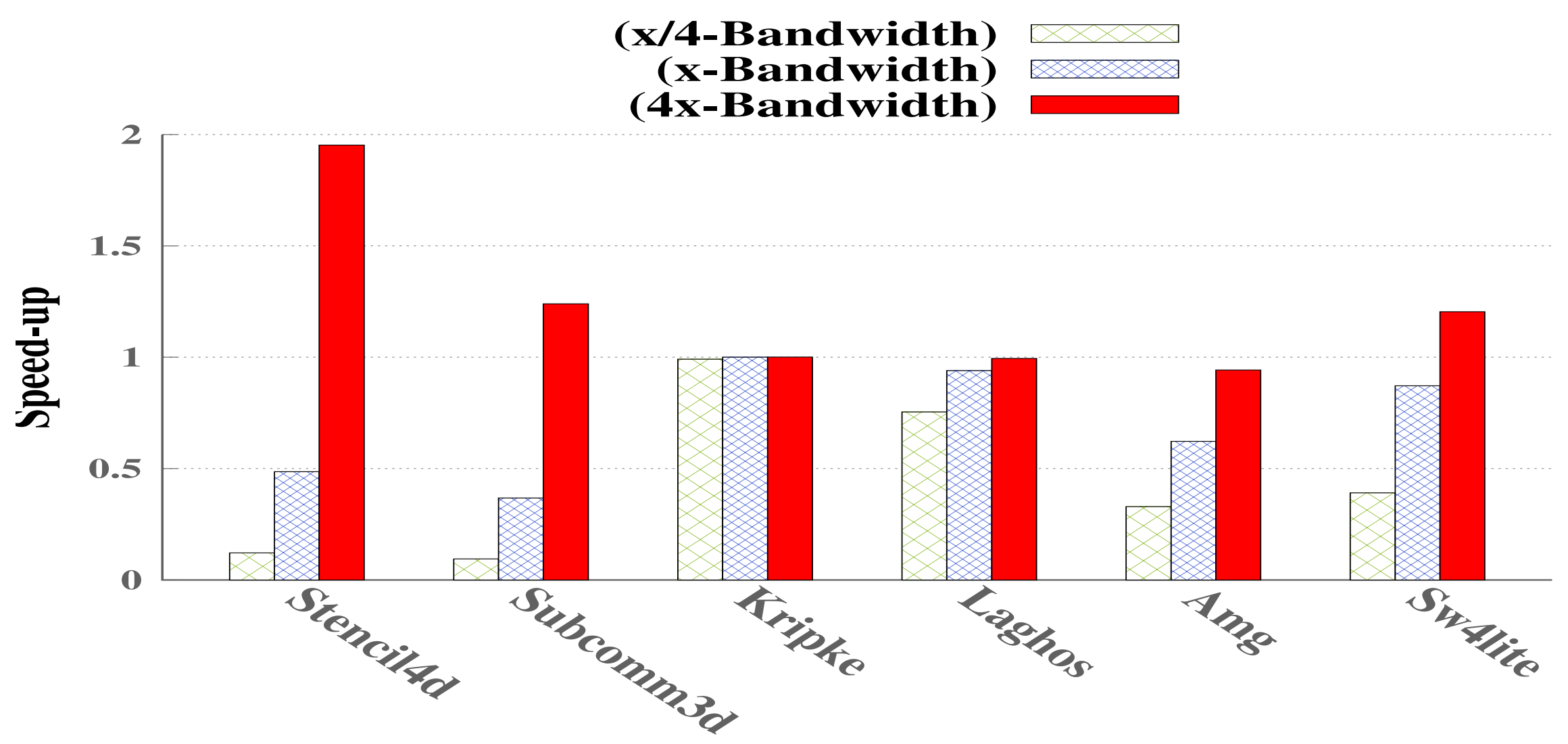

Figure: Speed-up over default setting for various bandwidths in 4 GPUs per node mapping, for all Applications of 128 ranks, in Fat-Tree

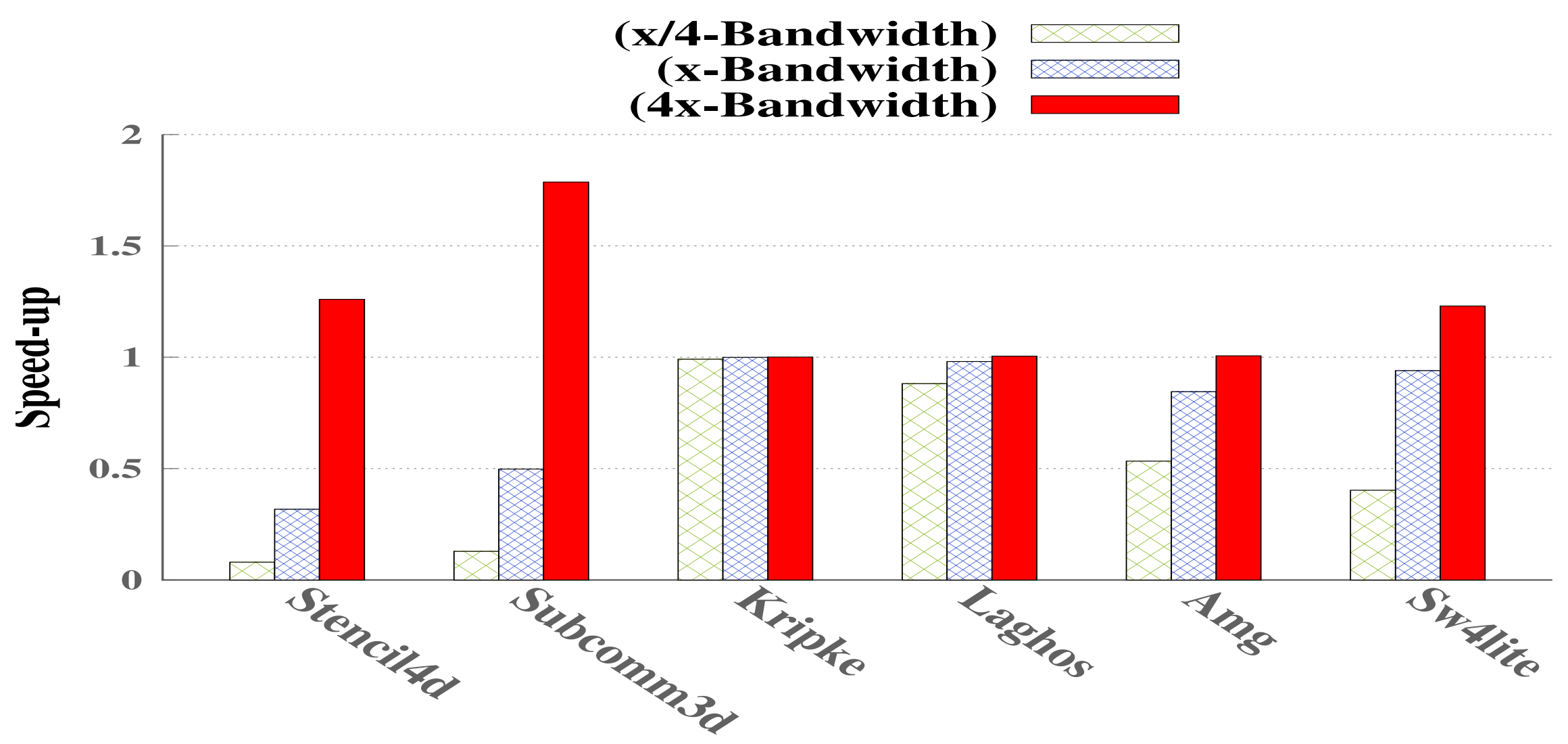

Figure: Speed-up over default setting for various bandwidths in 4 GPUs per node mapping, for all Applications of 128 ranks, in 1D-Dragonfly

For the applications that are sensitive to communication Stencil4D, Subcomm3d, Amg, and Sw4lite, as the number of GPUs per node increases, more link bandwidth is needed to sustain the performance -- insufficient bandwidth.

Conclusion

- As the number of GPU per node increases, the node becomes more computation intensive, and thus, there is a slowdown in application performance as the communication/computation ratio of the network reduces.
- As the number of GPU per node increases, more bandwidth is needed to substantiate the slowdown in application performance.
- Every application has a "sweet spot" where it is performing the best. This indicates that substantial benchmarking study will be needed to determine the best system configurations for the GPU-based systems.

Future Works

- Considering other interconnect choices, like Jellyfish and Edison Dragonfly
- Using more benchmark applications,
- Studying other system parameters such a NIC-level packet scheduling and buffer size.

References

[1] Jain, N., Bhatele, A., Howell, L. H., Böhme, D., Karlin, I., León, E. A., ... Leininger, M. L. (2017, November). Predicting the performance impact of different fat-tree configurations. In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (pp. 1-13).

[2] Kim, J., Dally, W. J., Scott, S., Abts, D. (2008, June). Technology-driven, highly-scalable dragonfly topology. In 2008 International Symposium on Computer Architecture (pp. 77-88). IEEE.

[3] Acun, B., Jain, N., Bhatele, A., Mubarak, M., Carothers, C. D., Kale, L. V. (2015, August). Preliminary evaluation of a parallel trace replay tool for hpc network simulations. In European Conference on Parallel Processing (pp. 417-429). Springer, Cham.

[4] Alzaid, Z. S. A., Bhowmik, S., Yuan, X., Lang, M. (2020, June). Global link arrangement for practical Dragonfly. In Proceedings of the 34th ACM International Conference on Supercomputing (pp. 1-11). Magnetics Japan, p. 301, 1982].