Analyzing Hardware Parameters in GPU based HPC Platform

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

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

Goals

- Use discrete-event simulations to study various hardware design parameters and their impact on the performance of HPC workloads
- Study several parameters including nework bandwidth, number of GPUs per node in the context of two most popular network topologies Fat-Tree and 1d-Dragonfly

Introduction

Today's GPU networks are largely dependant on communication performance of applications over the network. [1]. As such the design and environment effects of performance on application is very pronounced and needs to be studied in general.

Method

Define the desired network
We use two network topology that is implemented in TraceR-CODES simulator, 1d - Dragonfly and Fat-Tree with Adaptive Routing.
We use the Number of GPU's per node from 1-GPU/Node, 2-GPU/Node, 4-GPU/Node and 8-GPU/Node
We also use different Link bandwidths which is the ratio of base bandwidth X (10 Gbps), from x/8, x/4, x/2, x, 2x, 4x, 8x.

Traces	Computation	Communication
Stencil4d	✗	✔
Kripke	✔	✗
Laghos	✔	✗
Subcomm-a2a	✗	✔
Sw4lite	✔	✔
Amg	✔	✔

Table: Application Traces

Experimental Setup

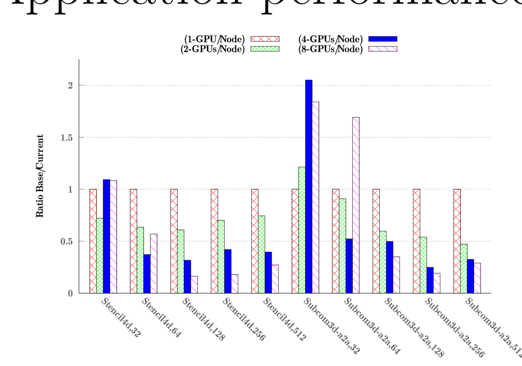
We are running 20 Workloads of randomly selected jobs from the above application from ranks 32, 64, 128, 256 and 512. We make sure that each application rank appears atleast 4 times across all workloads.

Jobs	32-Ranks	64-Ranks	128-Ranks	256-Ranks	512-Ranks
Kripke	4	4	4	4	5
Laghos	6	4	4	4	4
Subcomm-A2A	4	6	7	10	4
Stencil4D	4	4	8	8	4
Sw4lite	8	5	4	8	5
Amg	8	5	5	5	12

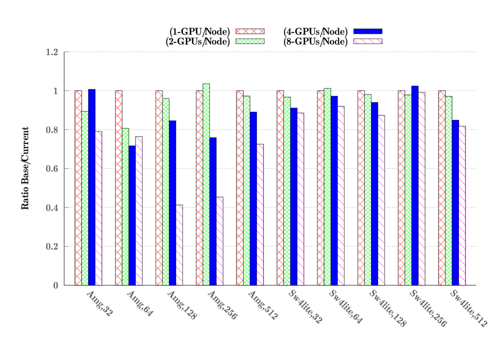
Table: Number of occurrences of each Application per rank

Results

First, Impact of Number of GPU's per node on Application performance.

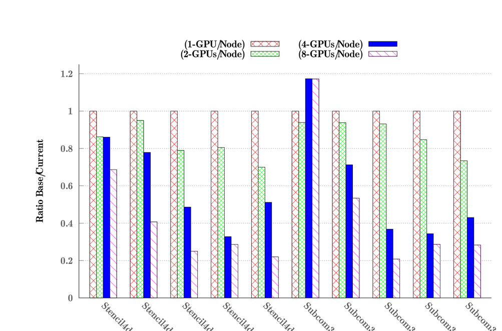


Communication Apps

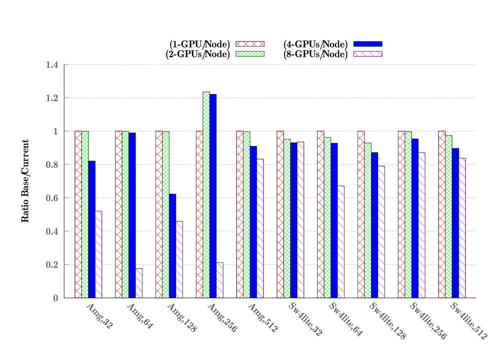


Amg, Sw4lite Proxy

1d-Dragonfly Topology



Communication Apps

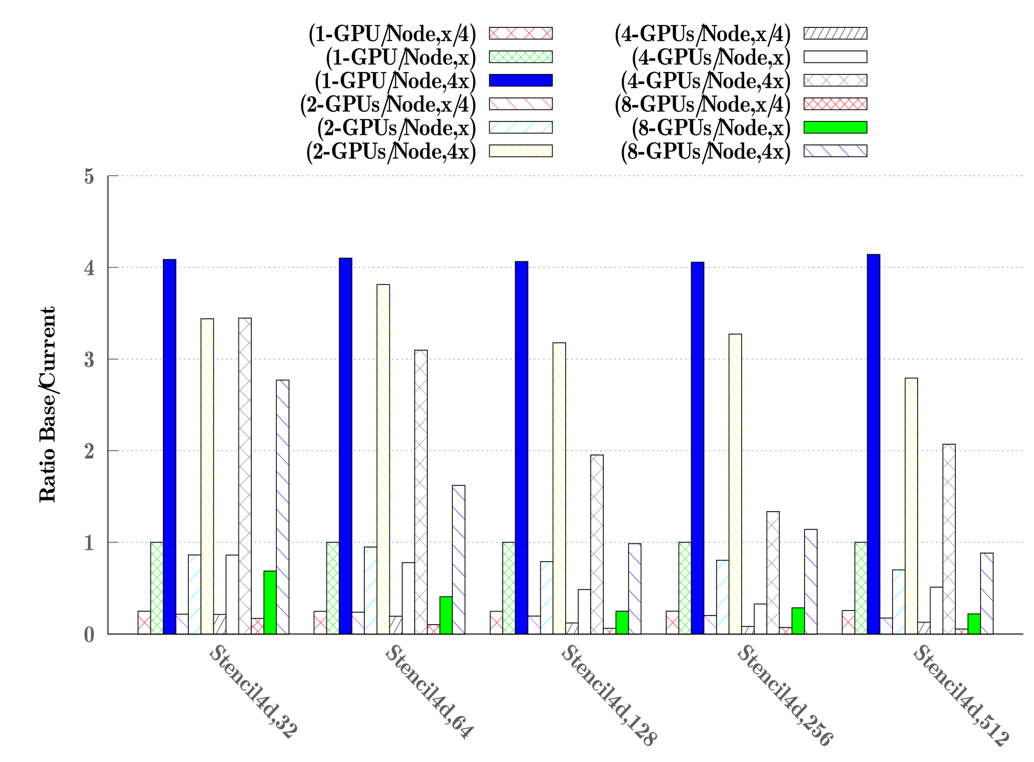


Amg, Sw4lite Proxy

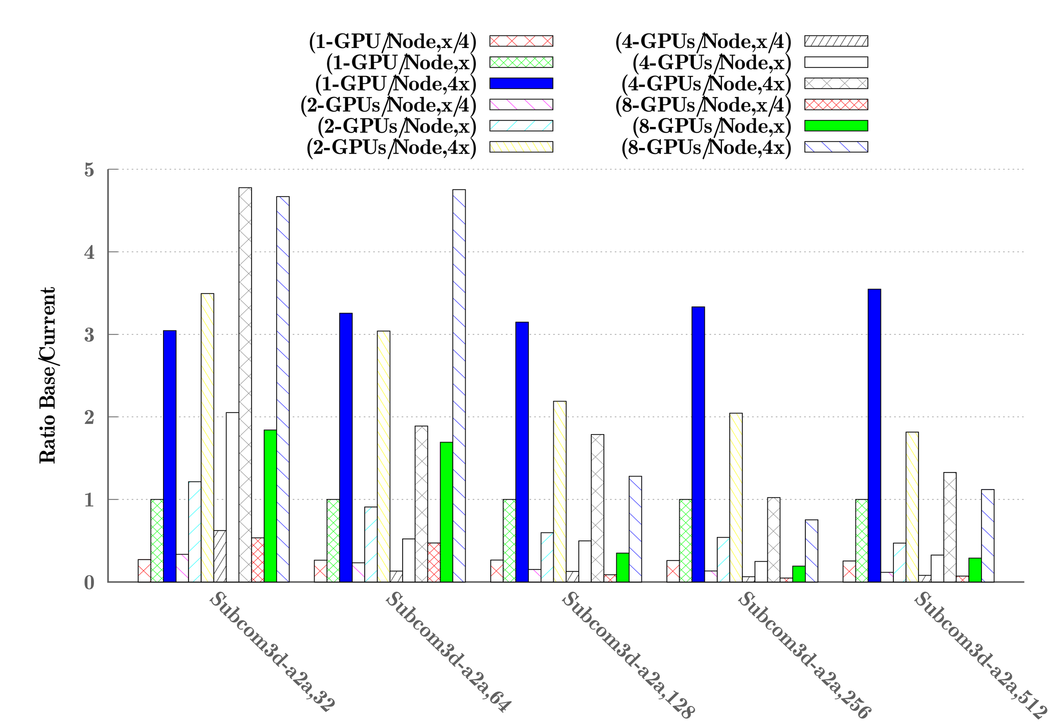
Fat-Tree Topology

Results

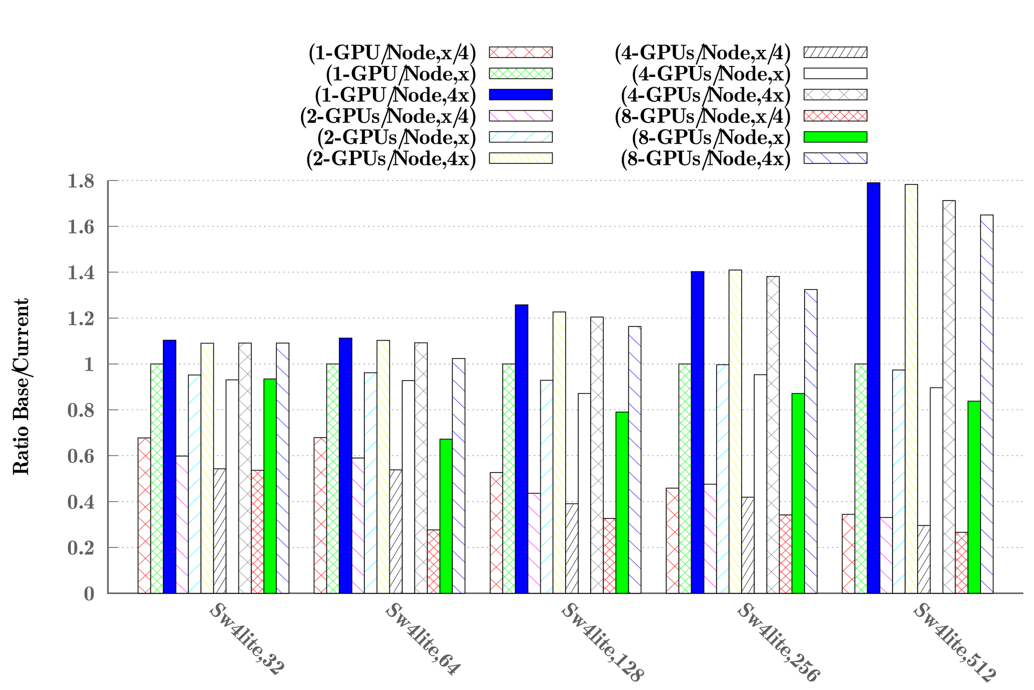
Second, Impact of Bandwidth and different GPU's per node on Application Performance.



Ratio of Bandwidth,GPUs per node mapping for Stencil in Fat-Tree



Ratio of Bandwidth,GPUs per node for Subcom3d-a2a in 1d-Dragonfly



Ratio of Bandwidth,GPUs per node mapping for Sw4lite in Fat-Tree

Conclusion

- **GPUs per node** The communication intensive applications slowdown when the number of GPUs per node is increased, among the proxy applications only AMG shows slowdown.
- **Link Bandwidth** As we increase the number of GPU's per node. More bandwidth is needed to make the performance at par with lower GPU's per node configuration.
- In Subcom3d-a2a applications, applications with fewer ranks are performing better as more GPU's are mapped to one node, as there is more intra-node communication.

Future Work

- Every application has a sweet spot where the performance is the best, figure out the sweet spot for other HPC applications.
- Try to study how other simulation environment and hardware design, such as NIC scheduling policies effect the performance of applications

References

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 (LLNL-POST-783752).