EUROfusion

Benchmarks and validation of the Pitagora HPC system

Serhiy Mochalskyy

3rd Annual meeting of EUROfusion HPC ACHs
November 25-26, 2025

Advanced Computing Hub Garching
Max-Planck-Institut für Plasmaphysik
Boltzmannstr. 2, D-85748 Garching, Germany

Pitagora CPU and GPU partitions

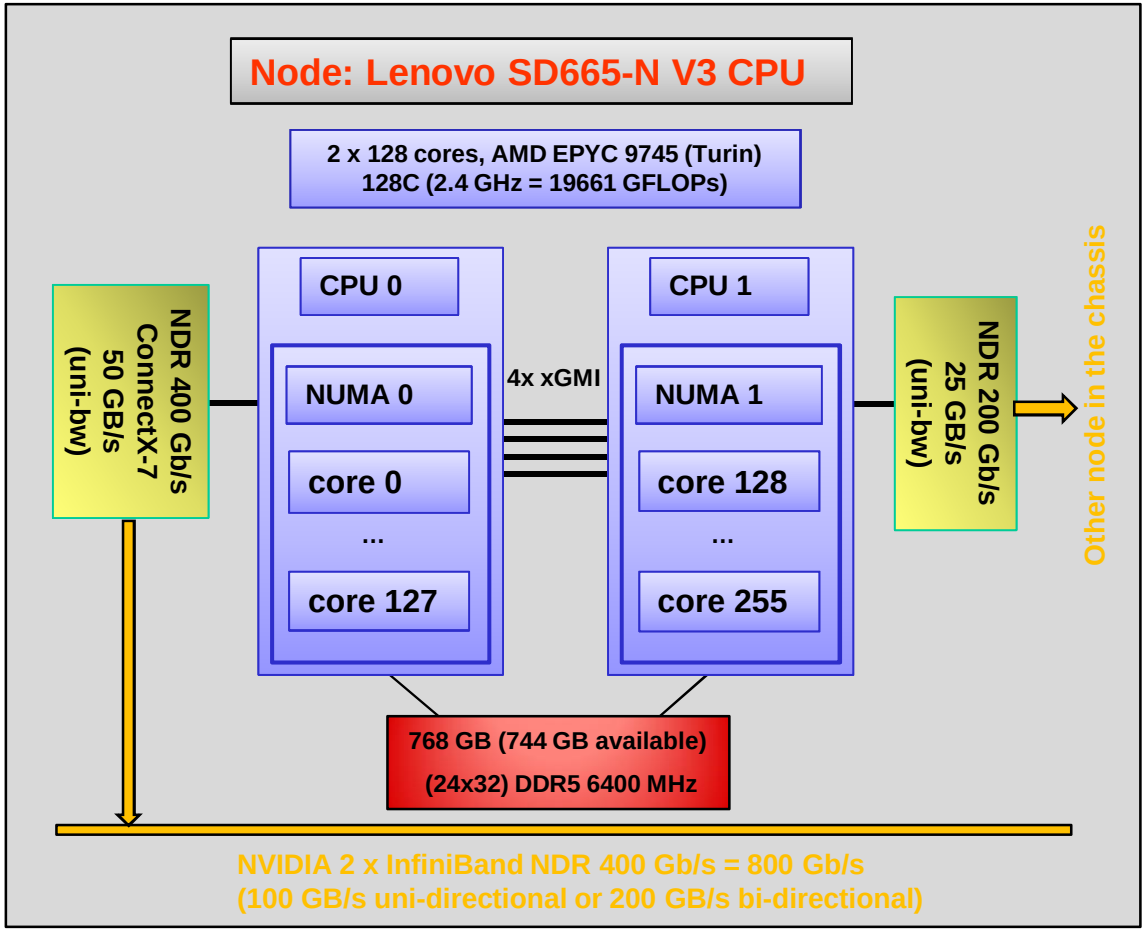
GPU

- 7 racks
- 37 PFlops (Rpeak, from top500: 58)
- 168 Compute nodes
- 2x Intel Emerald Rapids 32c
- 512 GB DDR5 6400 MT/s
- 4x NVIDIA H100 SXM 94GB HBM2e
- 2.3x performance over A100
- 4x NDR200 adapters (200 Gb/s each)

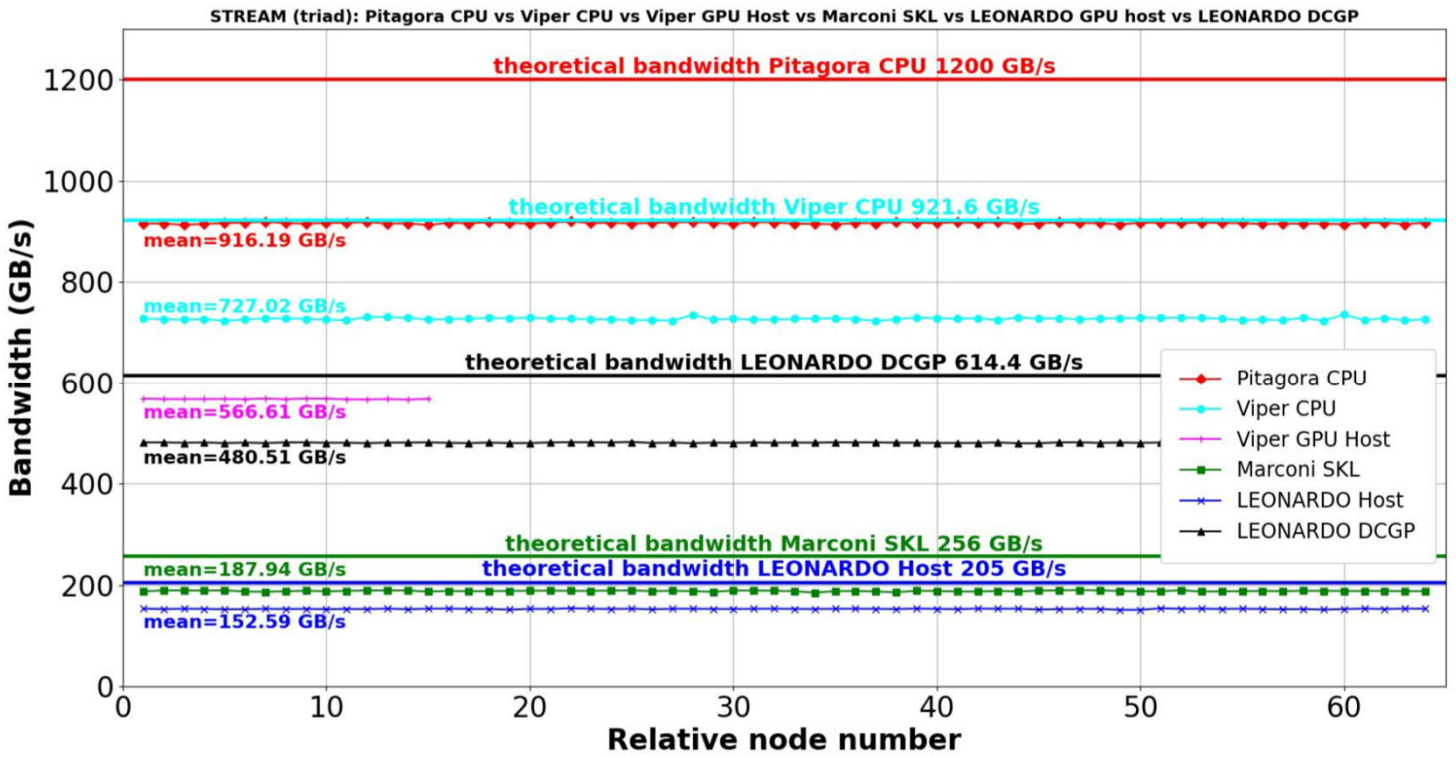
CPU

- 14 racks
- 20 PFlops (Rpeak, from top500: 104)
- 1008 Compute nodes
- 2x AMD Turin 128c (Zen5) 2.4 GHz
- 768 GB DDR5 6400 MT/s

Lenovo SD665-N V3 CPU

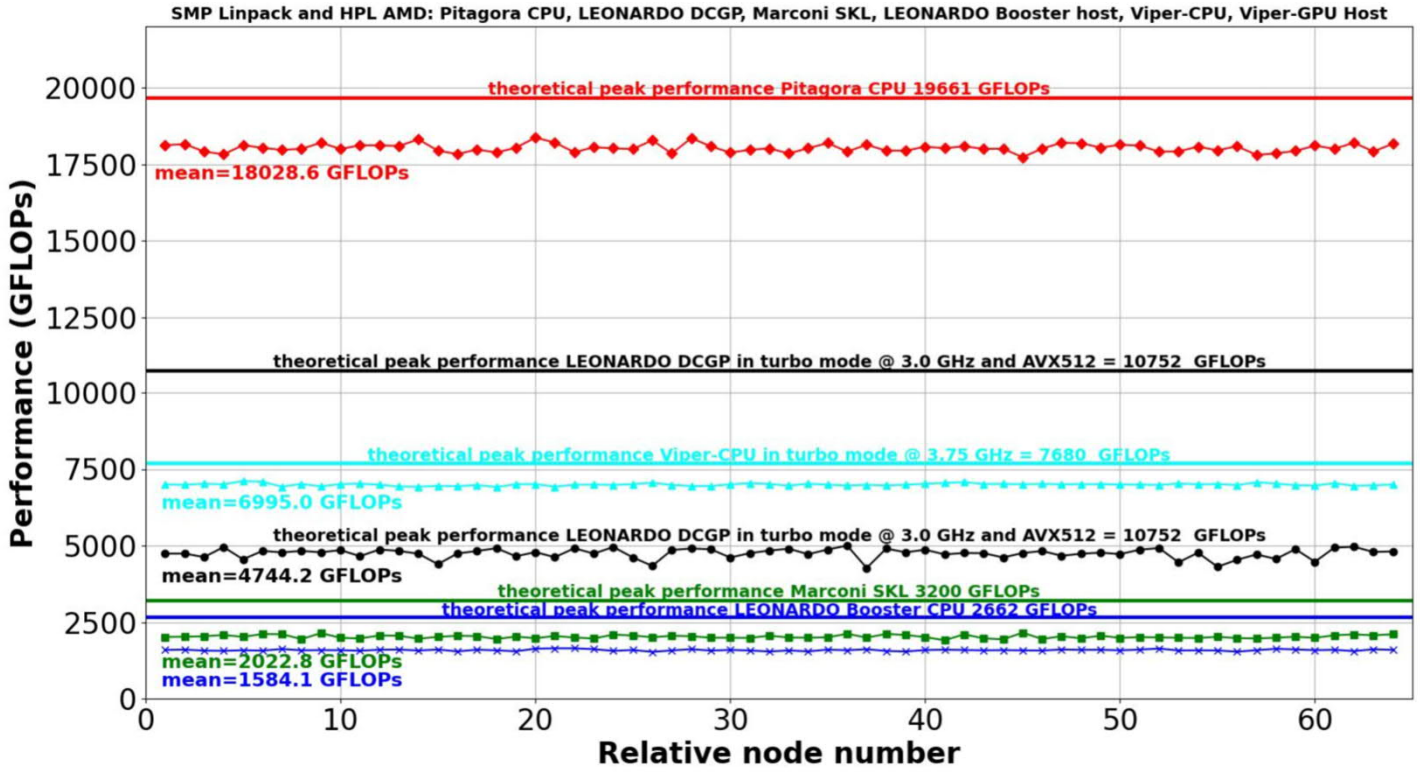


Stream benchmark on Pitagora CPU



- **Pitagora CPU (mean): 916 GB/s from 1200 GB/s theoretical value (76%).**
- **Viper-CPU (mean): 727 GB/s from 921.6 GB/s theoretical value (79%).**
- **Viper-GPU Host (mean): 567 GB/s.**
- **LEONARDO DCGP (mean): 481 GB/s from 614.4 GB/s theoretical value (78%).**
- **MARCONI SKL (mean): 188 GB/s from 255.94 GB/s theoretical value (73%).**
- **LEONARDO Booster Host (mean): 153 GB/s from 205 GB/s theoretical value (75%).**

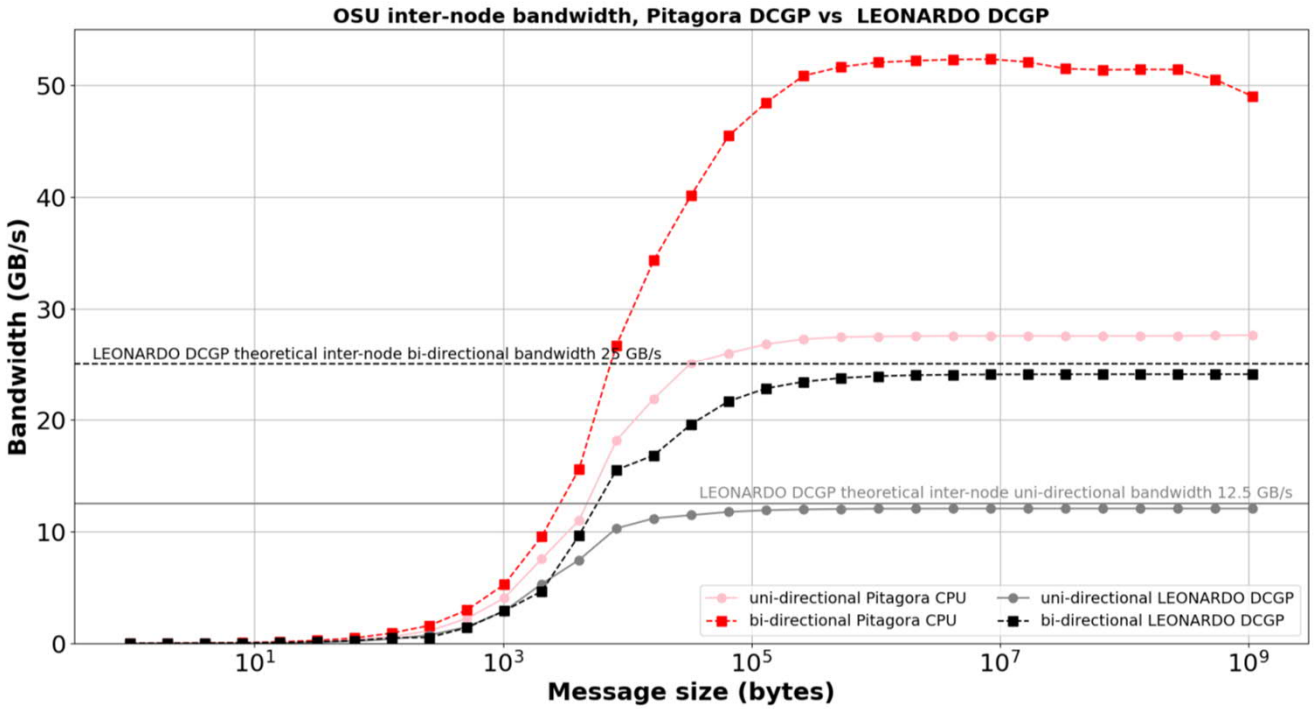
Pitagora CPU: performance stability test



- All nodes provide high, stable and symmetric performance close to the theoretical value.
- Pitagora CPU (mean): 18029 GFLOPs from 19661 GFLOPs theoretical value (92%).
- Viper-CPU (mean): 6995 GFLOPs from 7680 GFLOPs theoretical value (91%).
- LEONARDO DCGP (mean): 4744 GFLOPs from 10752 GFLOPs theoretical value (44%).
- MARCONI100 Host (mean): 2023 GFLOPs from 3200 GFLOPs theoretical value (63%).
- LEONARDO Booster Host (mean): 1584 GFLOPs from 2662 GFLOPs theoretical value (60%).

Pitagora-CPU inter-node network bandwidth

using osu_bw and osu_bibw benchmarks from OSU microbenchmark



- **Stable and high bandwidth for uni- and bi-directional data transfer.**
- **Pitagora-CPU: bi-directional bandwidth ~52 GB/s.**
- **Pitagora-CPU: uni-directional bandwidth ~27 GB/s.**
- **LEONARDO DCGP: bi-directional bandwidth ~24.2 GB/s from 25 GB/s of the theoretical value (97%).**
- **LEONARDO DCGP : uni-directional bandwidth ~12.1 GB/s from 12.5 GB/s of the theoretical value (99%).**

Key codes and their issues on Pitagora

GENE:
a) Compiled with GCC → hangs or UCX errors;
b) Compiled with Intel + verbs provider → sometimes produces **wrong results**;
c) Compiled with Intel + TCP provider → performance slower by a factor of two (and sometimes hangs).

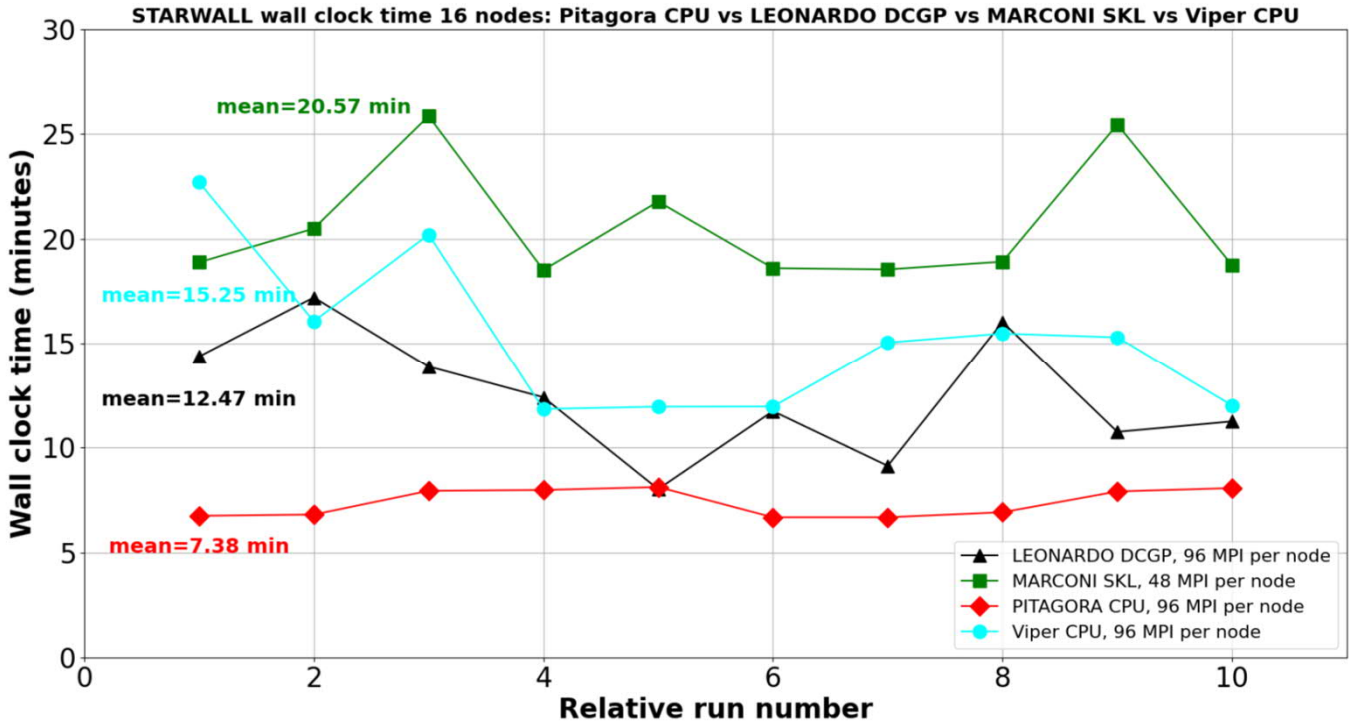
CINECA has assigned a dedicated **team of several engineers** to address all these issues.

Promising solution:
export FI_PROVIDER=mlx
export UCX_TLS=self,dc_mlx5

STARWALL performance

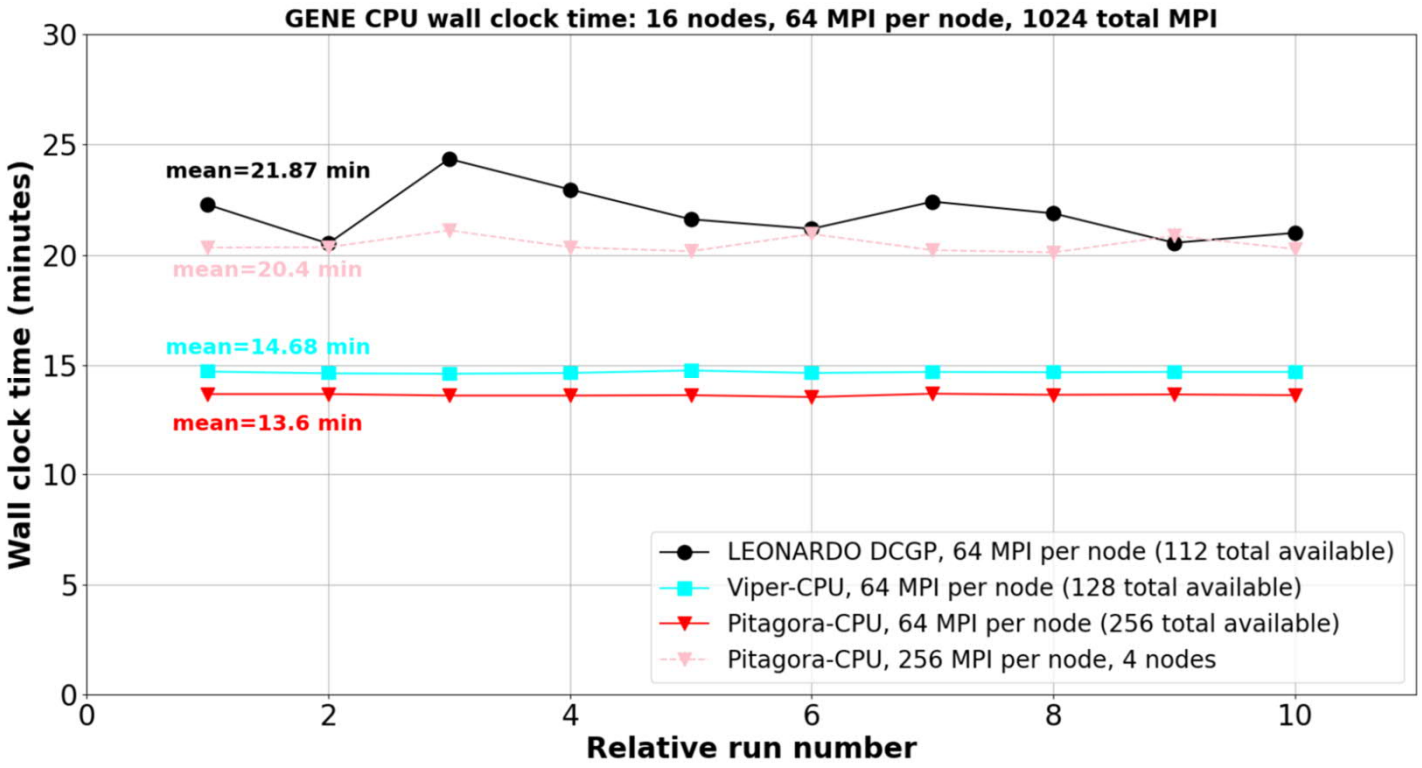
Pure MPI + ScaLAPACK

16 nodes, 96 MPI per node



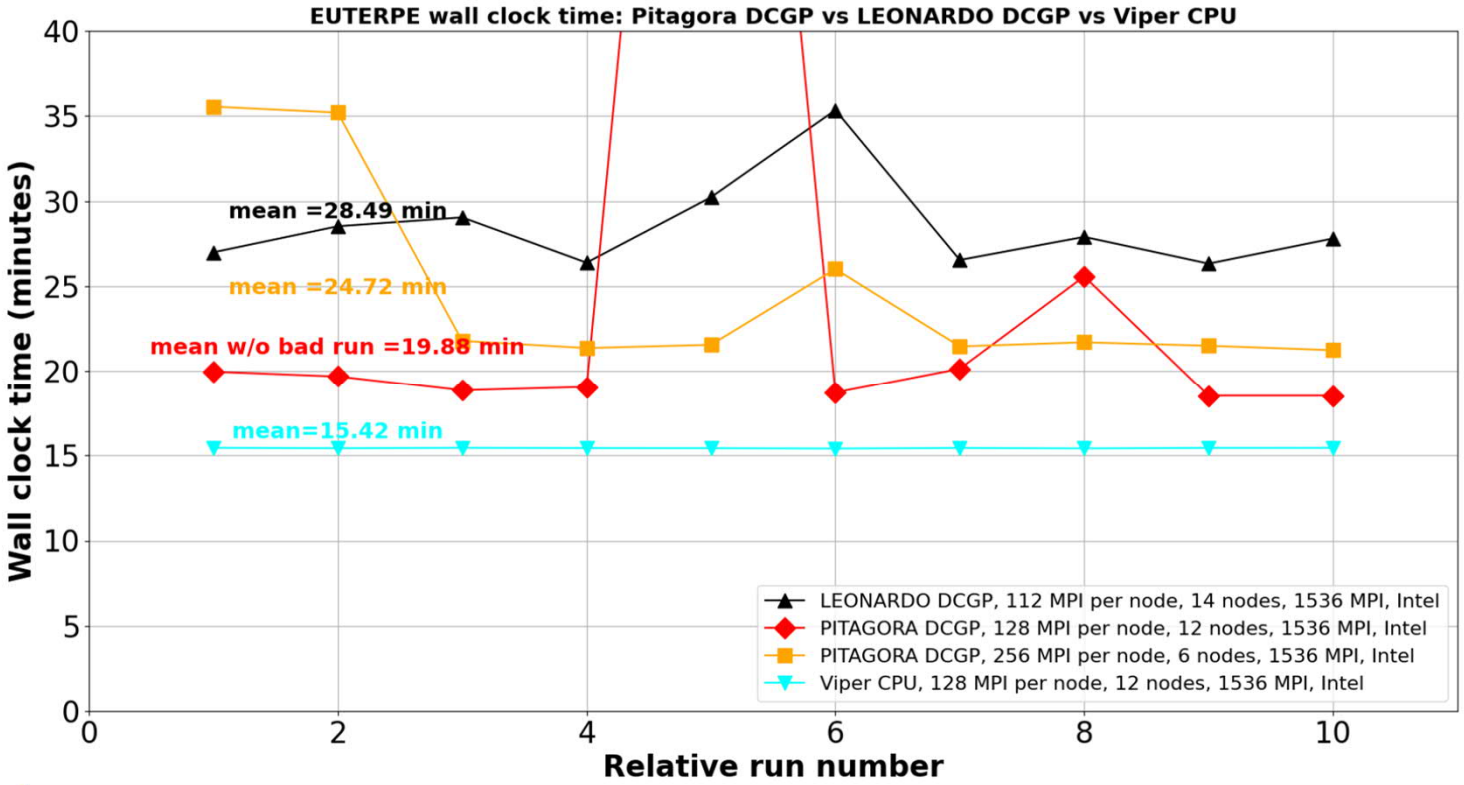
- **The execution time fluctuates on all supercomputers.**
- **Pitagora-CPU delivers the fastest performance, despite using under half of the node (96 out of 256 cores).**
- **Many runs failed on Pitagora-CPU due to an issue with a ScaLAPACK library subroutine that is still under investigation.**

GENE CPU performance (16 nodes)



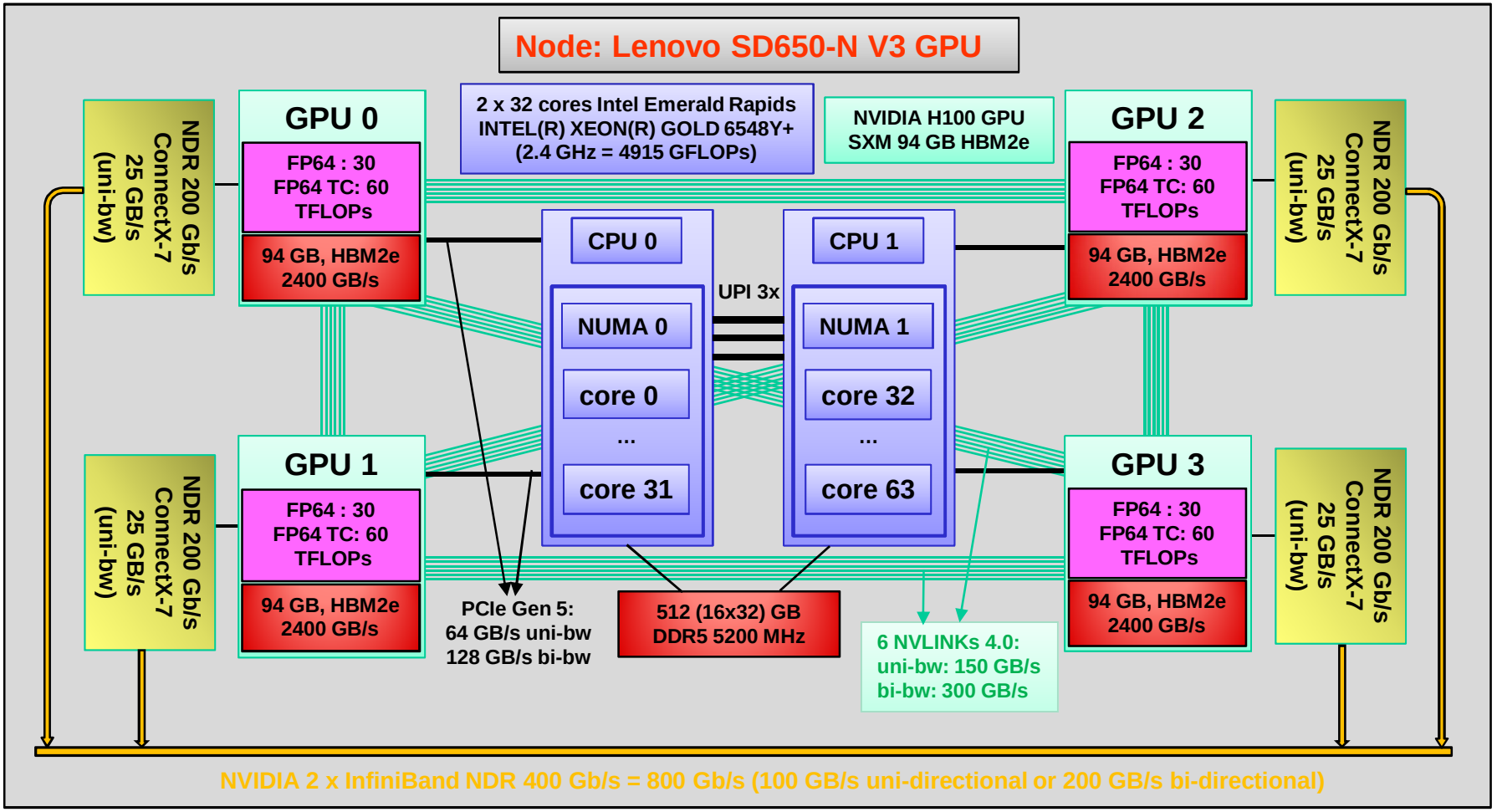
- The execution time is **stable** across all **Pitagora-CPU** runs.
- All runs completed **successfully** without any failed jobs.
- **Pitagora-CPU** is the fastest among all tested machines, though not as fast as expected compared to **Viper-CPU**.

EUTERPE CPU performance (6-14 nodes)

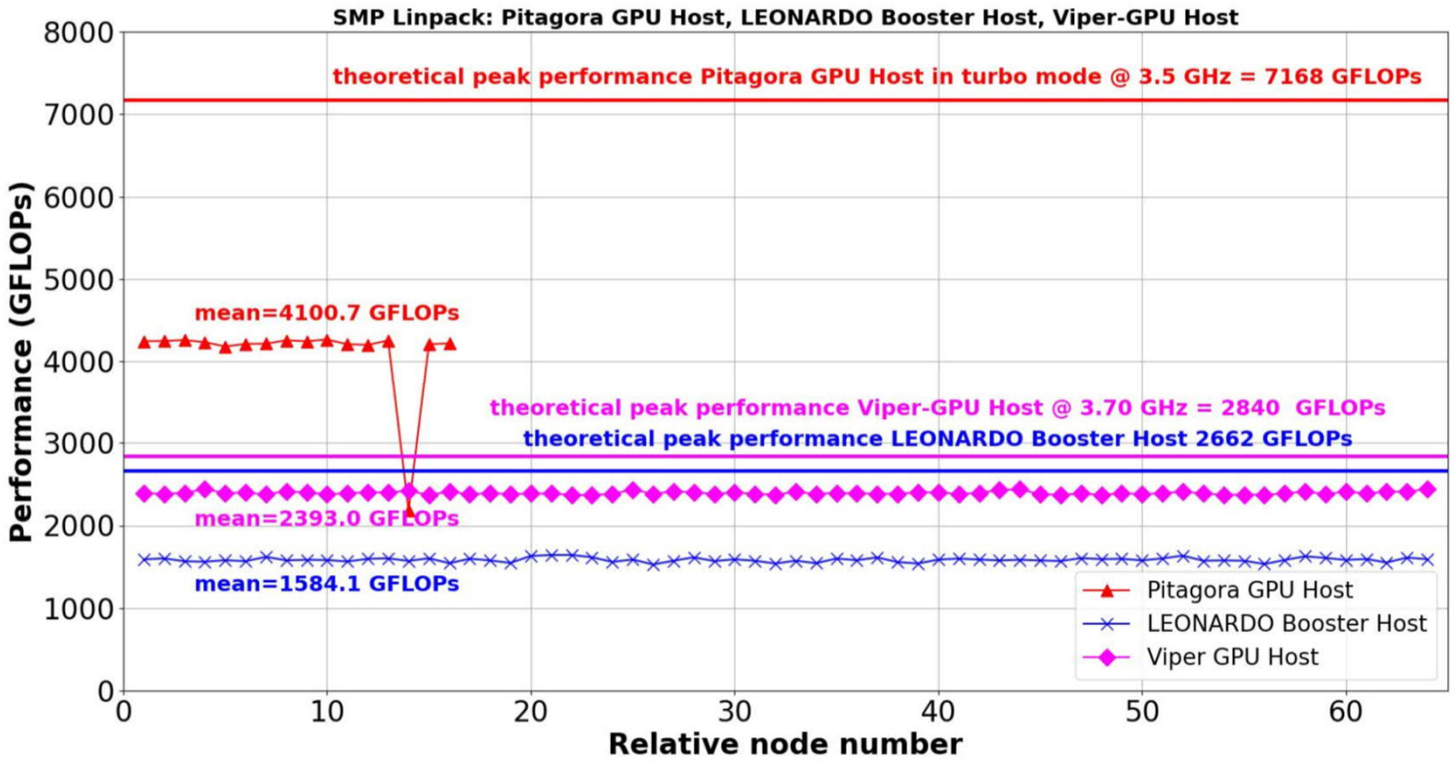


- **Using the same number of MPI tasks but fewer nodes (6 or 12), the execution time on Pitagora DCGP is faster by a factor compared to LEONARDO DCGP, which uses 14 nodes.**
- **The Viper CPU is not significantly faster when comparing core-to-core execution. However, in this test we used only half of a Pitagora DCGP node.**
- **Such tests also help detect ailing nodes, as shown by the red line.**

Lenovo SD650-N V3 GPU

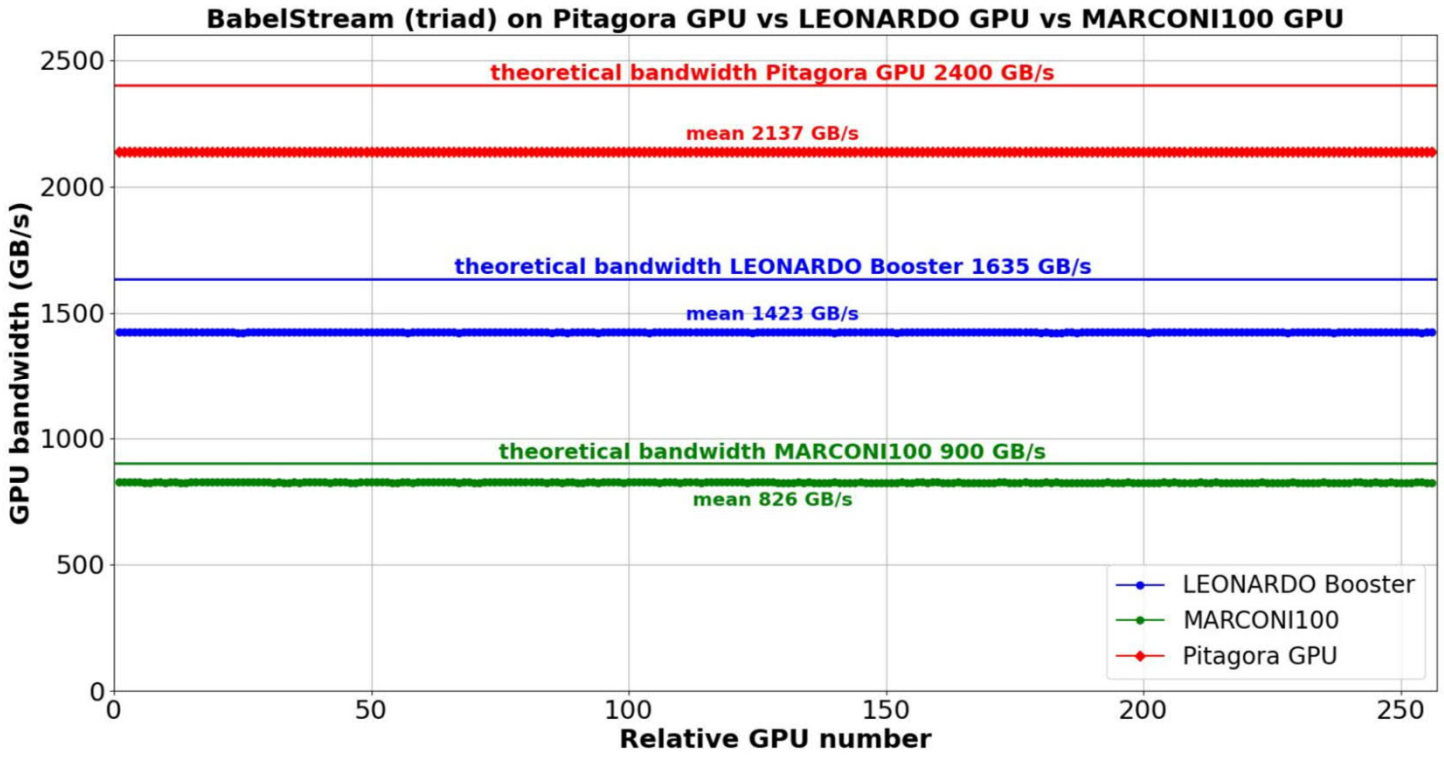


Pitagora GPU Host: performance stability test



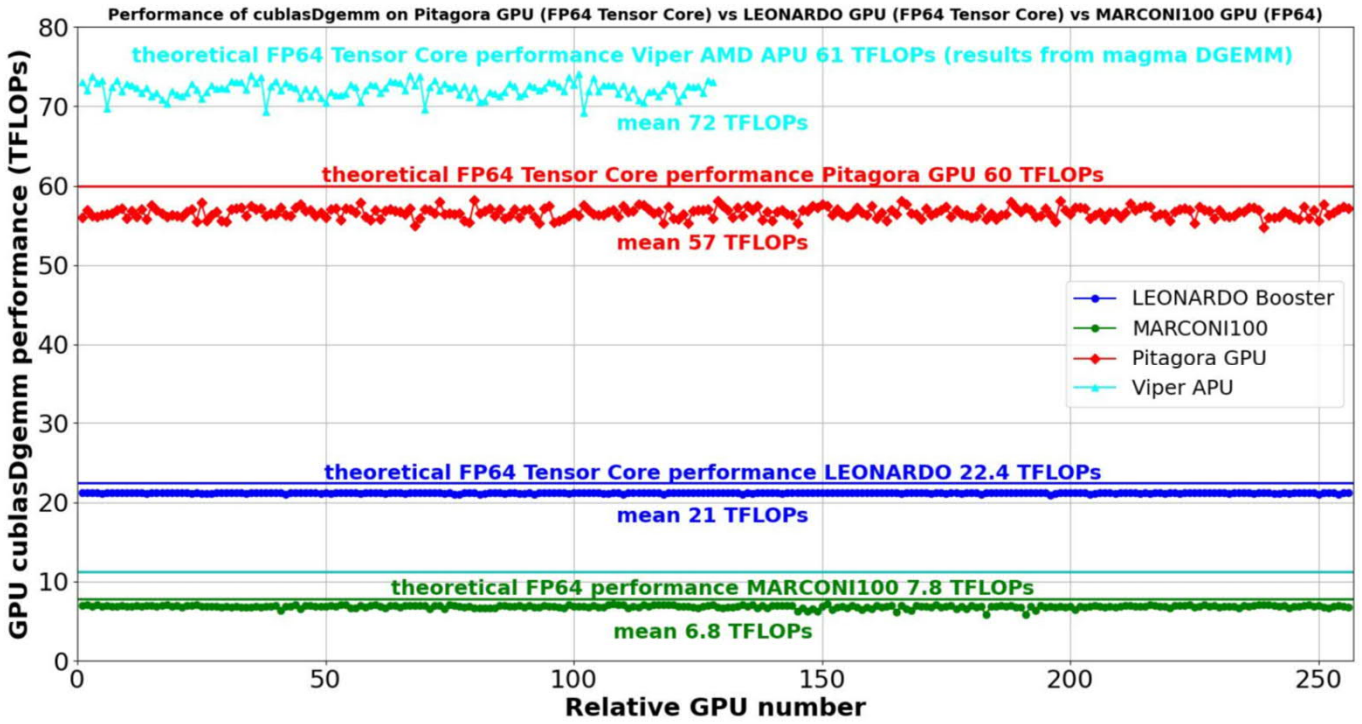
- **Pitagora GPU (mean): 4100 GFLOPs from 7168 GLOPs theoretical value (57%).**
- **Viper-GPU Host (mean): 2393 GFLOPs from 2840 GLOPs theoretical value (84%).**
- **LEONARDO Booster Host (mean): 1584 FLOPs from 2662 GFLOPs theoretical value (60%).**
- **We found several underperforming nodes.**

BabelStream benchmark on Pitagora GPU



- All GPUs provide high, stable and symmetric bandwidth close to the theoretical value.
- No difference between GPUs on different nodes or GPUs inside one node.
- Pitagora GPU (mean): 2137 GB/s from 2400 GB/s theoretical value (89%).
- LEONARDO Booster (mean): 1423.5 GB/s from 1635 GB/s theoretical value (87%).
- MARCONI100 (mean): 845 GB/s from 900 GB/s theoretical value (94%).

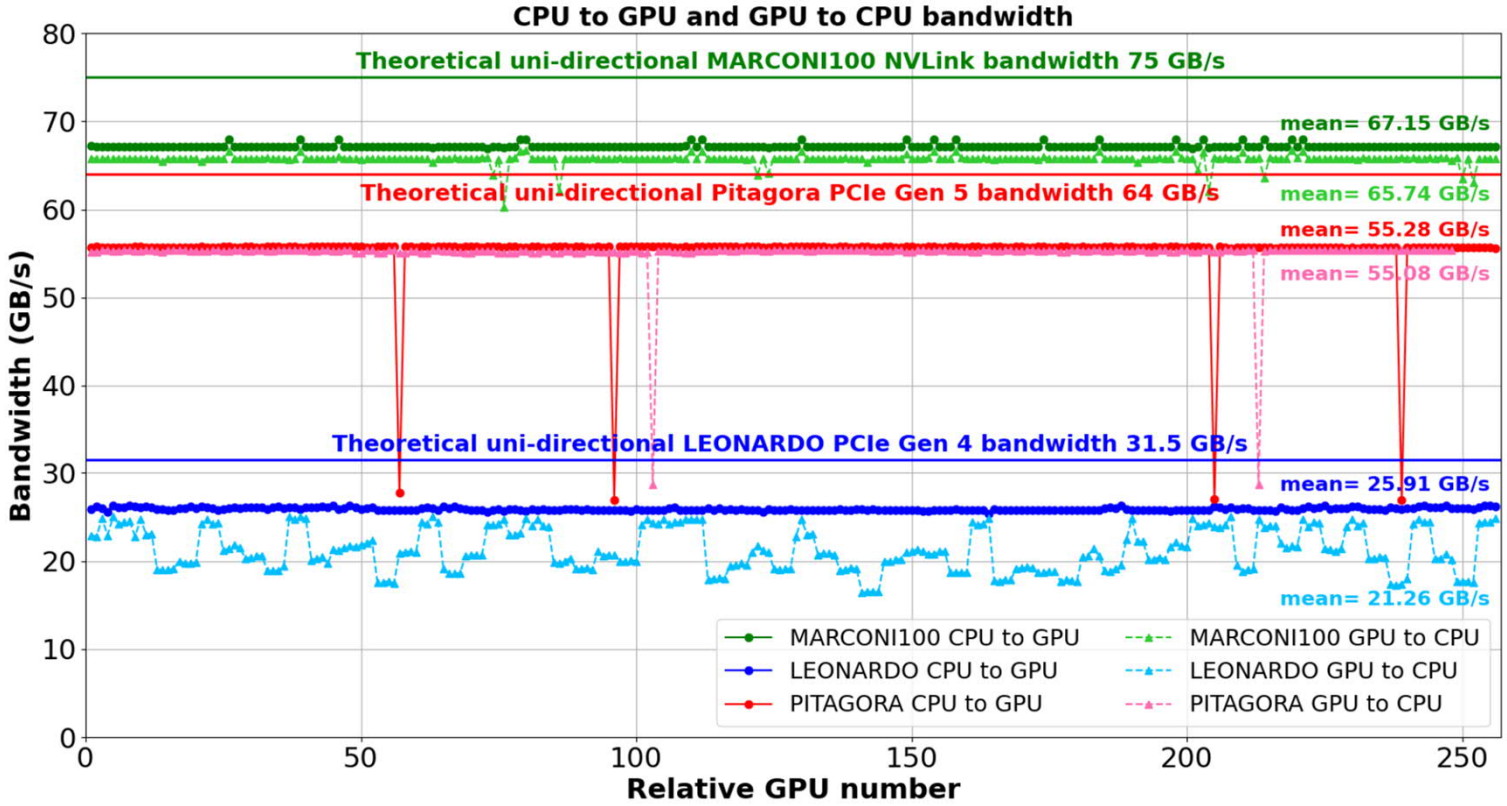
DGEMM (cublasDgemm) benchmark on Pitagora GPU



- All GPUs provide high, stable and symmetric performance close to the theoretical value.
- No difference between GPUs on different nodes or GPUs inside one node.
- Viper-GPU FP64 (mean): 72 TFLOPs per APU from 61 TFLOPs theoretical value (118%).
- Pitagora-GPU FP64 (mean): 57 TFLOPs per GPU from 60 TFLOPs theoretical value (95%).
- LEONARDO FP64 (mean): 21 TFLOPs per GPU from 22.4 TFLOPs theoretical value (94%).
- MARCONI100 FP64 (mean): 6.8 TFLOPs per GPU from 7.8 TFLOPs theoretical value (87%).

Pitagora-GPU: Host to Device connection

Host to Device connection is PCIe Gen 5:
- **128 GB/s bi-directional bandwidth**
- **64 GB/s uni-directional bandwidth**

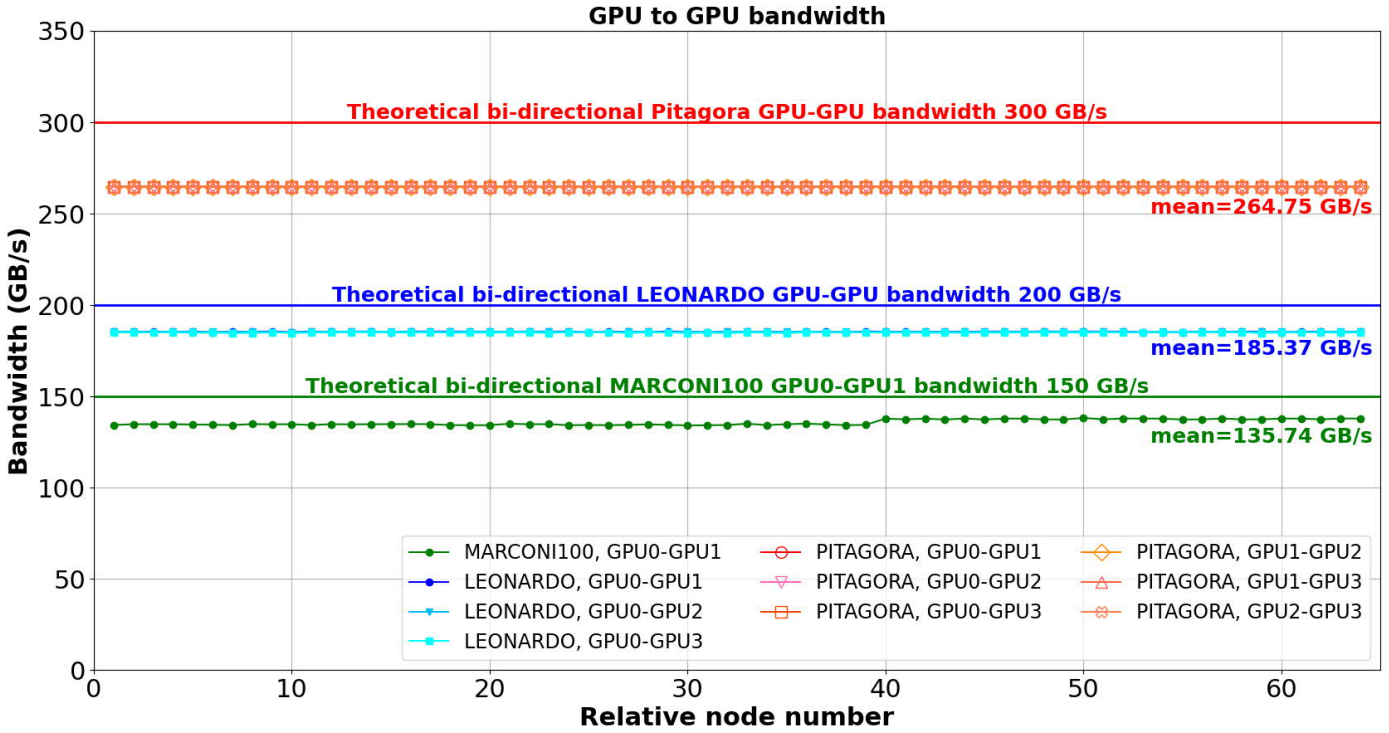


➢ **Some GPUs (or connections) are slower compared to others.**

Pitagora-GPU: Device to Device connection

Device to Device connection is NVLink 4.0:

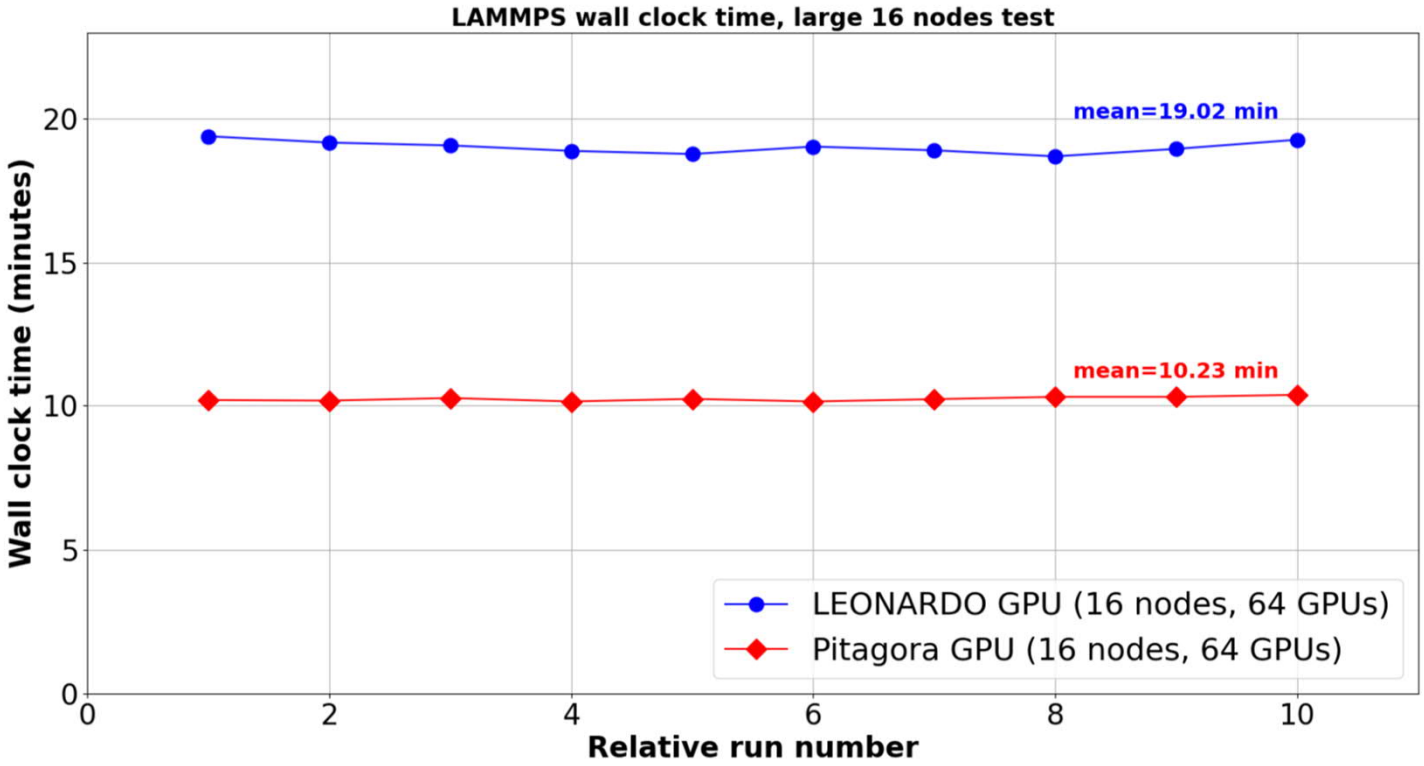
- **18 links (6 for each GPU pair) with 50 GB/s bi-directional bandwidth per link.**



- **The results are stable and symmetric.**
- **Pitagora-GPU: the mean bi-directional bandwidth of all GPU pairs 264.75 GB/s from 300 GB/s of the theoretical value (88%): 6 NVLinks with 50 GB/s each.**
- **LEONARDO: the mean bi-directional bandwidth of all GPU pairs 185.5 GB/s from 200 GB/s of the theoretical value (93%): 4 NVLinks with 50 GB/s each.**
- **MARCONI100: the mean bi-directional bandwidth of ~136 GB/s from 150 GB/s of the theoretical value (90%).**

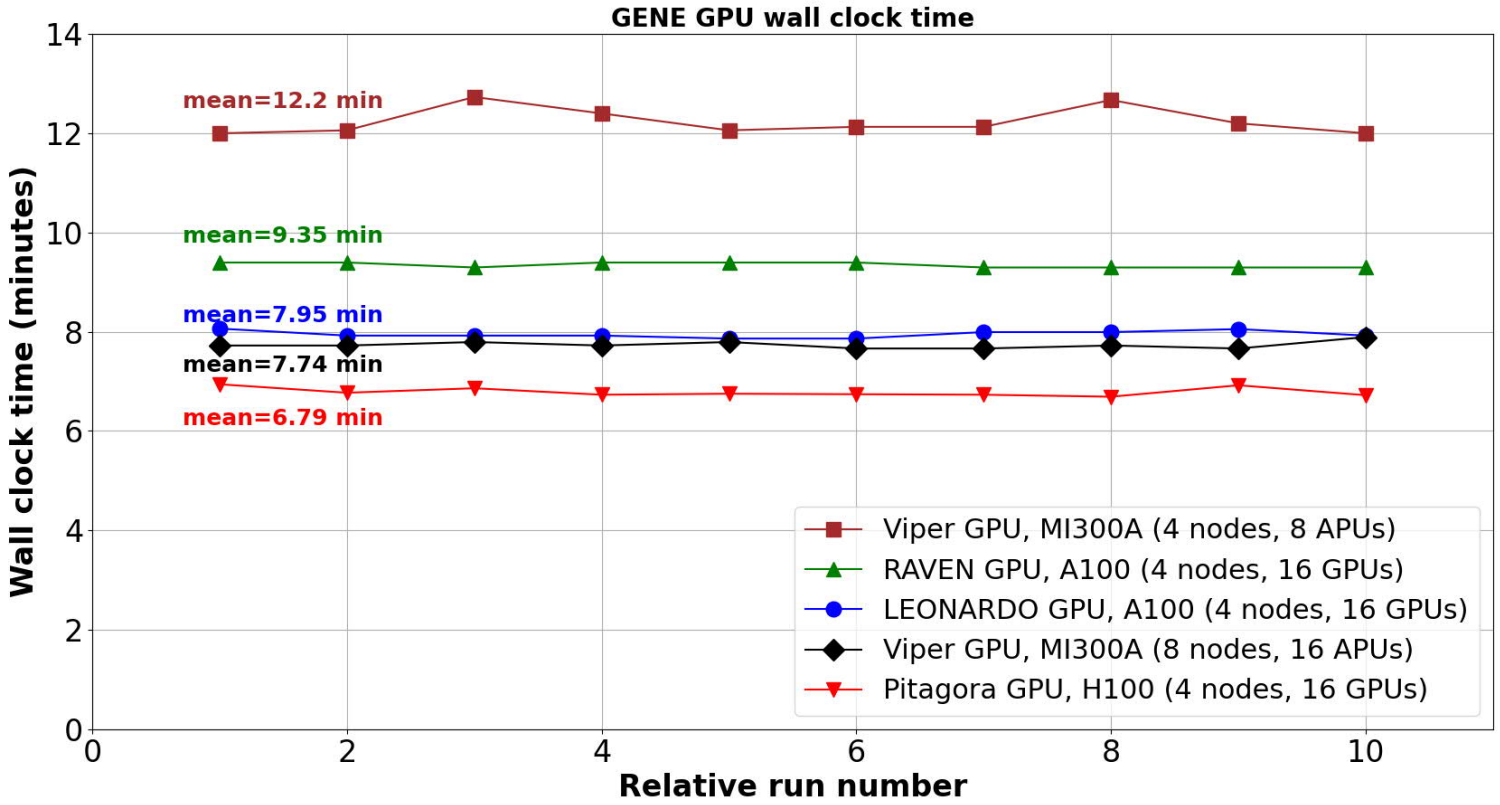
LAMMPS performance (large testcase)

16 nodes, 64 MPIs, 64 GPUs



- **The execution time is stable across supercomputers.**
- **On Pitagora, the code runs almost twice as fast as on LEONARDO Booster.**

GENE GPU performance (4 nodes, 16 GPUs)



- The execution time is **stable** across all **Pitagora-GPU** runs.
- All runs completed **successfully** without any failed jobs.
- **Pitagora-GPU** is the fastest among all tested machines, though not as fast as expected compared to **LEONARDO Booster**.

Thank you for our attention!

Key codes and their issues on Pitagora

Pitagora CPU:

SOLEDGE3X:

a) Compiled with Intel + verbs provider → multiple issues with OpenMP parallelization:

(i) Some occurred at compile time

(ii) Others occurred at runtime (segmentation faults)

→ The only workaround was to compile without OpenMP support, resulting in a performance downgrade for this hybrid MPI/OpenMP code.

b) Compiled with Intel + verbs provider → random crashes during PETSc calls.

c) Compiled with Intel + TCP provider → code appears to run properly, but:

(i) Random non-convergence of iterative solvers not seen on other machines

(ii) Slightly faster than GCC build, but still slower than on older Intel-based systems.

d) Compiled with GCC → UCX library for OpenMPI was built without. MPI_THREAD_MULTIPLE → very average performance.

CINECA recently provided OpenMPI + UCX-THREAD-MULTIPLE → tests still need to be performed.

e) Compiled with AOCC → compilation failed.

"In short, there is a clear issue with running with PETSc and IntelMPI that needs to be resolved. **The proposed work-around has a very bad impact on performances** and cannot be considered as a long-term solution."

Key codes and their issues on Pitagora

Pitagora CPU:

GENE:
a) Compiled with GCC → hangs or UCX errors;
b) Compiled with Intel + verbs provider → sometimes produces **wrong results**;
c) Compiled with Intel + TCP provider → performance slower by a factor of two (and sometimes hangs).

At the moment, **there is no good solution for the GENE code**.
CINECA has assigned a dedicated **team of several engineers** to address all these issues.

GeneX:
a) Compiled with Intel + verbs provider → the code can hang at any point of execution time: “I observed so far that sometimes after ~1500 steps, ~12000, ~30000, .. basically at any point this can happen. There is no abortion of the job, so the accounted time is essentially wasted.”
b) Compiled with Intel + TCP provider → “runtime explodes from 1s/step to roughly 13s/step”

At the moment, **there is no good solution for the GeneX code**.

Key codes and their issues on Pitagora

Pitagora CPU:

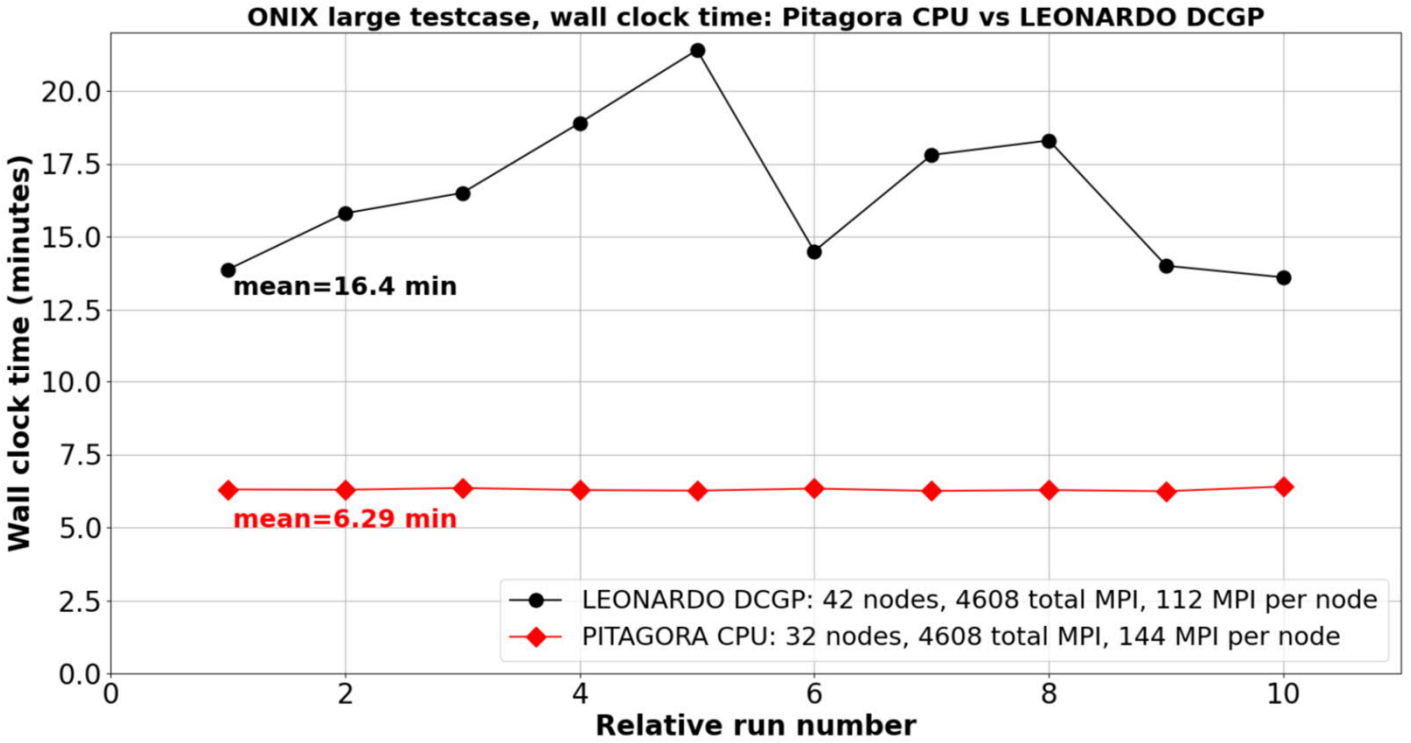
PETSc and **ScaLAPACK**:
a) Compiled with Intel + verbs provider → random crashes or hangs;
b) Compiled with Intel + TCP provider → significant performance degradation.

EUTERPE:
a) Compiled with AOCC → slow results with real version and **wrong results with complex version**.

Pitagora GPU:

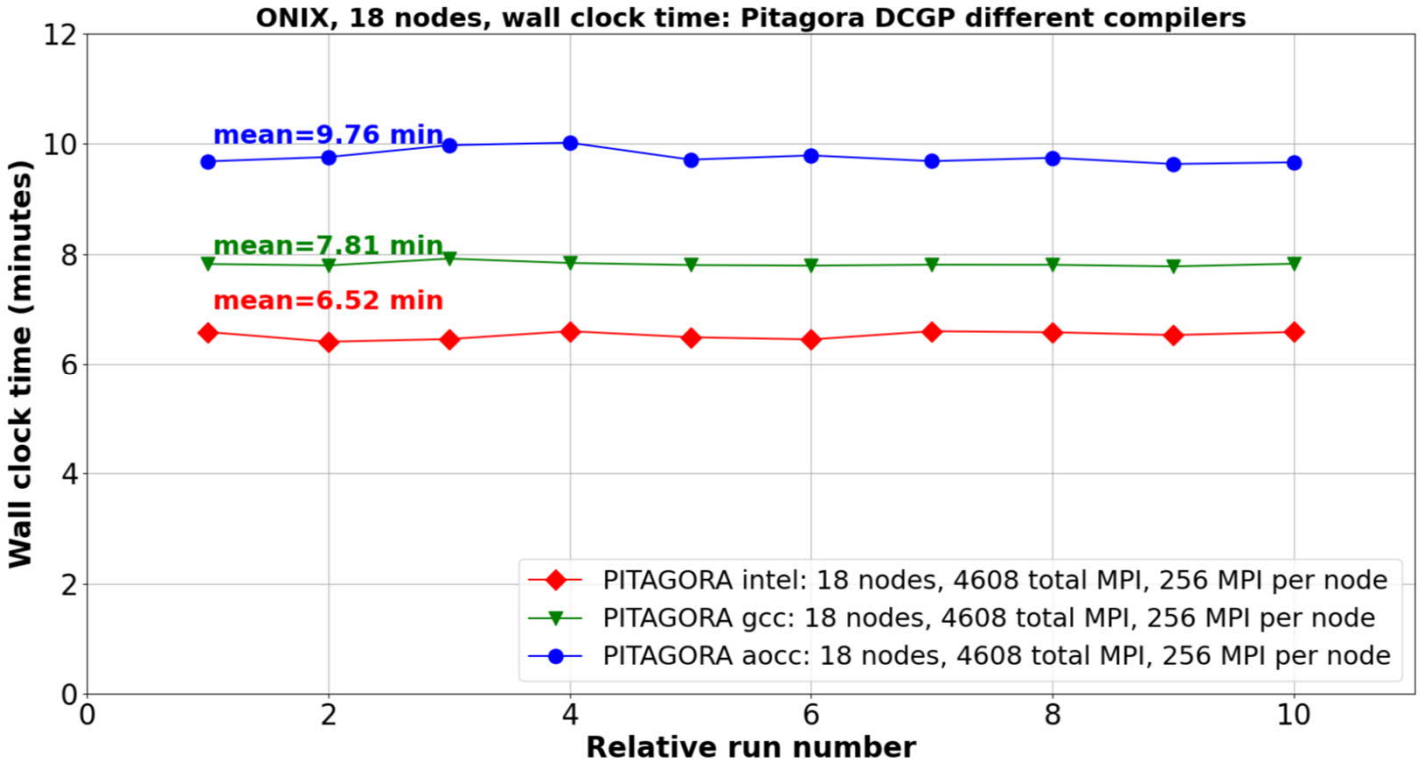
RDMA does not work, which has a performance impact on several applications. GENE is among them.

ONIX performance (large testcase)



- The execution time is stable across all Pitagora-CPU runs.
- Using fewer nodes (32 vs. 42) and only 144 MPI tasks per node (out of 256 available), Pitagora-CPU is 2.6× faster compared to LEONARDO DCGP.
- All runs completed successfully without any failed jobs.

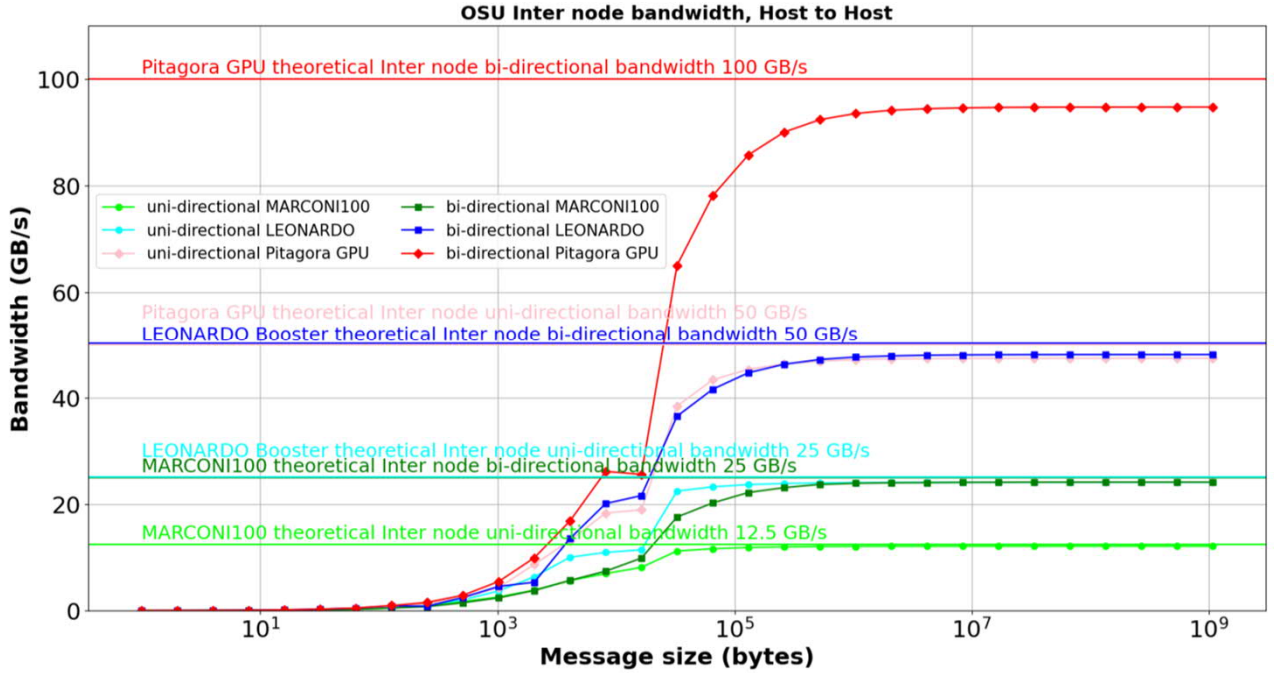
ONIX performance (different compilers)



- **Execution time is stable across all compilers.**
- **Intel compiler shows the best performance, while AOCC shows the worst.**
- **An old version of AOCC (4.2) is currently used; version 5.0, optimized for Zen5 CPUs (as on Pitagora), will be requested.**
- **This behavior does not generalize to all codes; for example, in some GENE test cases, GCC performs better than Intel.**

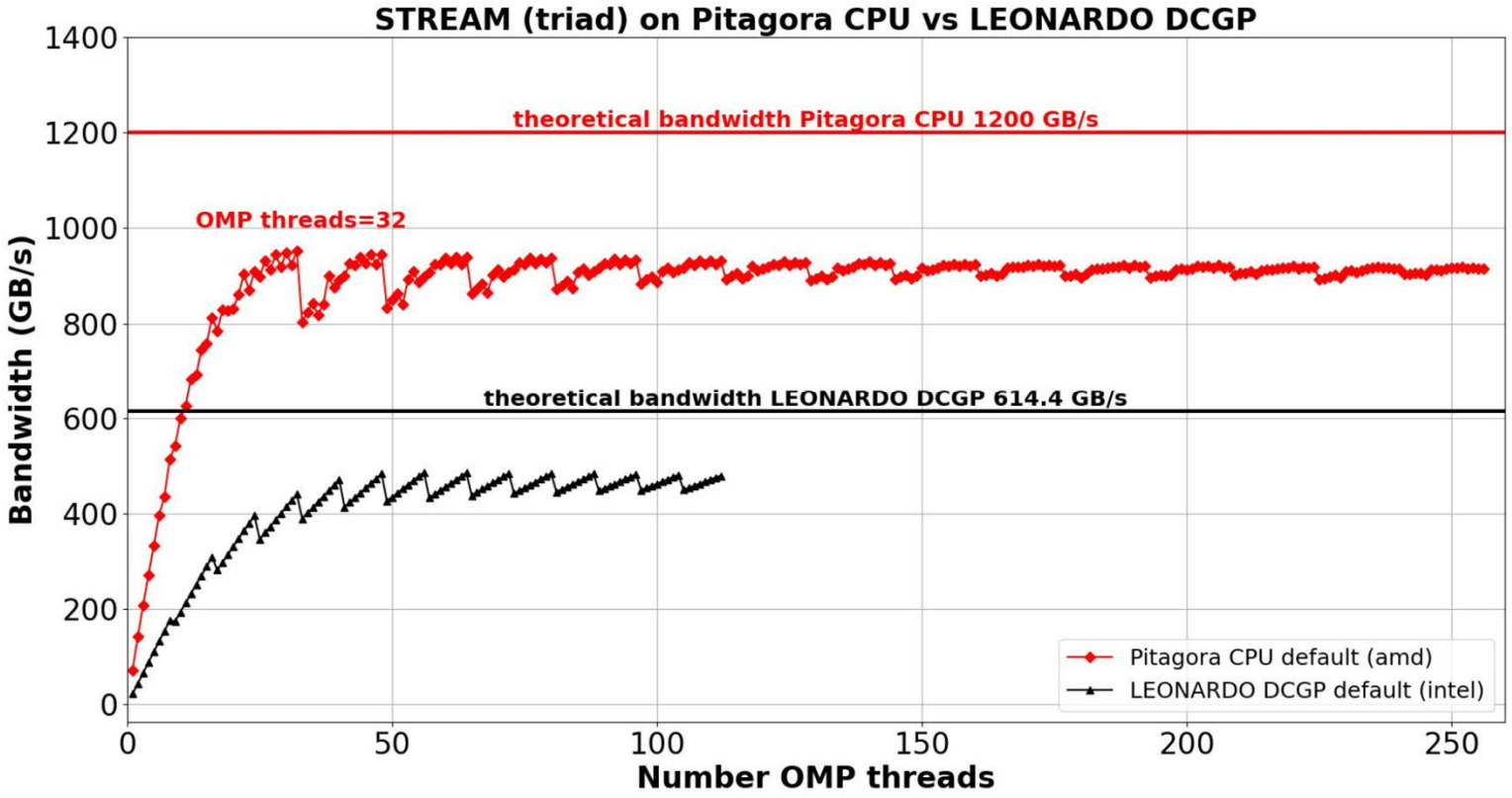
Pitagora-GPU inter-node network bandwidth

using osu_bw and osu_bibw benchmarks from OSU microbenchmark



- **Stable and high bandwidth for uni- and bi-directional data transfer.**
- **Pitagora-GPU: bi-directional bandwidth ~97 GB/s from 100 GB/s of the theoretical value (97%).**
- **Pitagora-GPU: uni-directional bandwidth ~49 GB/s from 50 GB/s of the theoretical value (98%).**
- **LEONARDO Booster: bi-directional bandwidth ~49 GB/s from 50 GB/s of the theoretical value (98%).**
- **LEONARDO Booster: uni-directional bandwidth ~24 GB/s from 25 GB/s of the theoretical value (96%).**
- **MARCONI100: bi-directional bandwidth ~24.2 GB/s from 25 GB/s of the theoretical value (97%).**
- **MARCONI100: uni-directional bandwidth ~12.1 GB/s from 12.5 GB/s of the theoretical value (99%).**

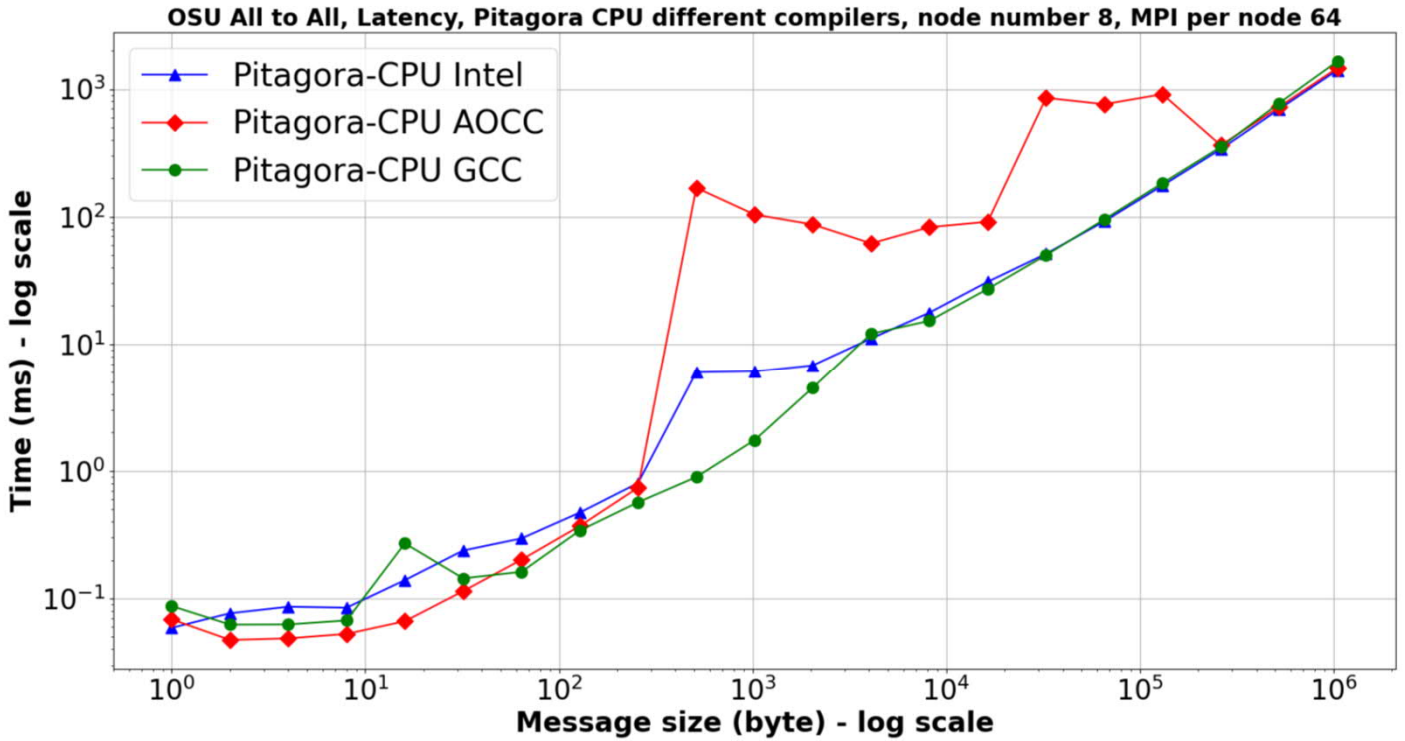
Pitagora CPU: STREAM on a single node



- **Pitagora CPU (mean): 916 GB/s from 1200 GB/s theoretical value (76%).**
- **Pitagora CPU: saturation with 32 OpenMP threads.**
- **LEONARDO DCGP (mean): 481 GB/s from 614.4 GB/s theoretical value (78%).**
- **LEONARDO DCGP: saturation with 48 OpenMP threads.**

Pitagora-CPU: *all_to_all* latency communication test

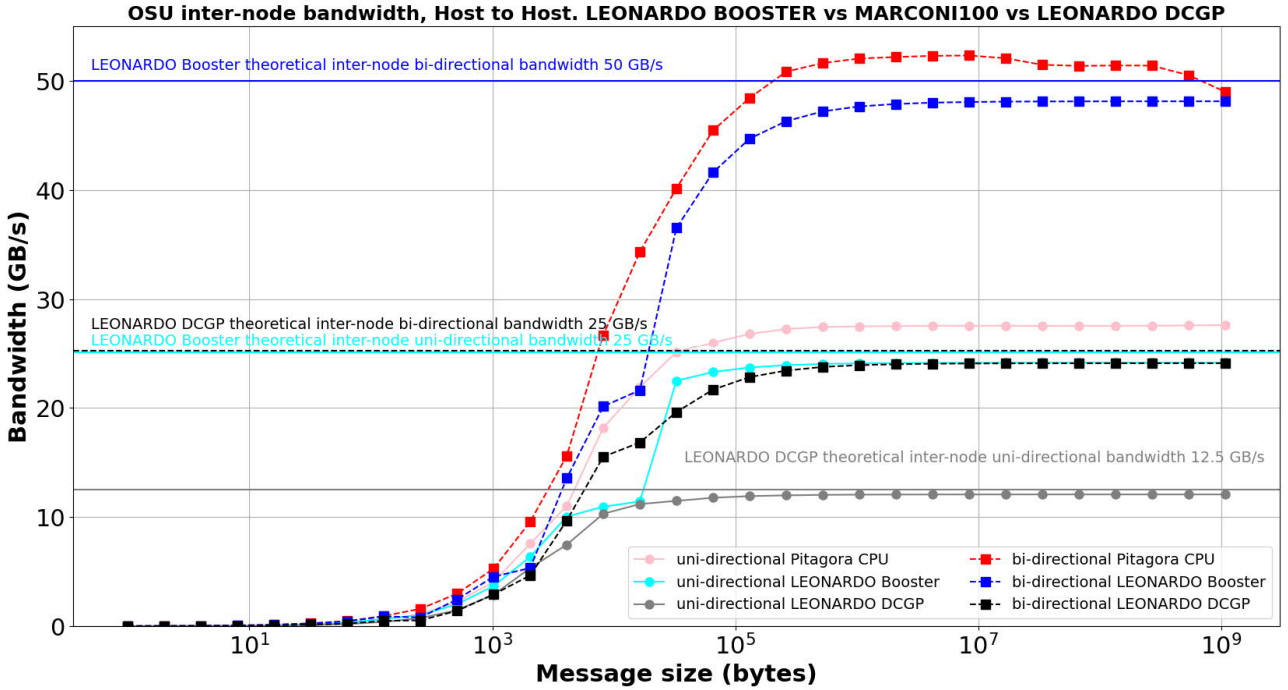
using osu_alltoall benchmark from OSU microbenchmark



- **Average time to complete the *all-to-all* operation increases with message size, as expected.**
- **AOCC performs better for small messages (<32 B), but is slower for intermediate sizes (256 B and 256 kB).**
- **For large messages (>256 kB), all three compilers show similar performance.**

Pitagora-CPU inter-node network bandwidth

using osu_bw and osu_bibw benchmarks from OSU microbenchmark

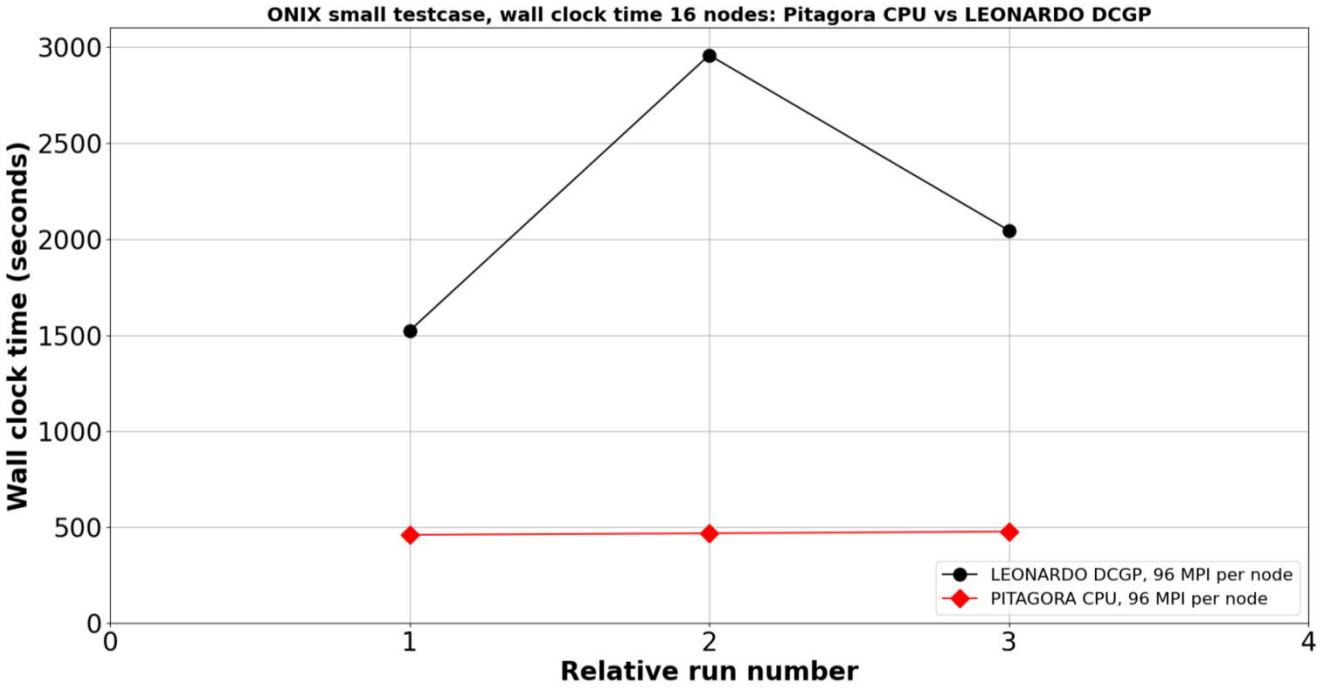


- **Stable and high bandwidth for uni- and bi-directional data transfer.**
- **Pitagora-CPU: bi-directional bandwidth ~52 GB/s.**
- **Pitagora-CPU: uni-directional bandwidth ~27 GB/s.**
- **LEONARDO: bi-directional bandwidth ~49 GB/s from 50 GB/s of the theoretical value (98%).**
- **LEONARDO: uni-directional bandwidth ~24 GB/s from 25 GB/s of the theoretical value (96%).**
- **LEONARDO DCGP: bi-directional bandwidth ~24.2 GB/s from 25 GB/s of the theoretical value (97%).**
- **LEONARDO DCGP : uni-directional bandwidth ~12.1 GB/s from 12.5 GB/s of the theoretical value (99%).**

ONIX performance (small testcase)

Pure MPI

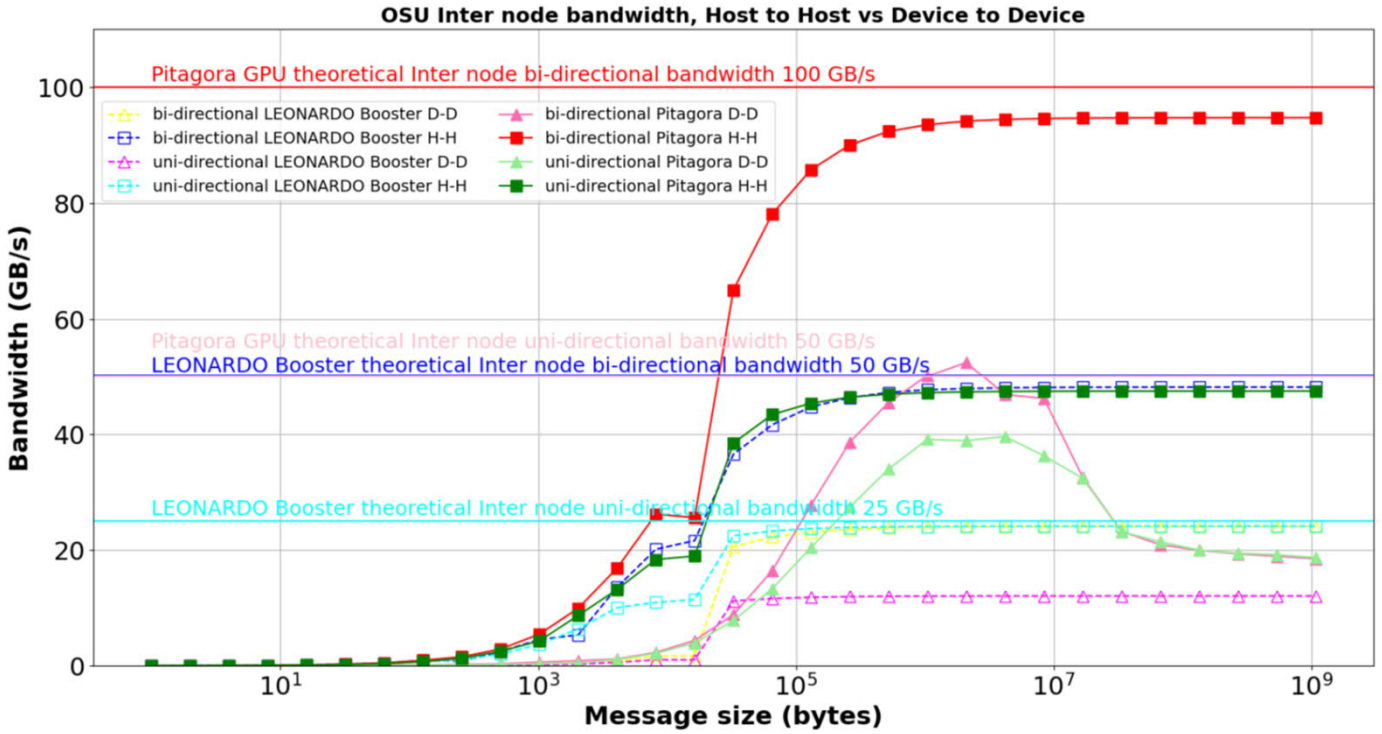
16 nodes, 96 MPI per node



- **The execution time is stable on all Pitagora-CPU.**
- **Pitagora-CPU delivers more than three times the performance of the LEONARDO DCGP partition, despite using less than half of the node (96 out of 256 cores).**

Pitagora-GPU inter-node network bandwidth

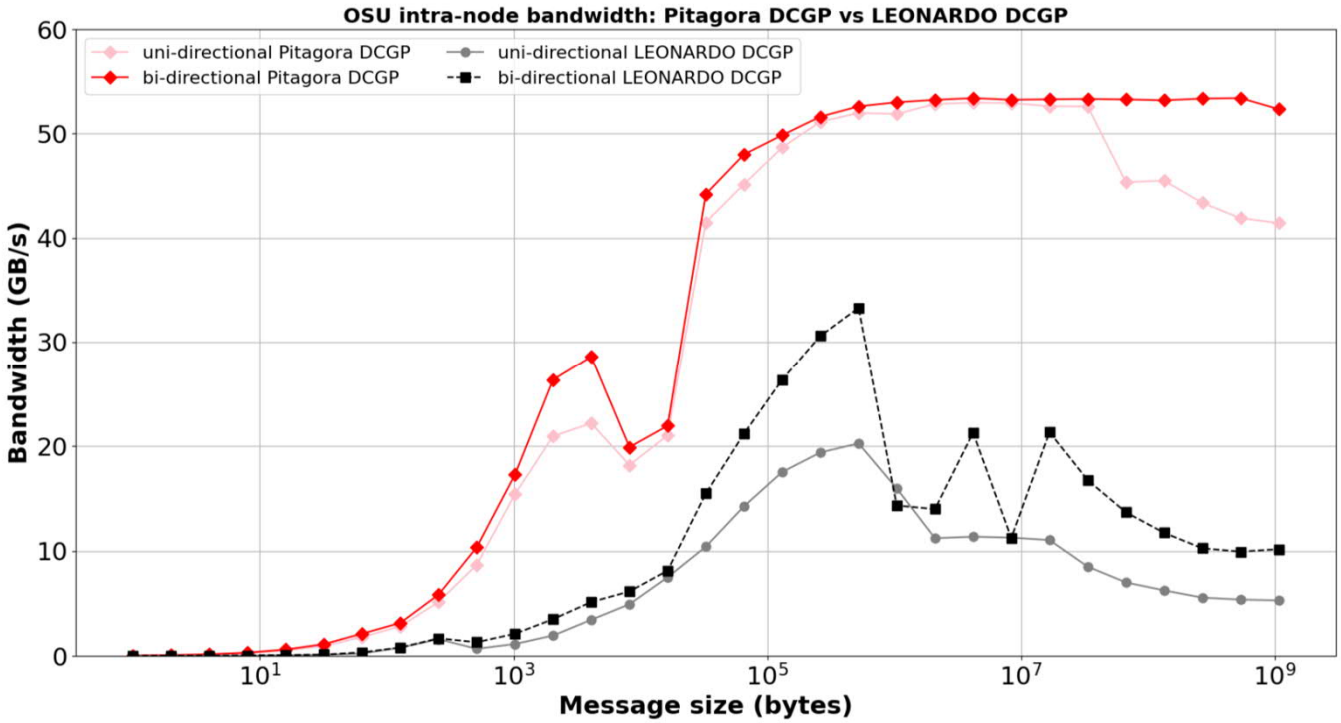
using osu_bw and osu_bibw benchmarks from OSU microbenchmark



- **Pitagora-GPU H-H: bi-directional bandwidth ~97 GB/s. Pitagora-GPU D-D: bi-directional bandwidth ~52 GB/s.**
- **Pitagora-GPU H-H: uni-directional bandwidth ~49 GB/s. Pitagora-GPU D-D: uni-directional bandwidth ~40 GB/s.**
- **LEONARDO Booster H-H: bi-directional bandwidth ~49 GB/s. LEONARDO Booster D-D: bi-directional bandwidth ~24 GB/s.**
- **LEONARDO Booster H-H: uni-directional bandwidth ~24 GB/s. LEONARDO Booster D-D: uni-directional bandwidth ~12 GB/s.**
- **Pitagora-GPU D-D transfer, for both uni- and bi-directional data, behaves as if the transfer is performed through the CPU.**

Pitagora-CPU: intra-node bandwidth

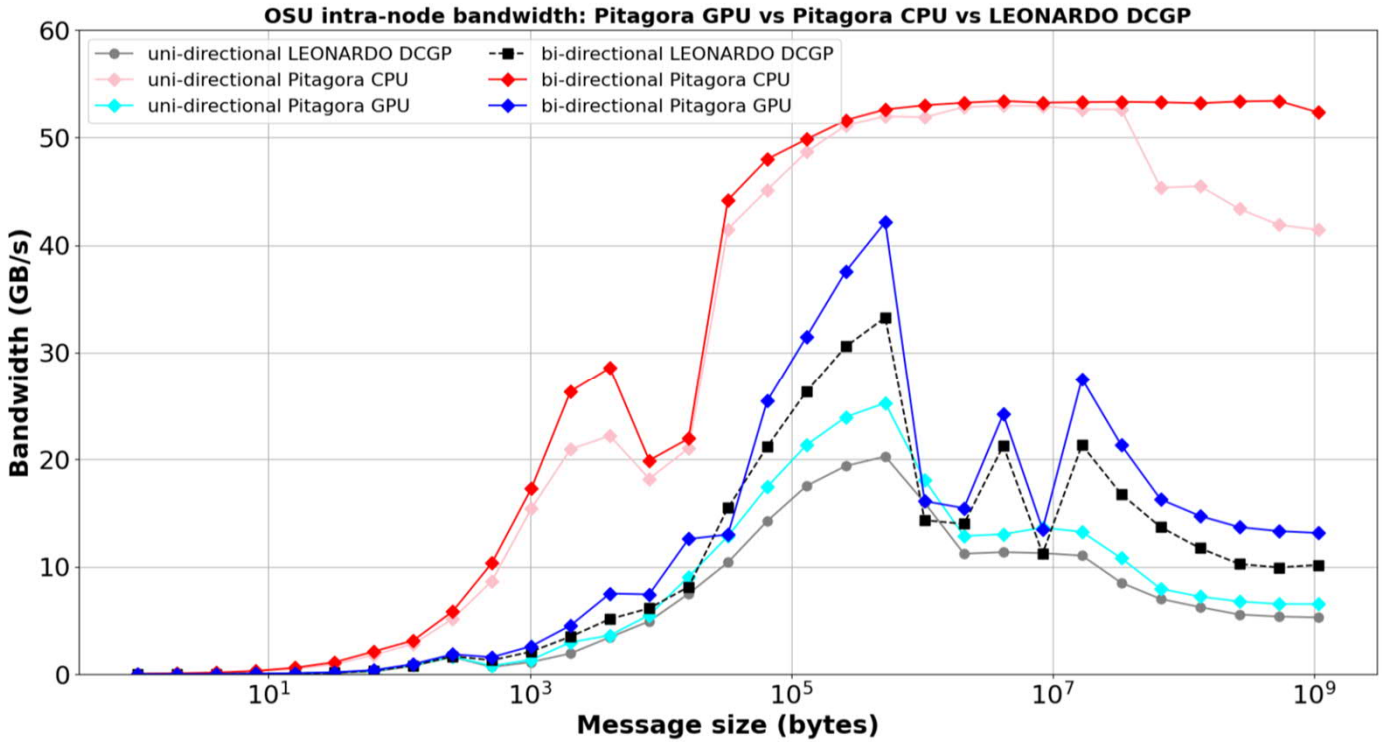
Host to Host connection is 4 xGMI links



- **MPI-Based Tools: Tools relying purely on MPI (e.g., OSU benchmarks) may underestimate bandwidth due to MPI internal buffering, library overheads, or communication scheduling. These are less reliable when targeting peak interconnect performance metrics.**

Pitagora-GPU and CPU: intra-node bandwidth

Host to Host connection is three UPI links 3:
Uni-directional: 3 × 44.8 GB/s = 134.4 GB/s
Bi-directional: 2 × 134.4 GB/s = 268.8 GB/s

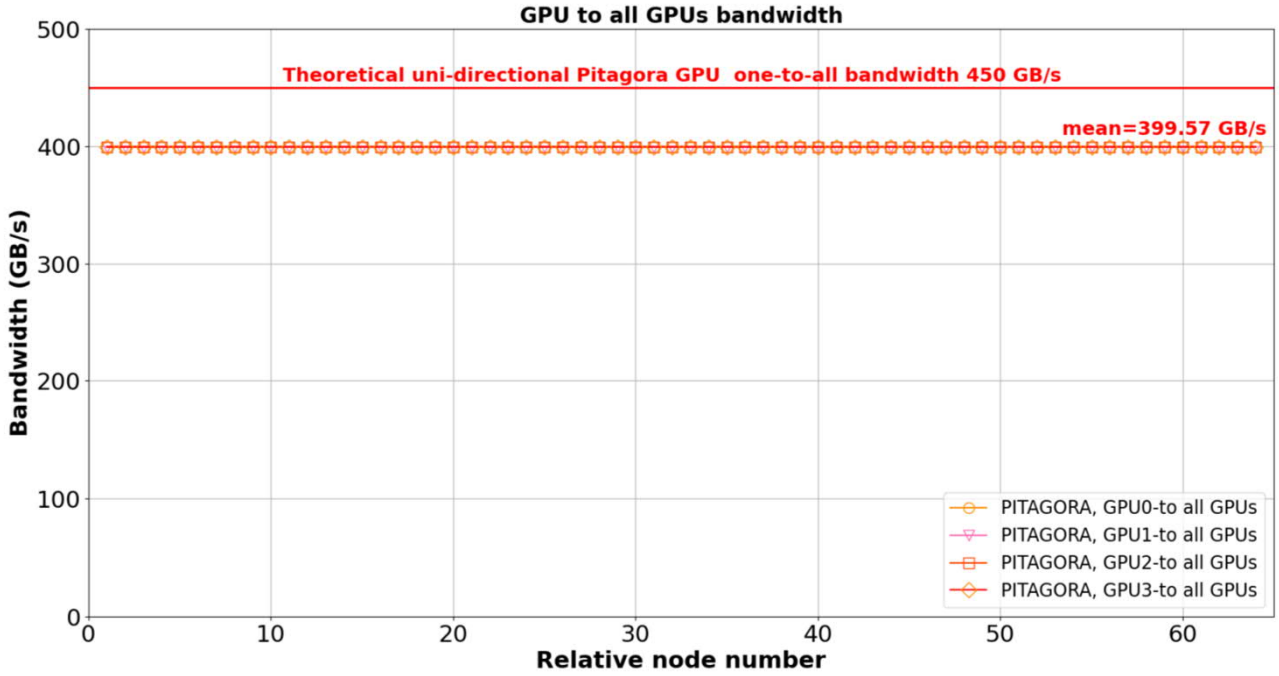


- **MPI-Based Tools: Tools relying purely on MPI (e.g., OSU benchmarks) may underestimate bandwidth due to MPI internal buffering, library overheads, or communication scheduling. These are less reliable when targeting peak interconnect performance metrics.**
- **Cineca in-house tool with multiple MPI pairs test reaches 209.62 GB/s for bi-directional data transfer**

Pitagora-GPU: GPU to all GPUs connection

Device to Device connection is NVLink 4.0:

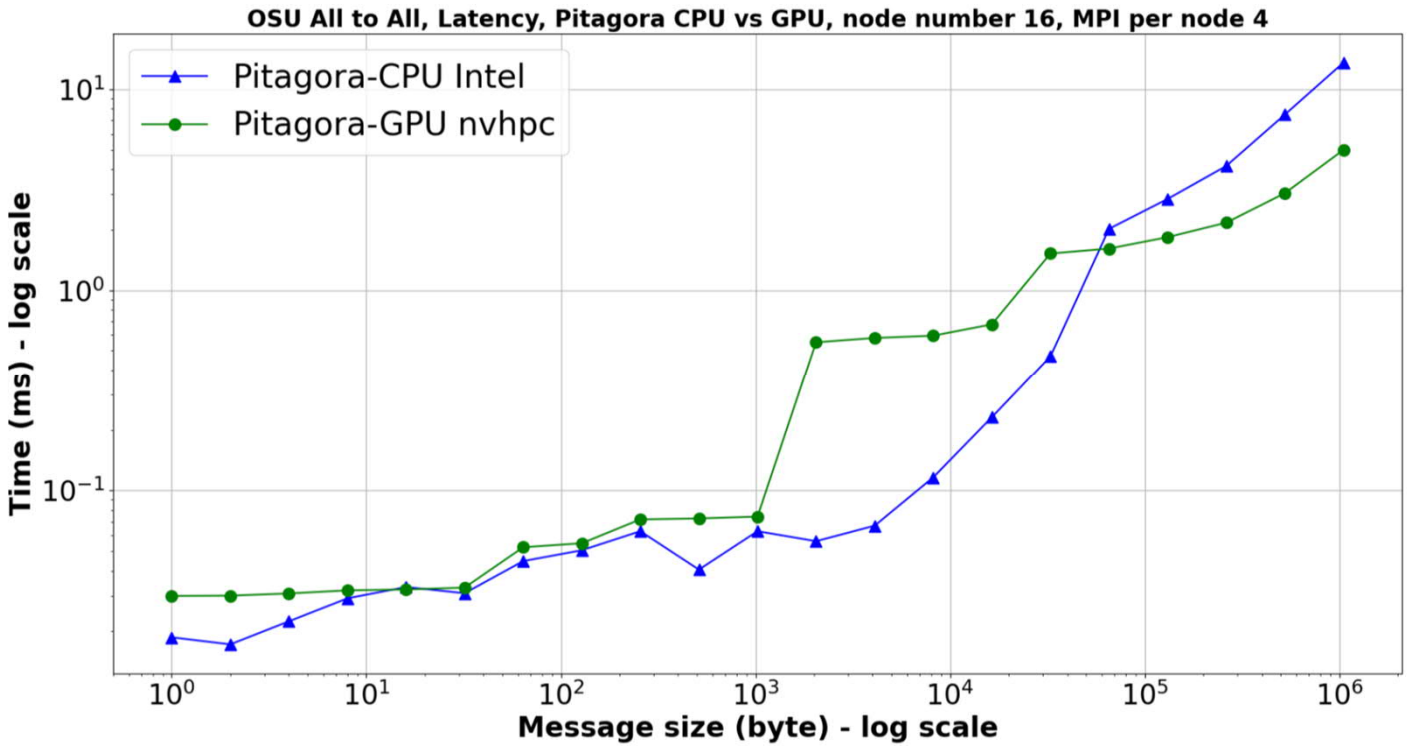
- **18 links (6 for each GPU pair) with 50 GB/s bi-directional bandwidth per link or 25 GB/s uni-directional.**



- **The results are stable and symmetric.**
- **Pitagora-GPU: the mean uni-directional bandwidth 399.57 GB/s from 450 GB/s of the theoretical value (89%).**

Pitagora-CPU vs GPU: *all_to_all*

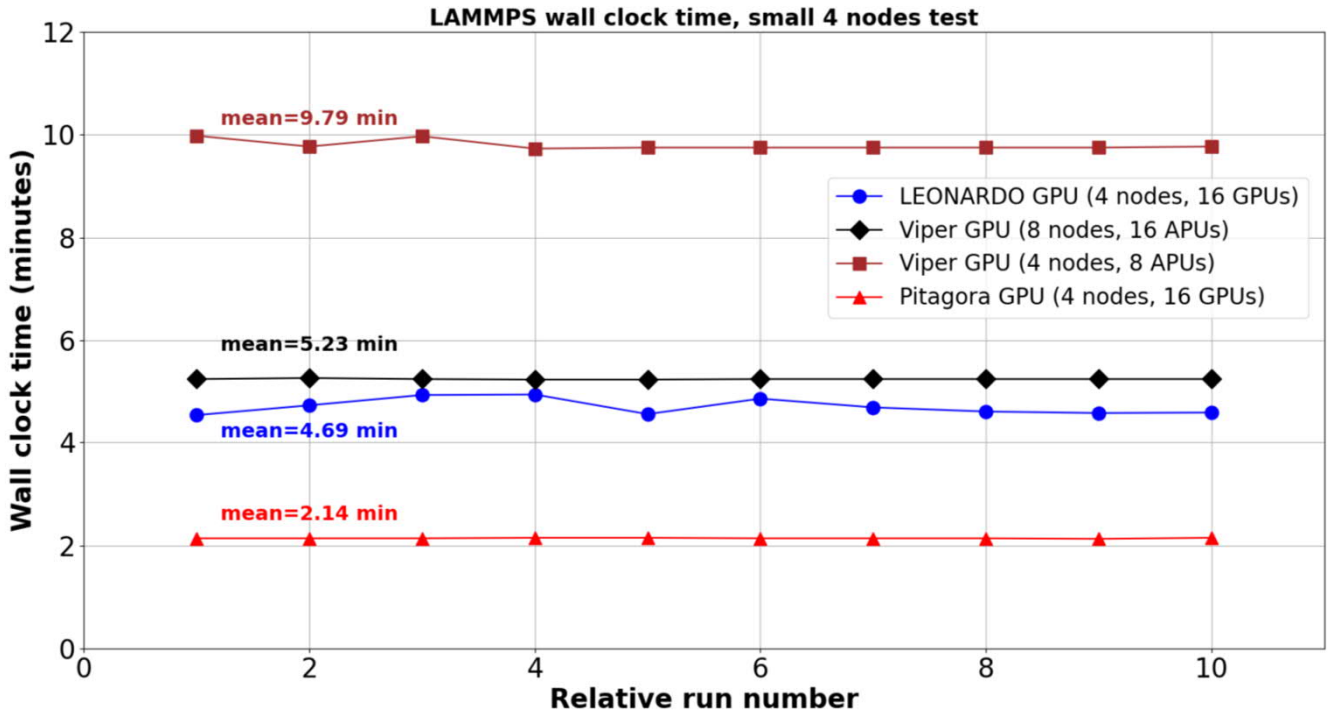
using osu_alltoall benchmark from OSU microbenchmark



- **Average time to complete the *all-to-all* operation increases with message size, as expected.**
- **Pitagora-CPU performs better for small messages (<32 kB).**
- **For large messages (>32 kB), Pitagora-GPU is faster.**

LAMMPS performance (small testcase)

4 nodes, 16 MPIs, 16 GPUs



- **The execution time is stable across supercomputers.**
- **On Pitagora, the code runs more than twice as fast compared to LEONARDO Booster and Viper-GPU.**