MareNostrum5's Graceful Landing

AHUG25, Hamburg

Presented by: Majesa Trimmel & Fabio Banchelli (BSC)

David Vicente (BSC) Marta Garcia-Gasulla (BSC) Filippo Mantovani (BSC)

Special thanks to: Filippo Spiga (NVIDIA)

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

European Processor Initiative

Introduction

Previous experience with Arm-based clusters

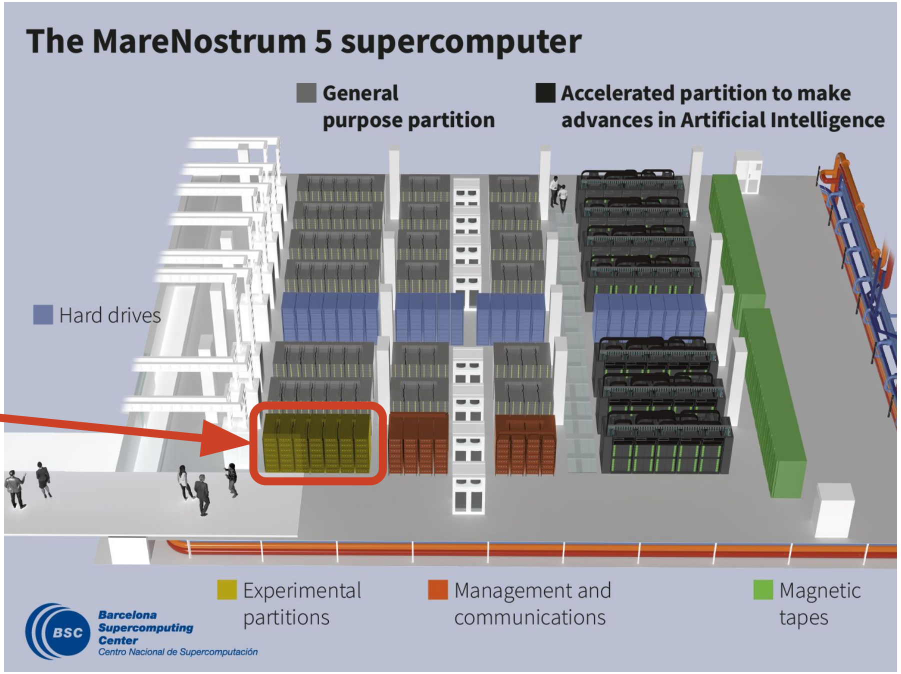
- Mont-Blanc EU Project
  - Odroid-XU
  - NVIDIA Jetson-TX
  - Cavium ThunderX
  - Marvell ThunderX2
- BSC-Huawei collaboration
  - Kunpeng
- MareNostrum4 Clusters of Emerging Technology
  - Fugaku
- Student cluster competition
  - Ampere Altra MAX ISC23
  - Grace-Grace ISC24
  - AmpereOne + H100 ISC25

📖 Cluster of emerging technology: evaluation of a production HPC system based on A64FX

MareNostrum5

Structure

- General Purpose Partition
  - Intel CPUs
- Accelerated Partition
  - Intel CPUs + NVIDIA GPUs
- Next Generation General Purpose Partition (Grace Partition)
  - NVIDIA Grace CPUs
- Next Generation Accelerated Partition
  - NVIDIA Grace CPUs + NVIDIA Hopper GPUs



Early evaluation cluster and MareNostrum5 comparison

- Early evaluation cluster with NVIDIA engineering samples
- Two hardware configurations
  - Grace-Grace (3 nodes)
  - Grace-Hopper (2 nodes)

- Master's Thesis
  - Cross-architecture benchmarking
  - Comparison with MN5 General Purpose and Accelerated Partition

⚠️ **Engineering samples:** Functional parts that only partially reflect the exact final product available to the mass market.

📖 NVIDIA Grace Superchip Early Evaluation for HPC Applications

Hardware

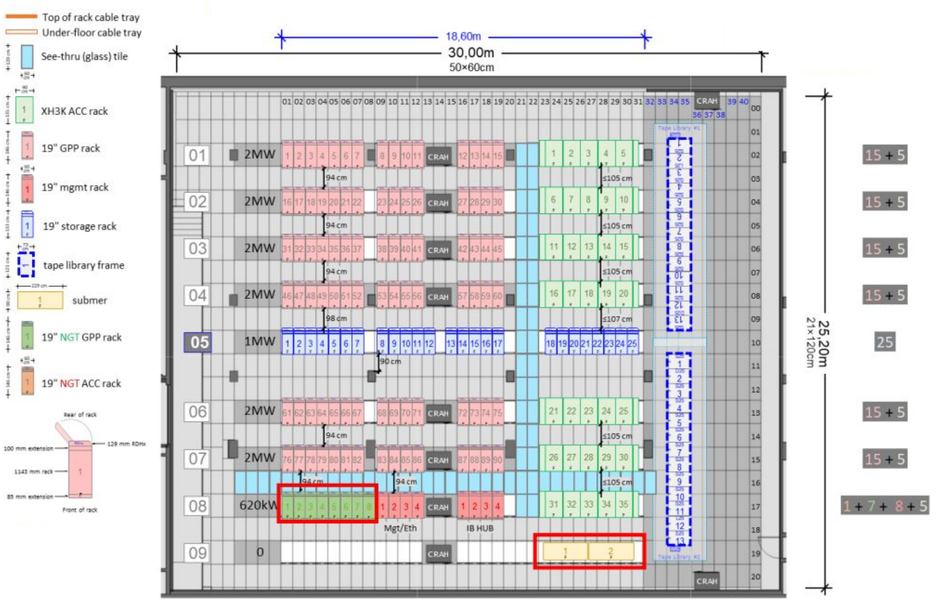
MareNostrum5 - Grace Partition

Partition

- 7 racks
- 60 nodes per rack + 48 nodes
- Total: 408 nodes

Node

- 2x NVIDIA Grace CPUs
- 144 cores @ 3.1 GHz
- 240 GB LPDDR5
- IB NDR200 - Full fat tree
- 2U chassis with 4 nodes



MareNostrum5 Partitions

	General Purpose Partition GPP	Accelerated Partition ACC	Grace Partition NGP
Architecture	x86	x86	Armv9
Micro-Architecture	Intel Sapphire Rapids	Intel Sapphire Rapids	Neoverse-V2
Frequency	2 GHz	2.3 GHz	3.1 GHz
Number of sockets	2	2	1
CPUs per socket	1	1	2
Cores per CPU	56	40	72
Cache sizes	L1: 32 KB (private) L2: 2048 KB (private) L3: 105 MB (shared)	L1: 32 KB (private) L2: 2048 KB (private) L3: 105 MB (shared)	L1: 64 KB (private) L2: 1024 KB (private) L3: 114 MB (shared)
Memory per node	256 GB	256 GB	240 GB
Memory Technology	DDR5	DDR5	LPDDR5

Scope

Benchmarks

Memory Latency:

- LMBench

Memory Bandwidth:

- LMBench
- STREAM (OpenMP)

Node-to-node Communication:

- OSU - Ohio State University Micro-Benchmarks

Floating-point Performance:

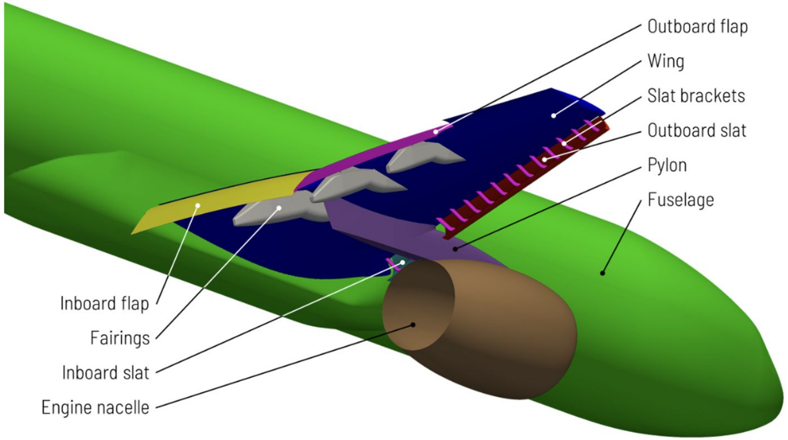
- HPL
- HPCG

Scientific Application

OpenFOAM:

- OpenFOAM v2312
- Developed by OpenCFD Ltd.
- Written in C++
- Parallelized with MPI

- MB9 micro-benchmark
- Simulation of the HPC Grand Challenge test case of the High Lift Common Research Model (CRM-HL)
- 1 node



System Software

Compilers

- GNU Compiler (`gcc/14.2.0`)
- LLVM Compiler (`llvm/18.1.14`)
- Arm Compiler (ACFL) (`acfl/24.10.1`)
- NVIDIA Compiler (`nvidia-hpc-sdk/24.11`)
- Intel Compiler (`oneapi/2023.2.0`)

Runtime

- OpenMPI (`openmpi/4.1.6`)
- MPICH (`mpich/4.3.0`)
- NVIDIA HPC SDK (`nvidia-hpc-sdk/24.11`)

Flags

- `-mcpu=neoverse-v2` vs. `-march=native`

🔑 **More options → More decisions to make**

System Software

TALP Performance Metrics:

- *Parallel Efficiency:* Load balance + communication
- *Communication Efficiency:* Wait time + data transfer delays
- *Load Balance:* Work evenly split?

Power Monitoring - Energy Aware Runtime (EAR):

- *Power:* Rate of energy consumption over time
- *Energy:* Total power consumed during execution
- *Energy-Delay Product (EDP):* Balances energy use and elapsed time

📖 A Generic Performance Analysis Technique Applied to Different CFD Methods for HPC

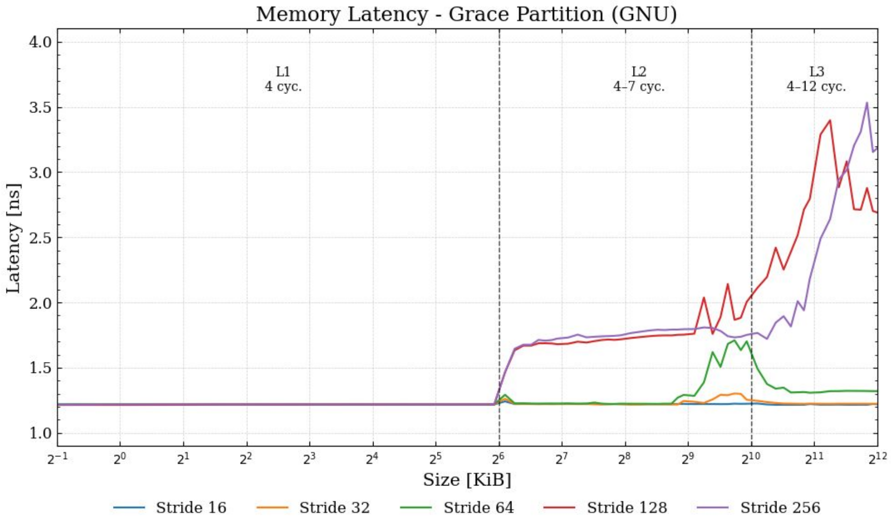
📖 Energy Optimization and Analysis with EAR

Low-Level Benchmarks

Memory Latency

- Does not match expectations
  - Step function that increases with every level

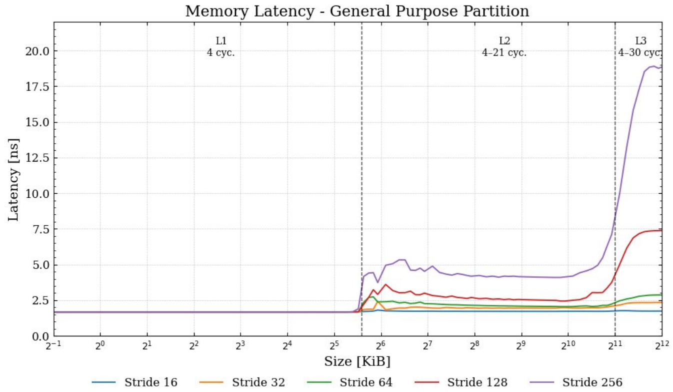
- No increase in latency for smaller strides (16, 32, 64)
- More typical behavior for larger strides

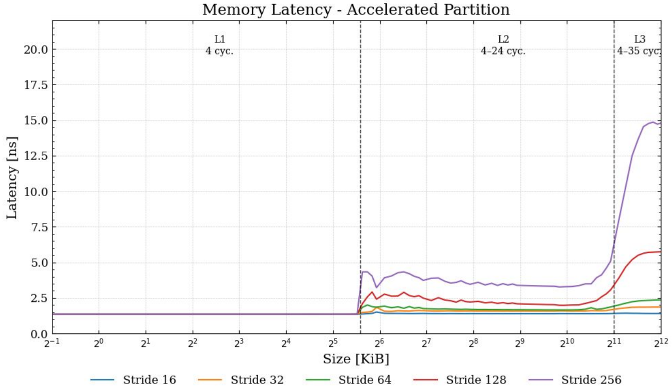


Memory Latency

- Matches predicted performance trends
- Step function that increases with each cache level

- Stark contrast to Grace Partition
- Aggressive hardware prefetcher
- Can dereference pointers with a "Sampling Indirect Prefetcher"

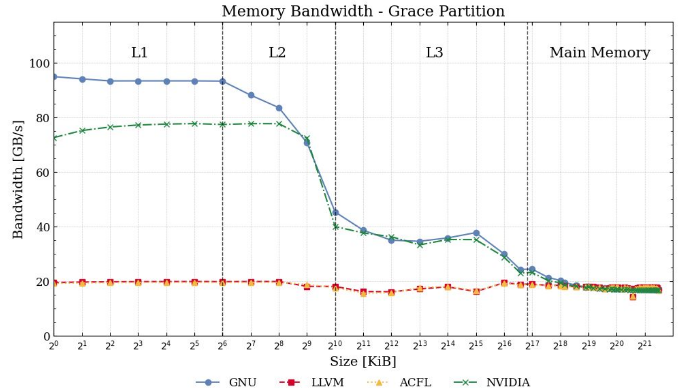


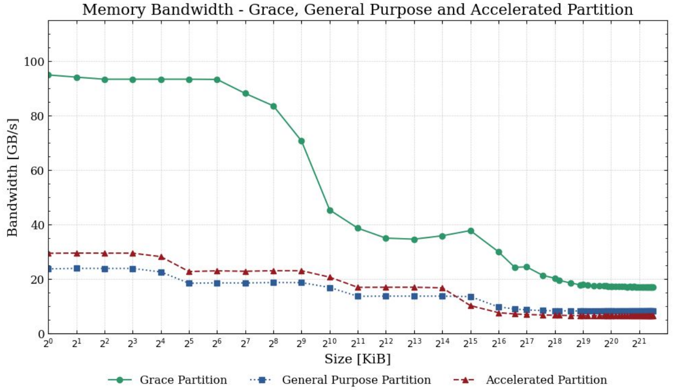


Memory Bandwidth

- Memory bandwidth values differ with compiler
- GNU and NVIDIA deliver higher bandwidth
- LLVM and ACFL show flat behavior
- Similar behavior in other memory related benchmarks with LLVM and ACFL

- Grace partition achieves significantly higher bandwidth compared to GPP and ACC





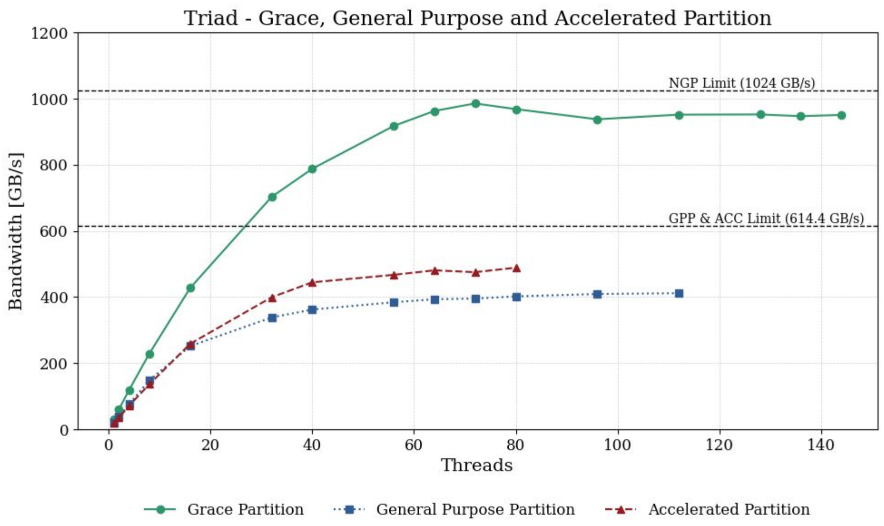
Different behaviors with different compilers

Memory Bandwidth

- Grace Partition scales best
  (~ 1 TB/s)
- 95.59% of peak
- Diminishing return after 72 cores

- General Purpose Partition reaches ~400 GB/s
- 65% of peak

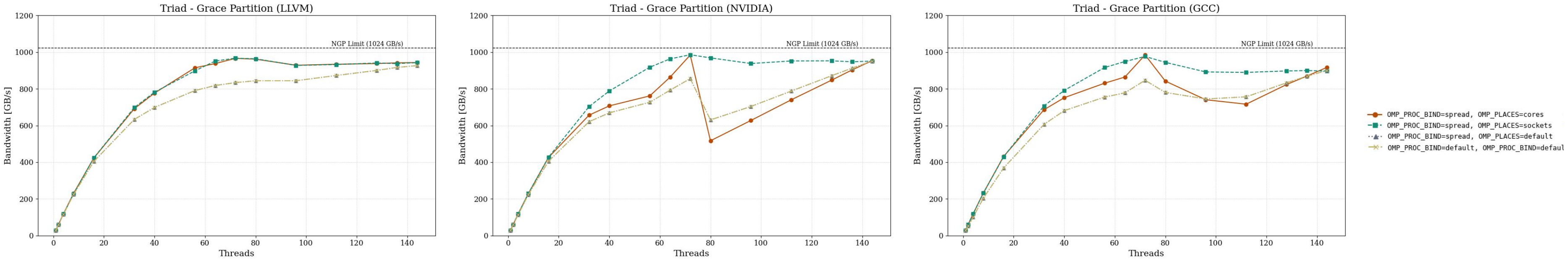
- Accelerated Partition reaches ~450 GB/s
- 75.38% of peak



Memory Bandwidth

- Substantial variations with different OpenMP bindings
- Dependent on compiler runtime
- LLVM and ACFL work as expected
- NVIDIA and GNU atypical behavior after 1 full CPU

```
OMP_PROC_BIND = [default, spread]
OMP_PLACES = [default, cores, sockets]
```

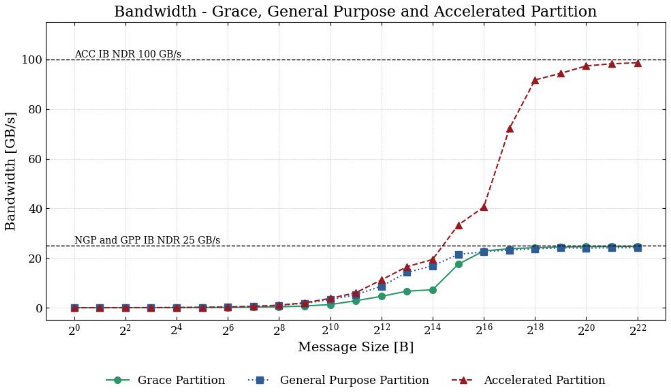


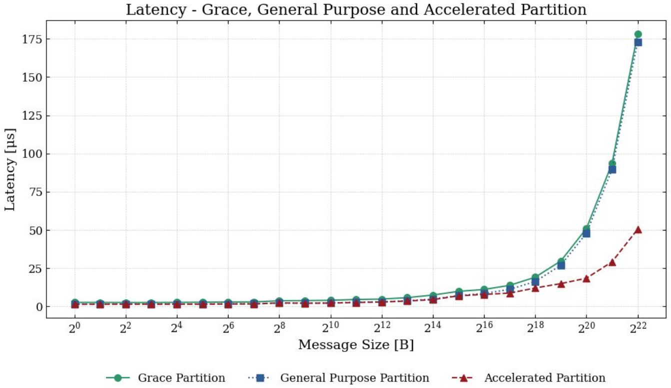
Check bindings and affinity!

Node-to-node Communication

- ACC has highest bandwidth and lowest latency
- NGP and GPP similar behavior

- Grace partition able to utilize Infiniband network well
- Bandwidth stays flatter longer for NGP



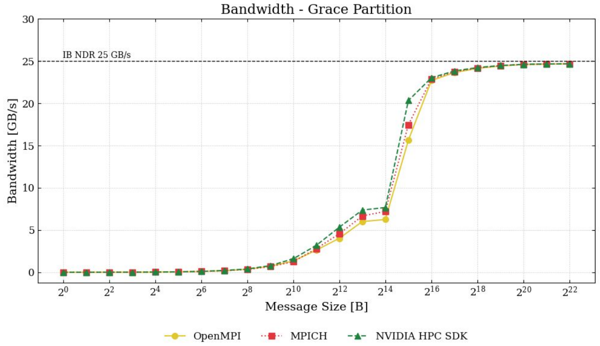


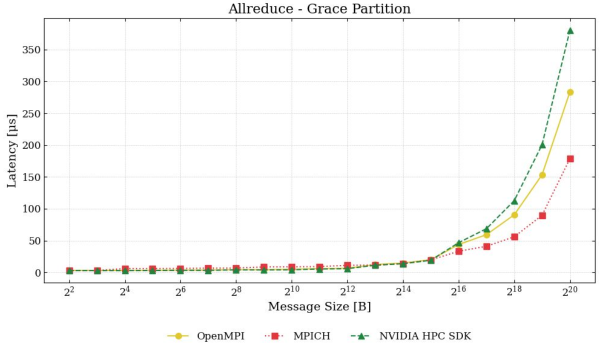
Node-to-node Communication

- Steep increase after 2^{14} B
- NVIDIA HPC SDK ramps up fastest
- All implementations saturate near 25 GB/s

- Similar behavior in other Arm-based machines

- Allreduce is stable for small messages, but increases sharply beyond 2^{15} B
- MPICH shows lowest latency overall





📖 Performance and energy consumption of HPC workloads on a cluster based on Arm ThunderX2 CPU

💡 **Check runtime options to get the best available one**

HPC Benchmarks

HPL

NGP (1 node):

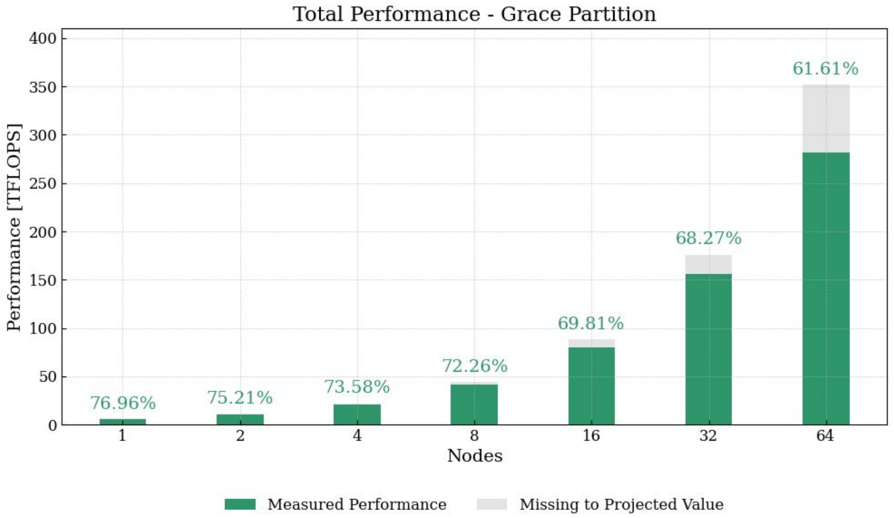
- Rpeak: 7.14 TFLOPS
- RMax: 5.50 TFLOPS
- %Rpeak: 76.96%
- %Memory: 83%

GPP (1 node):

- RMax: 6.61 TFLOPS
- %Rpeak: 92.19%

ACC (1 node - CPU and GPU):

- RMax: 179.70 TFLOPS
- %Rpeak: 77.46%



NVIDIA Grace is not a floating-point centric CPU

HPL

NGP (1 node):

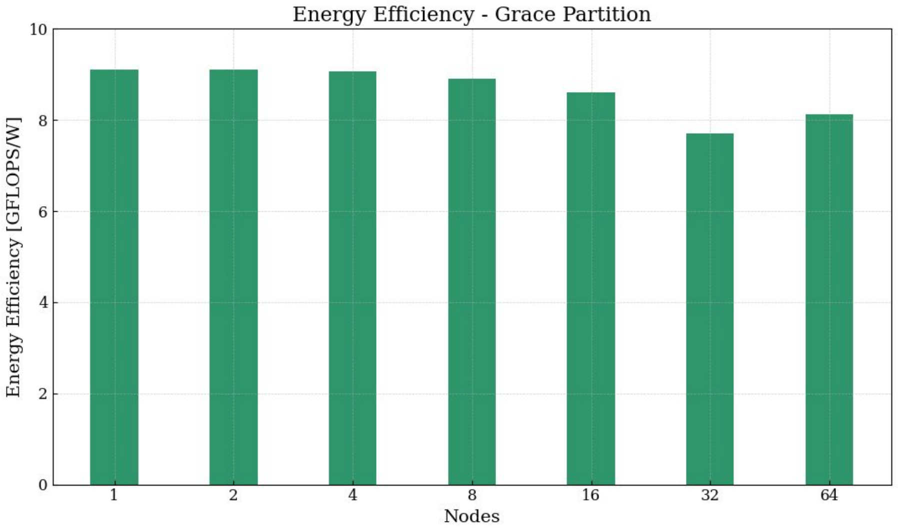
- Power: 0.59 kW
- Efficiency: 9.10 GFLOPS/W

GPP (1 node):

- Power: 0.87 kW
- Efficiency: 7.59 GFLOPS/W

ACC (1 node - CPU and GPU):

- Power: 3.50 kW
- Efficiency: 66.29 GFLOPS/W



HPCG

NGP (1 node):

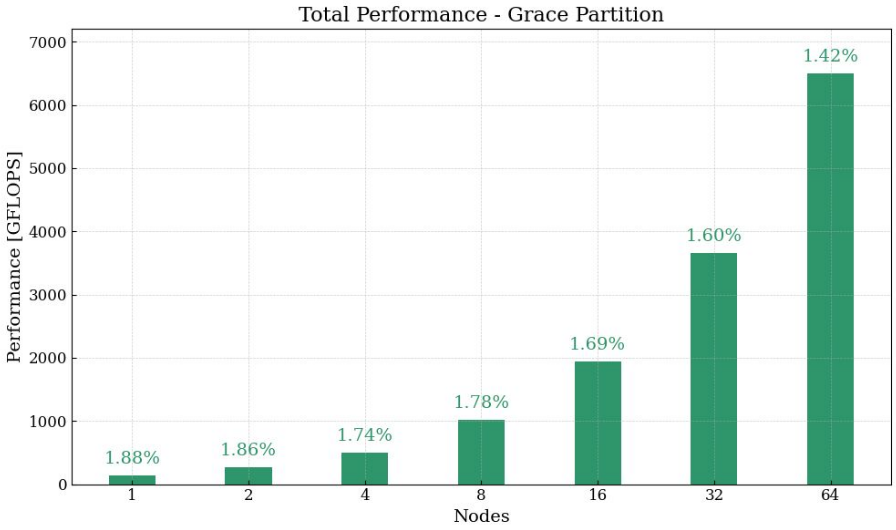
- Rpeak: 7.14 TFLOPS
- RMax: 134 GFLOPS
- %Rpeak: 1.88%
- %Memory: 10%

GPP (1 node):

- RMax: 80 GFLOPS
- %Rpeak: 1.16%

ACC (1 node):

- RMax: 1 060 GFLOPS
- %Rpeak: 0.46%



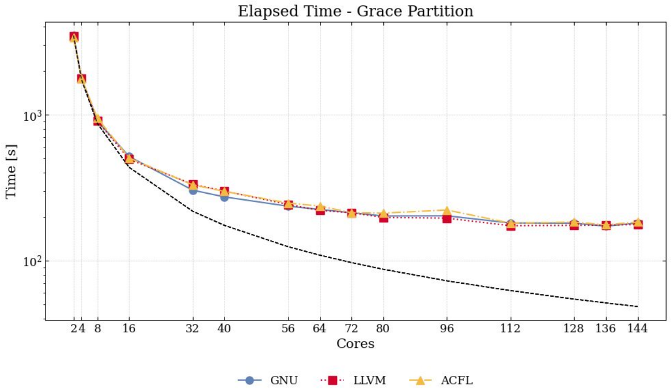
Scientific Application

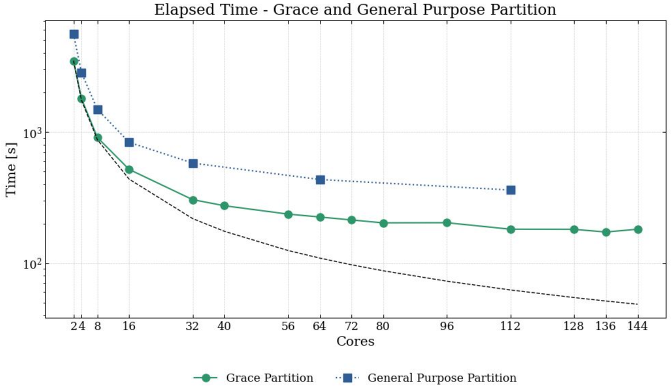
OpenFOAM - Scalability Study

- Execution time decreases as the number of CPU cores increases
- Close to ideal up to 16 cores
- Diminishing return beyond 32 cores

- NGP faster than GPP
- 1.6x faster (3 488 s instead of 5 579 s)

- Internal compiler error when using NVIDIA
- Compiler performance comparable





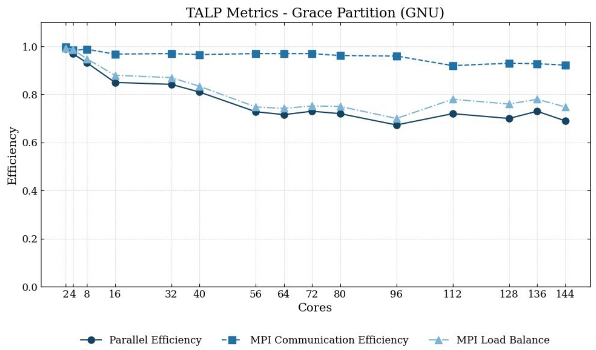
NVIDIA Grace delivers good performance out of the box

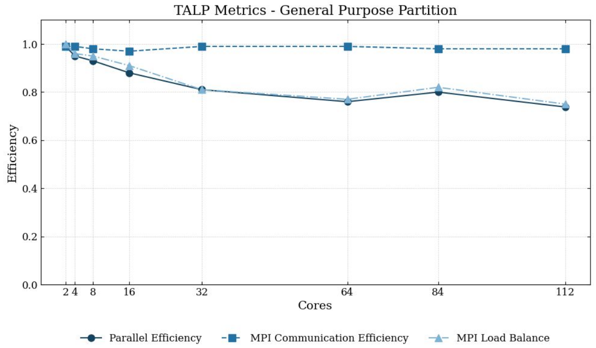
OpenFOAM - Efficiency Metrics

- MPI Communication Efficiency stays close to ideal
- Both Parallel Efficiency and Load Balance show decrease

- GPP has slightly better Load Balance, but similar Communication Efficiency

- Possible memory bandwidth saturation within 1 node



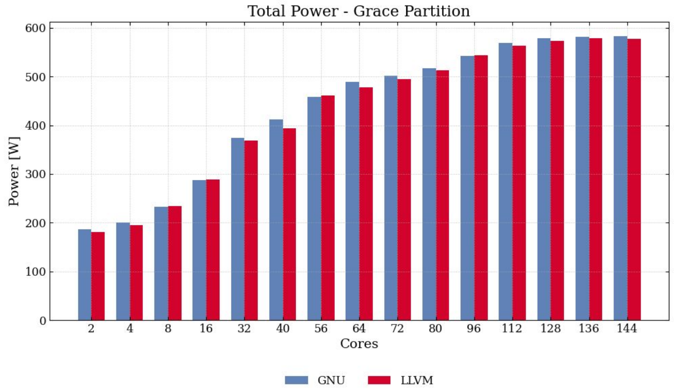


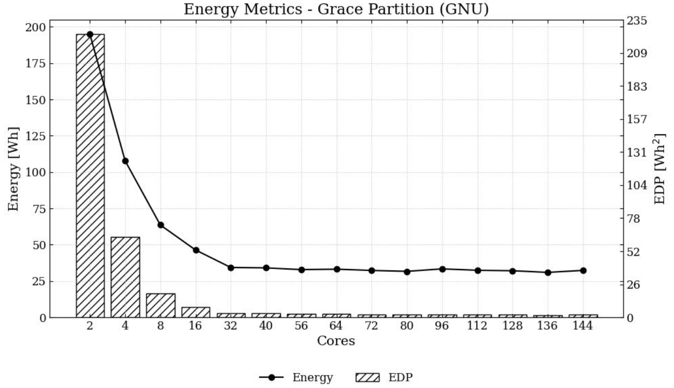
📖 NVIDIA Grace Superchip Early Evaluation for HPC Applications

OpenFOAM - Energy

- Around 200 W for 2 cores up to 600 W for a full node

- Energy decreases with increasing core count
- Energy Delay Product stagnates too
- 112 cores seems to be the optimal configuration





Conclusions

How does the NVIDIA Grace CPU compare to others?

- Significant improvements in memory bandwidth
- Often times out of the box improvements

- Underlying hardware behavior is not always fully transparent, making optimization difficult
- Software stack still exhibits minor inconsistencies or bugs

- Check compilers, runtime and flags for differences!

Next Steps

Future Work

Bindings and Flags

- Understanding their behavior better with different compilers and runtime

Hardware

- Gaining deeper insight into memory access patterns
  - Prefetching, compiler issues, NUMA Balancing, page sizes

Next Generation Accelerated cluster

- How does it compare?

Defending the thesis

MareNostrum5's Graceful landing

AHUG25, Hamburg

Majesa Trimmel

majesa.trimmel@bsc.es

Fabio Banchelli

fabio.banchelli@bsc.es

Barcelona Supercomputing Center
Centro Nacional de Supercomputación

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

European Processor Initiative