

WP2 cross cutting activity on PETSc: workflow

PETSc Codes involved: **Tandem, Lamem, pTatin3D**

Phases 1 – PETSc bottlenecks

- Review issues and bottlenecks on PETSc codes (deliverables, POP reports etc ...)
- Contact code owners for better understanding
- Identify relevant cross cutting CPU/GPU issues/bottlenecks

Phase 2 - PETSc solutions

- Contacting PETSc developers and defining solutions
- Evaluate solutions: efforts vs potential performance improvements

Phase 3 - Implementation and reporting

- Implement minor changes to codes and do some performance analysis
- Report major changes as future developments in deliverable

WP2 cross cutting activity on PETSc: workflow

PETSc Codes involved: **Tandem, Lamem, pTatin3D**

Code owners and developers: involved in the discussion through 3 phases

Phases 1 – PETSc bottlenecks

- Review issues and bottlenecks on PETSc codes (deliverables, POP reports etc ...)
- Contact code owners for better understanding
- **Identify relevant cross cutting CPU/GPU issues/bottlenecks**

Phase 2 - PETSc solutions

- Contacting PETSc developers and defining solutions
- Evaluate solutions: efforts vs potential performance improvements

Phase 3 - Implementation and reporting

- Implement minor changes to codes and do some performance analysis
- Report major changes as future developments in deliverable

Phase 1 – PETSc Performance bottlenecks (CINECA)

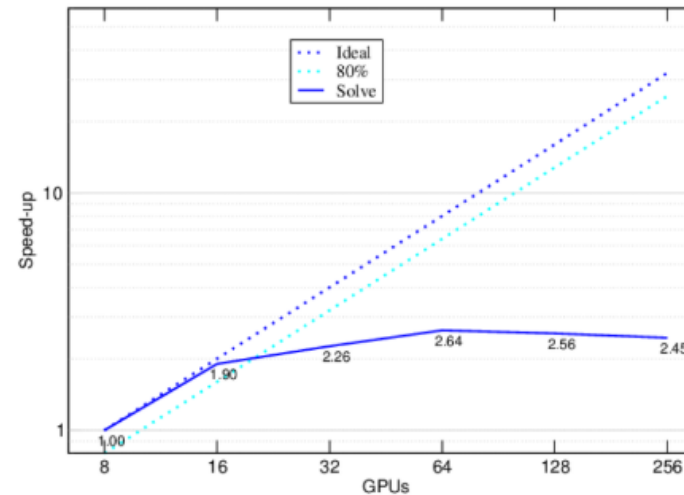
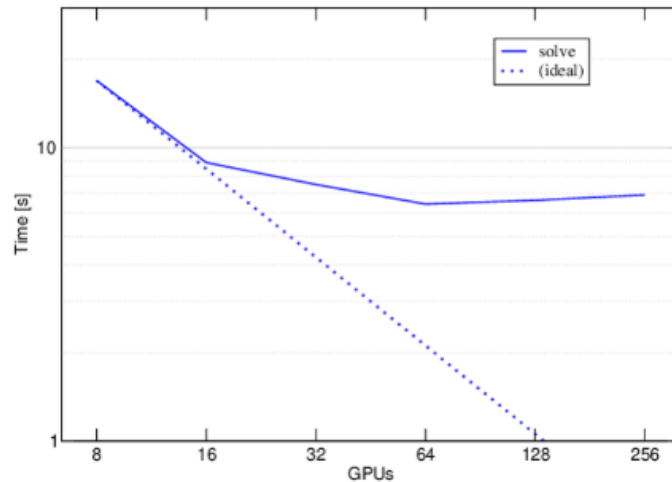
TANDEM

Review of audits, POP reports and discussion with Thomas.

Versions: ? Backend: ?

GPU performance: shows poor scaling, with efficiency decreasing beyond two nodes.

- Krylov methods: frequent synchronization (MPI_Allreduce)
- Triangular solves: ILU preconditioner performs poorly on GPUs
- Presence of many small kernels, where overhead becomes significant
- Large memory usage (due to the multigrid method?), which prevent for using more elements per GPU



Phase 1 – PETSc Performance bottlenecks (CINECA)

pTatin3D

Review of audits, POP reports and discussion with Leatitia.

Versions: ? Backend: ?

CPU performance:

- load imbalance caused by a centralized computation on the coarse multigrid solver: **solved by developing the pc_telescope feature**

GPU performance:

- increasing number of operations with the increase of GPUs, copy data back and forth between GPUs and CPUs memory?
Solved by upgrading the petsc version of the code?

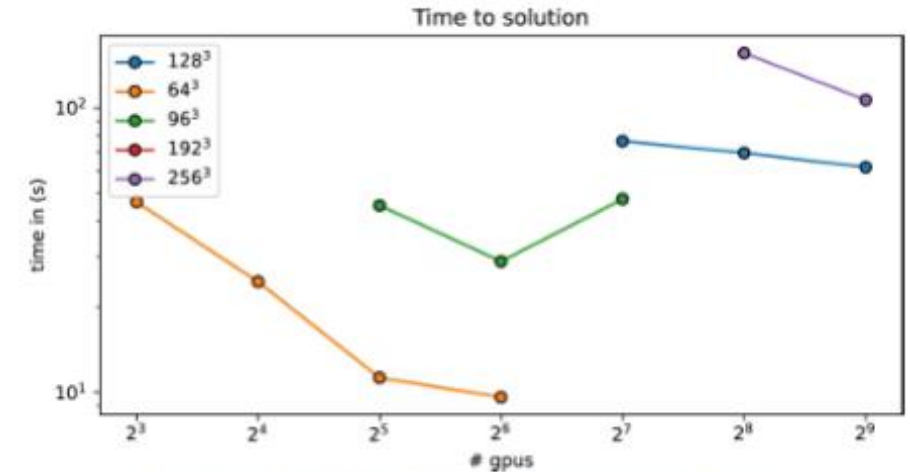


Figure 10-2 Strong scaling obtained on LUMI-G during the first audit.

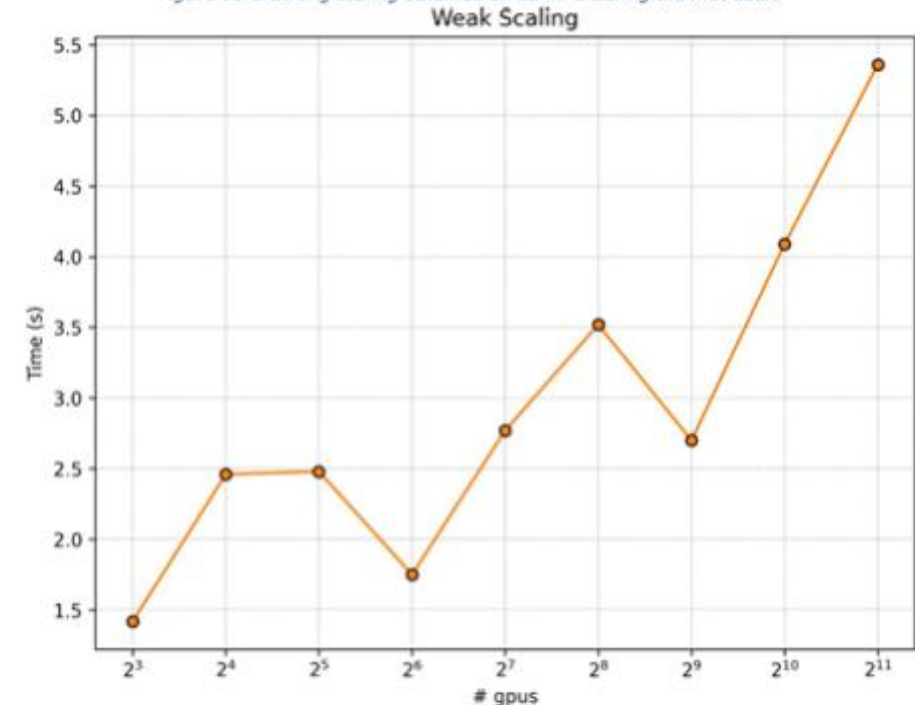


Figure 10-3 Weak scaling obtained on LUMI-G during the second audit.

Phase 1 – PETSc Performance bottlenecks (CINECA)

LAMEM

Review of audits and discussion with Boris and Anton.

Versions: ? Backend: ?

CPU performance:

- dominant costs within PCApply are matrix-vector multiplications (MatMult) and smoothing operations: **solved with PETSc's pc_telescope sub-communicator framework in combination with HYPRE multigrid solver**

GPU performance:

- Significant result: on LUMI scaling on 16384 GPUs
- observed drop in efficiency at very large GPU counts (16384) without an increase in iteration counts: **excessive host-device synchronization in PETSc? Communication intranode, internode?**
- **Weak scaling / Strong scaling? One iteration run.**

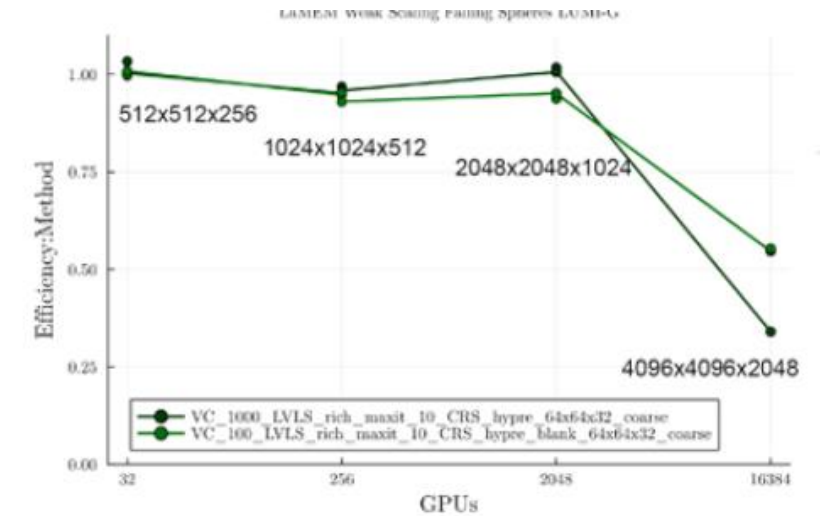


Figure 7-5 Results of full machine run of LaMEM on LUMI-G. Efficiency of the whole run per iteration.

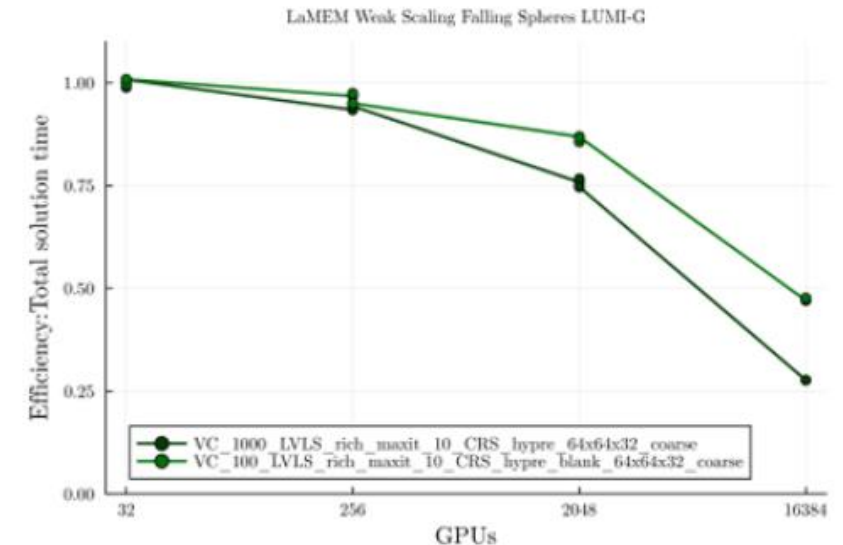


Figure 7-6 Results of full machine run of LaMEM on LUMI-G. Efficiency of the whole run.

Phase 1 – PETSc Performance bottlenecks (CINECA)

CONCLUSIONS

Cross-cutting issues for CPU:

- multi-grid solver scalability issues: solved with pc_telescope framework for LaMEM and pTantini3D.
- TANDEM could benefit from it?

Cross-cutting issues for GPU:

- Excessive communication between GPU-CPU identified in the three codes? -> to investigate
- Tandem choices of solver and preconditioner not adapted for GPUs? -> to investigate

Next steps:

- Discussing with all code-owners about possible worthwhile (effort vs potential) investigation
- Contacting Tandem developers team
- Contacting PETSc developers team