PRACE Preparatory Access Type A

Final Report

Peer-Review Office – V0.1 – 22/11/2018

1 General Information

Type of project granted: Preparatory Access Type A – Code scalability and performance.

Tests to obtain the relevant parameters necessary when applying to future PRACE calls for Project Access and for EuroHPC calls for Regular Access.

1.1 Proposal ID

2010PA6132

1.2 Period of access to the EuroHPC facilities

October 2021 – January 2022

1.3 Name of the EuroHPC facility assigned

Vega CPU Standard and Karolina CPU

2 Project information

2.1 Project name to which the tested code corresponds

Scalability of a Large Eddy Simulation solver on Vega and Karolina clusters

2.2 Research field

- ☐ Biochemistry, Bioinformatics and Life sciences
- ☐ Chemical Sciences and Materials
- ☐ Earth System Sciences
- ☐ Economics, Finance and Management
- ☐ Fundamental Physics
- ☐ Linguistics, Cognition and Culture
- ☐ Mathematics and Computer Sciences
- ☐ Physiology and Medicine

☒ Engineering

☐ Universe Science

☐ Fundamental Constituents of Matter

2.3 Institutions and research team members

Riccardo Broglia, CNR-INM, Institute of Marine Engineering, National Research Council of Italy

Antonio Posa, CNR-INM, Institute of Marine Engineering, National Research Council of Italy

2.4 Summary of the project interest

The present project is aimed at demonstrating the scalability of an in-house, academic Large-Eddy Simulation solver in Fortran language on both Vega and Karolina CPU partitions, in order to provide evidence of its portability and suitability for future production runs on those systems. In particular, we aim at identify the best option for our code, to be considered for our future proposals in the framework of regular EuroHPC calls. Finite-differences are utilized to discretize the filtered Navier-Stokes equations. An immersed-boundary methodology enables the use of regular grids, as Cartesian or cylindrical, making the decomposition of the overall flow problem into subdomains very straightforward, efficient and suitable to parallel computing. Communications across subdomains are handled via calls to MPI libraries. I/O operations are performed using calls to parallel HDF5 libraries. The solver is not I/O intensive, with I/O operations taking only about 5% of the overall computational cost of a typical simulation. The computational grid we will consider in this project is a cylindrical one, composed of about 5 billion points. Although the scalability of the present solver was already tested on several architectures, also part of the PRACE infrastructure (Marconi KNL, Joliot-Curie KNL, Joliot-Curie SKL, Joliot-Curie Rome), the test-case that will be considered in this project was specifically designed to be representative of the computational effort of the problem we aim to tackle in the framework of the upcoming EuroHPC calls for regular projects. Results of these tests will be included in the proposal we are going to submit in the next future to EuroHPC when asking for allocation of computational resources on Vega and/or Karolina clusters.

3 Main features of the code

3.1 Name of the code

Eddy (in house Large-Eddy Simulation Fortran solver)

3.2 Type of the code distribution

In-house academic solver

3.3 Computational problem executed

Filtered Navier-Stokes equations

3.4 Computational method

Finite-differences, fractional-step and immersed-boundaries

3.5 Kind of parallelism used

MPI

3.6 Main libraries used, version and language. Did you use the /usr/local one?

MKL, OpanBLAS, ScaLAPACK, HDF5 already available on both Vega and Karolina via modules.

On Vega CPU Standard:

- HDF5/1.12.0-iimpi-2021a
- oneapi/mkl/2021.2.0

On Karolina CPU:

- HDF5/1.10.7-iimpi-2020b
- OpenBLAS/0.3.12-GCC-10.2.0
- ScaLAPACK/2.1.0-gompic-2020b

3.7 Which other software did you use on the PRACE machines? Did you use some post-processing or pre-processing tools?

No additional software was required for testing the scalability of the solver.

4 Compilation step

4.1 How is the program compiled?

A makefile was utilized.

4.2 Difficulties met to compile, if any, and how they were tackled.

Compilation was straightforward, thanks to the use of modules.

4.3 Which version of the compiler and version of the MPI library did you use?

On both Vega CPU Standard and Karolina CPU:

- intel-compilers/2021.2.0
- impi/2021.2.0-intel-compilers-2021.2.0

4.4 Did you use any tools to study the behaviour of your code?

We utilized timers already implemented within our solver.

5 Execution step

5.1 How is the program launched?

The executable file was launched using scripts (for SLURM on Vega and PBS on Karolina).

5.2 Difficulties met to launch the code, if any, and how they were tackled.

None.

6 Communication patterns

If you know which are the main communication patterns used in your code configuration, select the ones from the mentioned below:

☒ Few point to point communications

☒ Few collective communications

☐ Barrier

☒ Reduction

☒ Broadcast

☐ Scatter/gather

☐ All to all

7 Results of the scalability testing

7.1 Summary of the obtained results from the scalability testing

The images reported in Sec. 7.2 show results about the scalability performance of our solver on both Vega CPU Standard and Karolina CPU. Tests were conducted assuming as a reference a problem, dealing with a theoretical submarine geometry, consisting of a cylindrical grid of 5 billion points. The purpose of these tests was: i) to demonstrate the suitability of our solver to run on Vega and Karolina clusters in the framework of EuroHPC Regular Access; ii) to define the amount of computational resources we need on those clusters to carry out our simulations through EuroHPC Access. Overall, the results of our tests demonstrated that, for the particular size of the problem, the best option is running on 4,096 cores on both Vega and Karolina. Up to that number of cores scaling was verified almost linear, especially on Vega, which was eventually selected for requesting resources in the framework of the coming cut-off date of the EuroHPC call for Regular Access. Further increasing the number of cores, the relative cost of the communications across subdomains resulted in a decline of the scaling performance.

7.2 Images or graphics showing results from the scalability testing

Results of strong scaling tests on Vega CPU Standard are reported in ***Figure 1***. They show almost linear scaling from 256 cores (2 nodes) up to 4,096 cores (32 nodes). Smaller core counts were not allowed by memory limitations. The arrow in ***Figure 1*** indicates the selected number of cores for the computations we scheduled in the framework of EuroHPC Regular Access.

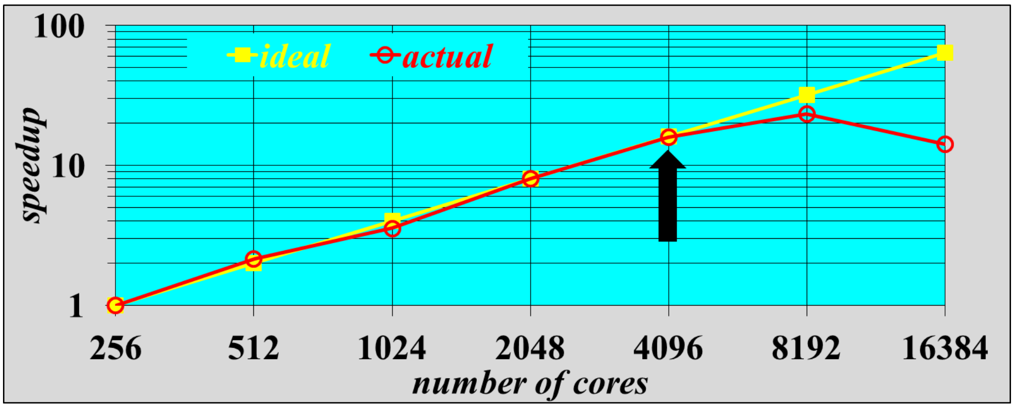

Figure 1. Strong scaling tests on Vega CPU Standard

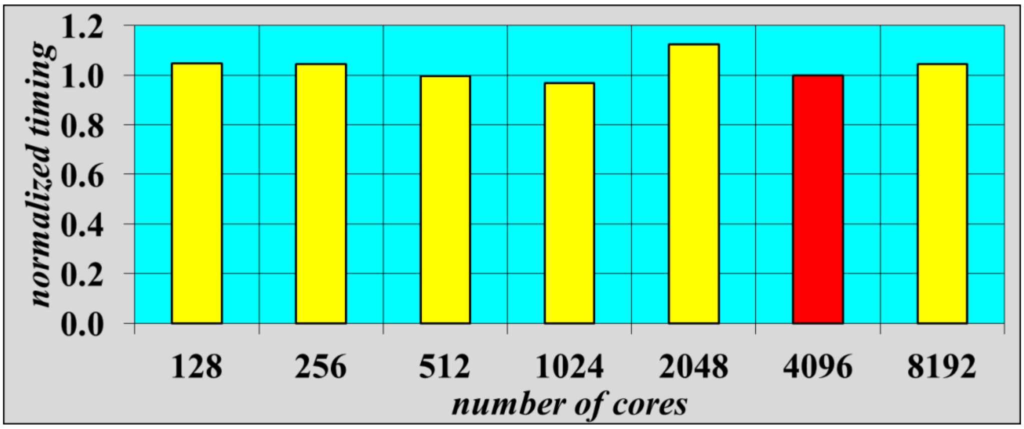

Figure 2. Weak scaling tests on Vega CPU Standard

Results of weak scaling tests on Vega CPU Standard are shown in ***Figure 2***, ranging between 128 cores (1 node) to 8,192 cores (64 nodes). The column in red indicates the reference case of the computational problem, consisting of a cylindrical grid of 5 billion points, running on 4,096 cores. ***Figure 2*** highlights that weak scaling on Vega CPU Standard is also excellent.

Strong scaling was demonstrated up to 4,096 cores also on Karolina CPU, as shown in ***Figure 3***.

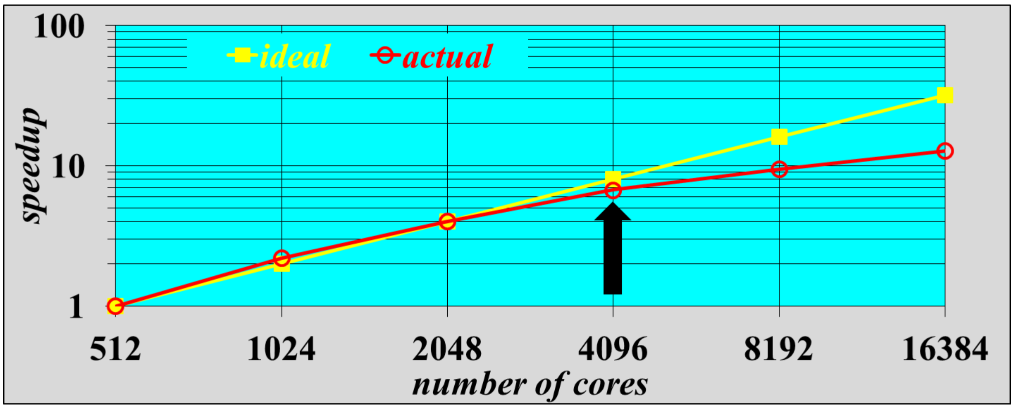

Figure 3. Strong scaling tests on Karolina CPU

Also weak scaling on Karolina was found satisfactory (***Figure 4***), although demonstrating a little more dependence on the size of the computational problem, in comparison with Vega.

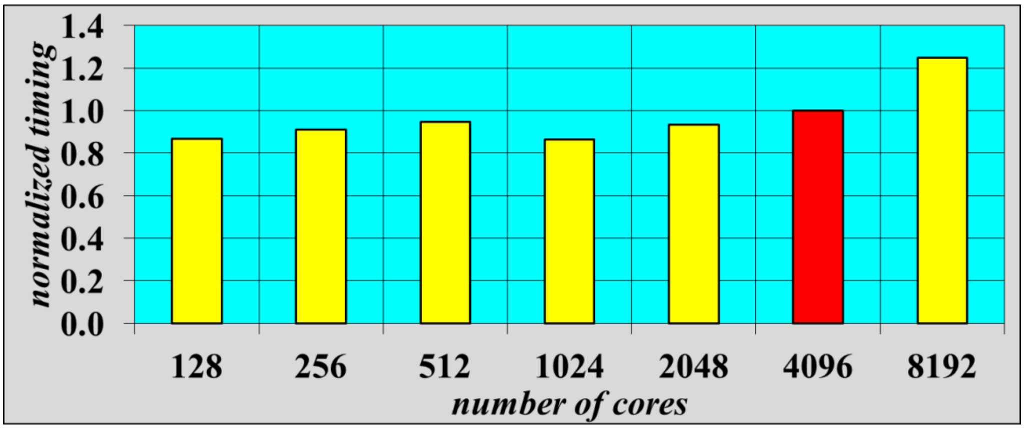

Figure 4. Weak scaling tests on Karolina CPU

7.3 Data to deploy scalability curves

A) Some typical user test cases

Vega CPU Standard

Number of cores	Wall clock time (s/step)	Speed-up vs the first one	Number of Nodes	Number of process
1,024	64.04	1.00	8	1,024
2,048	28.20	2.27	16	2,048

4,096	14.26	4.49	32	4,096

Karolina CPU

Number of cores	Wall clock time (s/step)	Speed-up vs the first one	Number of Nodes	Number of process
1,024	36.07	1.00	8	1,024
2,048	19.69	1.83	16	2,048
4,096	11.75	3.07	32	4,096

B) Strong scaling curve

Vega CPU Standard

Number of cores	Wall clock time (s/step)	Speed-up vs the first one	Number of Nodes	Number of process
256	226.81	1.00	2	256
512	105.68	2.15	4	512
1,024	64.04	3.54	8	1,024
2,048	28.20	8.04	16	2,048
4,096	14.26	15.90	32	4,096
8,192	9.70	23.39	64	8,192
16,384	16.05	14.13	128	16,384

Karolina CPU

Number of cores	Wall clock time (s/step)	Speed-up vs the first one	Number of Nodes	Number of process
512	78.80	1.00	4	512
1,024	36.07	2.18	8	1,024
2,048	19.69	4.00	16	2,048
4,096	11.75	6.70	32	4,096
8,192	8.40	9.38	64	8,192
16,384	6.15	12.81	128	16,384

C) Weak scaling curve

Vega CPU Standard

Number of cores	Wall clock time (s/step)	Speed-up vs the first one	Number of Nodes	Number of process

128	14.95	1.00	1	128
256	14.90	1.00	2	256
512	14.23	0.95	4	512
1,024	13.80	0.92	8	1,024
2,048	16.05	1.07	16	2,048
4,096	14.26	0.95	32	4,096
8,192	14.88	1.00	64	8,192

Karolina CPU

Number of cores	Wall clock time (s/step)	Speed-up vs the first one	Number of Nodes	Number of process
128	10.18	1.00	1	128
256	10.71	1.05	2	256
512	11.11	1.09	4	512
1,024	10.15	1.00	8	1,024
2,048	10.96	1.08	16	2,048
4,096	11.75	1.15	32	4,096
8,192	14.67	1.44	64	8,192

7.4 Publications or reports regarding the scalability testing

The results of the scalability tests were included in our recent proposal submitted to the EuroHPC call for Regular Access "*Wake analysis of a tip loaded marine propeller using high-fidelity Large-Eddy Simulations (PROPLES)*".

8 Results on Input/Output

8.1 Size of the data and/or the number of files

Tests were also conducted to verify the ability of the I/O subroutines implemented within our solver of working properly on both Vega and Karolina clusters. This involved I/O operations on a small number of files (5 for each run) having a size of about 40GB each. However, since the purpose of the project was demonstrating scalability, those files were generated for testing the I/O subroutines only and then immediately removed from both systems. A few additional small files (orders of Mbytes across a few tens of files) were also generated during each run for profiling and diagnostics. Therefore, the overall size of the data generated as a result of the tests conducted on both Vega and Karolina is of the order of a few hundreds Mbytes.

8.2 Please, let us know if you used some MPI-IO features

I/O operations were conducted using calls to parallel HDF5 libraries.

9 Main results

Our solver was verified ready to run on both Vega CPU Standard and Karolina CPU. Scaling performance was demonstrated almost linear up to 4,096 cores. This number of cores was verified the most suitable for the size of the problem we will tackle through the simulations in our recent proposal to the EuroHPC call for Regular Access "*Wake analysis of a tip loaded marine propeller using high-fidelity Large-Eddy Simulations (PROPLES)*". The results on the present tests were also fundamental to provide to EuroHPC an accurate estimate of the amount of core hours we need on Vega CPU Standard to complete all planned simulations (25 million).

10 Feedback and technical deployment

10.1 Feedback on the centres/EuroHPC mechanism

We found EuroHPC Benchmark and Development very useful to demonstrate scalability and portability of our solver as well as to select the most suitable machine where to conduct our computations in the framework of EuroHPC Regular Access. In addition, scaling tests were also important to define the optimal size of our computations and the allocation of core hours needed to complete all. Unfortunately, for the first cut-off date of the EuroHPC call for Regular Access we had the opportunity to tests our solver on Vega and Karolina only, since the other EuroHPC clusters currently available through Regular Access were still not available through EuroHPC Benchmark and Development.

10.2 Explanation of how the computer time was used compared with the work plan presented in the proposal. Justification of discrepancies, especially if the computer time was not completely used.

We were able to complete all planned scaling tests within a few days on both Vega and Karolina. All information needed for selecting the most suitable cluster and size of the computational problem and defining the amount of computational resources needed for our future production runs were generated within a week from getting access to both Vega and Karolina.

10.3 Please, let us know if you plan to apply for PRACE Project Access or EuroHPC Regular Access in the future? If not, explain us why.

Yes. Scaling tests were actually conducted for the preparation of the proposal we recently submitted to the EuroHPC call for Regular Access (cut-off date on 3 December 2021). In the future we plan to keep applying both for PRACE Project Access and EuroHPC Regular Access, taking also advantage of the resources made available through PRACE Preparatory Access and EuroHPC Benchmark and Development. The latter were demonstrated very useful to us both for the selection of the cluster where to run our production runs and for fine-tuning of our proposals to both PRACE and EuroHPC regular calls.