Developing Code_Saturne for Multi-Billion Cell Mesh Simulations

Charles Moulinec[1], Yvan Fournier[2], Juan C. Uribe[3], Pascal Vezolle[4], David R. Emerson[1]

[1]STFC Daresbury Laboratory, Sci-Tech Daresbury, Keckwick Lane, Daresbury, Warrington, WA4 4AD, UK
[2]EDF R&D, 6 quai Watier, 78400 Chatou, FR
[3]EDF R&D UK Centre, Modelling and Simulation Centre, The University of Manchester, Manchester, M13 9PL, UK
[4]IBM France, La Pompignane, CS 81021, 34060 Montpellier, FR

Introduction

Computing Large-Eddy Simulations (LES) of a full power plant reactor would require about 13 to 15 billion cells. Some algorithms designed to reach this target are presented for Code_Saturne [1, 2], which is a highly scalable multi-purpose CFD software developed by EDF-R&D since 1997 and open source since 2007 (GPL license). The code is based upon a co-located finite volume approach that can handle three-dimensional meshes built with any type of cell and with any type of grid structure. MPI is used for communication. The code is able to simulate either incompressible or weakly compressible flows, with or without heat transfer, and has a variety of turbulence models (see Fig. 1 for the toolchain).

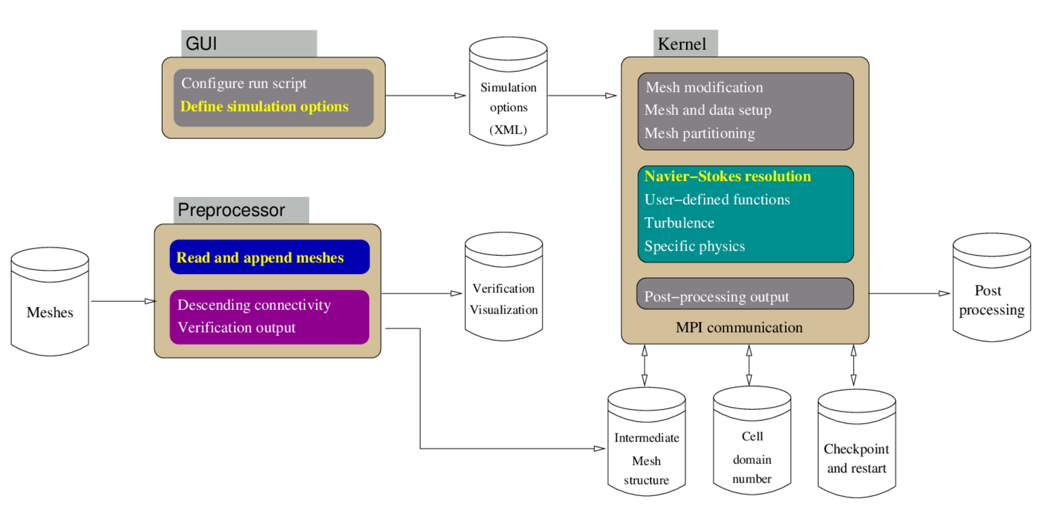

Figure 1: Code_Saturne's toolchain

The test cases consist of LES in staggered-distributed tube bundles (see Fig. 2). The configuration is the one experimentally studied by Simonin and Barcouda [3], and the mesh is obtained by copying and shifting several elemental patterns made of a whole tube in its centre and four quarter of tubes in the corners. The elemental pattern contains about 13 million cells (2-D cross-section: 100, 040 cells; 3^{rd} direction: 128 layers) and meshes of 51 (51M), 204 (204M), 816 (816M) million and 3.2 billion (3.2B) cells are used.

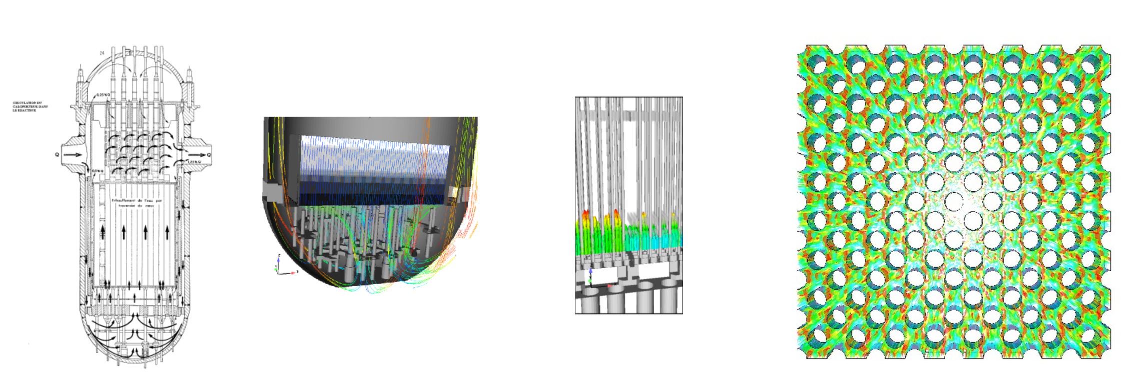

Figure 2: Location of the bundle of tubes in the reactor

Hardware - Software

	HECToR	Jaguar
Architecture	XE6	XK6
Processor	32-Core AMD	16-Core AMD
Nodes	2,816	18,688
Cores/Nodes	32	16
Memory/node	32GiB	32GiB
Memory/core	1GiB	2GiB
Compiler	GNU 4.6.3	PGI 12.1.0
MPI	MPICH2 5.4.5	MPICH2 5.4.1

Mesh Joining - Periodicity

Meshes are generated by joining parts of existing meshes. Each new mesh is built from 4 old meshes. Periodicity is handled as mesh joining. Table 1 shows that the time to complete the overall joining (3.2B case) is similar on both machines.

	HECToR	Jaguarpf
8192	95	127
16384	146	178
32768	400	414
65536	-	1630

Table 1: Time for overall joining in seconds (3.2B case)

Partitioning - Halos

A Space-Filling algorithm based on Morton Curves (SFC) and ParMetis-4.0.2 (ParM), used as partitioners are run with the number of cores as the number of sub-domains. Partitioning and computing halos take less than 2 minutes on 65536 cores for the 3.2B case on Jaguar (see Table 2).

	Partitioning				Halos			
	HECToR		Jaguar		HECToR		Jaguar	
	SFC	ParM	SFC	ParM	SFC	ParM	SFC	ParM
8192	21.7	45.6	14.8	14.3	10.6	8.2	11.1	8.7
16384	28.2	90.7	9.7	16.1	9.5	8.3	8.9	7.8
32768	53.5	181.0	16.4	40.8	22.6	21.2	18.7	17.7
65536	-	-	57.6	94.1	-	-	81.3	63.1

Table 2: Time for partitioning in seconds (3.2B case)

IO Performance

Two strategies to handle IOs are available in the code, MPI-IO and a serialisation method [2]. Table 3 shows that MPI-IO is faster for the 816M on the HECToR's Lustre file system.

	MPI-IO	Ser-IO
2048	633	1203
4096	608	640
8192	859	1147
16384	732	747

Table 3: Time for IOs in seconds (816M case) on HECToR

Solver

The Navier-Stokes solver is segregated. The pressure is solved by an Algebraic Multigrid algorithm (V-cycle), with the Conjugate Gradient as a solver. The Jacobi algorithm is used for the velocity components.

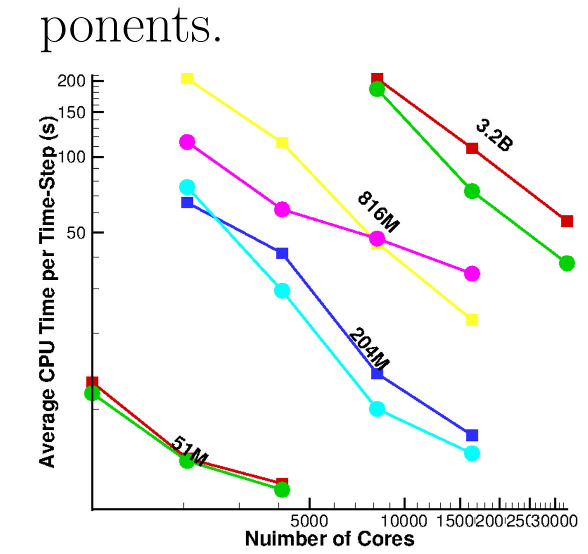

Figure 3: Solver performance for various mesh sizes (HECToR)

All cases but the 816M one show that the solver exhibits better results with SFC than with ParM as a partitioner (see Fig. 3) when run on HECToR.

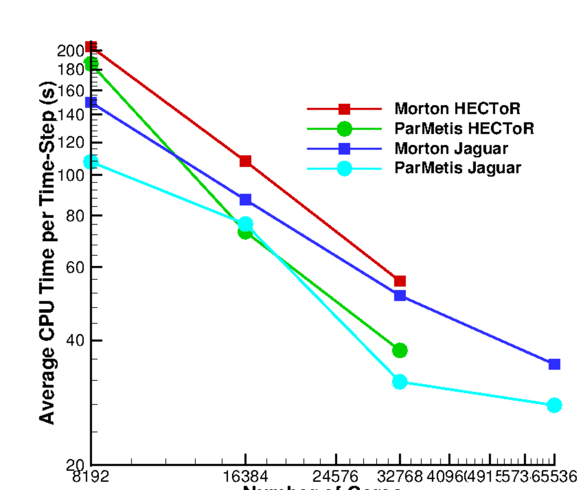

Figure 4: Solver performance comparison for the 3.2B cell mesh (HECToR and Jaguar)

The 3.2B case is used for comparison on both machines, where the average CPU time per time-steps decreases as a function of the number of cores, for both partitioners. A speed-up of 1.46 (resp. 1.14) is still observed for SFC (resp. ParM) on Jaguar, going from 32768 to 65536 cores.

Conclusions - Perspectives

The time spent in the algorithms used in Code_Saturne for mesh joining, periodicity, partitioning, halo builing have proven to be modest in comparison with the IO and solver time on Cray machines with Lustre file system, especially for a 3.2 billion cell mesh.

More testing will be performed on new architecture/file sytems such as IBM Blue Gene/GPFS for instance, to assess the performance of the software.

Acknowledgements

This research used resources of the Oak Ridge Leadership Computing Facility located in the Oak Ridge National Laboratory, which is supported by the Office of Science of the Department of Energy under Contract DE-AC05-00OR22725. The authors also thank the UK Engineering and Physical Sciences Research Council (EPSRC) for their support of Collaborative Computational Project 12 (CCP12).

References

[1] F. Archambeau, N. Mechitoua and M. Sakiz. Int. J. on Finite Volumes, **1**, (2004) pp. 1-62.

[2] Y. Fournier, J. Bonelle, C. Moulinec, Z. Shang, A. G. Sunderland and J. C. Uribe. Comp. & Fluids, **45**, Issue 1, (2011), pp. 103-108.

[3] O. Simonin and M. Barcouda. 4^{th} Int. Symp. on Apps of Laser Anemometry to Fluid Mechanics. (1988).

[4] Y. Fournier, J. Bonelle, P. Vezolle, J. Heyman, B. D'Amora, K. Magerlein, J. Magerlein, G. Braudaway, C. Moulinec, A. G. Sunderland. Conc. and Comp. - Practice and Experience, (2012) DOI: 10.1002/cpe.2852.