Contents lists available at SciVerse ScienceDirect

Computers & Fluids

journal homepage: www.elsevier.com/locate/compfluid

TELEMAC: An efficient hydrodynamics suite for massively parallel architectures ☆

C. Moulinec [a,*], C. Denis [b], C.-T. Pham [c], D. Rougé [c], J.-M. Hervouet [c], E. Razafindrakoto [c], R.W. Barber [a], D.R. Emerson [a], X.-J. Gu [a]

[a] *STFC Daresbury Laboratory, Warrington WA4 4AD, Cheshire, UK*
[b] *EDF R&D, SINETICS Department, 1 avenue du Général de Gaulle, 92140 Clamart, France*
[c] *EDF R&D, LNHE, 6 quai Watier, 78400 Chatou, France*

ARTICLE INFO

Article history:
Received 27 August 2010
Received in revised form 8 July 2011
Accepted 10 July 2011
Available online 23 July 2011

Keywords:
Hydrodynamics
High Performance Computing
Open-source
Round-off errors
TELEMAC

ABSTRACT

This paper investigates the use of TELEMAC (a Finite Element-based hydrodynamics suite) on massively parallel computer architectures. The performance of TELEMAC is illustrated using two separate test cases. The first considers the use of TELEMAC-2D for simulating tidal currents in the vicinity of a renewable energy marine turbine farm, in order to provide reliable estimates of the expected energy yield. The second demonstrates the use of TELEMAC-3D for assessing the effects of fresh water discharges on the salinity distribution in a coastal lagoon. The simulations have been performed with meshes ranging from 2 to 12 million elements, and good scaling performance is achieved on a variety of different computer architectures.

1. Introduction

The Computational Fluid Dynamics (CFD) software suite, TELEMAC [1,2], is a powerful integrated modelling tool for simulating offshore, coastal and river systems, and shallow lagoons and estuaries. The software is able to model free-surface flows, including flooding and drying effects, and discharges of pollutants or freshwater. The suite has been continuously developed by EDF R&D for over 20 years. The main codes within the TELEMAC suite include TELEMAC-2D, which uses the depth-integrated shallow water (hydrostatic) equations to describe the conditions when the horizontal length scale of the flow is greater than the vertical scale, and TELEMAC-3D, where the full Navier–Stokes (or non-hydrostatic) equations are solved [1]. Additional modules, such as SISYPHE for modelling sediment transport, are also available but will not be considered in the present study. Each package is based on a Finite Element approach. In TELEMAC-3D, the 2-D bottom surface is meshed by triangles, with extrusion layers representing the three-dimensional geometry. A σ transform is used to capture the variation in the free-surface elevation. In both TELEMAC-2D and TELEMAC-3D, the algorithms and parallel code capabilities are provided by the integrated BIEF library. Communication is performed using MPI, and grid partitioning is carried out using METIS [3]. Most of the TELEMAC suite has been available as open-source software since July 2010, including TELEMAC-2D, SISYPHE (sediment transport), TOMAWAC (wave propagation) and ARTEMIS (wave agitation in harbours). In addition, TELEMAC-3D will be available as open-source software by summer 2011.

The objective of this work is to demonstrate that both TELEMAC-2D and TELEMAC-3D, which are currently used on relatively coarse grids (typically involving a few hundred thousand elements), are ready for more challenging simulations involving much higher resolution grids. The paper will describe the improvements that are necessary to get the best from High Performance Computing (HPC), and will illustrate the performance of TELEMAC on several high-end platforms.

2. Improvement of TELEMAC for HPC applications

The TELEMAC system parallelism is based on domain decomposition. The 2-D Finite Element mesh is partitioned with no subdomain overlapping. This is performed using a utility called PARTEL which uses METIS [3] for the partitioning. The original version of TELEMAC was not designed for large-scale HPC applications but for simulations having the order of 2–32 subdomains, with around ten thousand elements per subdomain. For simulations involving thousands of cores, the preprocessing requires too much computing time; for example, it takes ~10 h to preprocess a 2-million finite element mesh into 8192 subdomains.

We have recently improved PARTEL by modifying the algorithms to reduce the computing time. These changes have

☆ This paper was originally submitted to the ParCFD 2010 special issue which was published in Volume 45, Issue 1 of Computers and Fluids.
* Corresponding author.
E-mail address: charles.moulinec@stfc.ac.uk (C. Moulinec).

0045-7930/$ - see front matter Crown Copyright © 2011 Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.compfluid.2011.07.003

substantially reduced the preprocessing time from the previous 10 h to about 200 s. However, as auxiliary arrays are used, PARTEL requires more memory and this increase is expected to become a bottleneck for extremely large meshes (i.e. >25 million elements). To overcome this potential limitation, a parallel version of PARTEL is currently being developed that will distribute the memory across the processors and will increase the parallel data flow of the TELEMAC suite.

3. HPC and numerical validation

Scientific computation has intrinsic unavoidable approximations. The model development process, from the physical world to the mathematical model, then to the computational model and finally to the computer implementation, involves a number of approximations. For example, certain physical effects may need to be neglected, continuous functions are replaced by discretised ones and real numbers have to be replaced by finite precision representations. One important source of error, that is both esoteric and difficult to manage, originates from the use of finite precision arithmetic. It is well known that floating-point arithmetic commonly used in scientific computing only approximates exact arithmetic. Arithmetic expressions are no longer associative, commutative or distributive. More importantly, the evaluation of most arithmetic expressions generates round-off errors. It is not uncommon to find that the same code, using the same data, produces different results when executed on different platforms. Recently, Goel and Dash [4] have demonstrated the numerical differences obtained by running the same weather prediction code, using the same input data, on three different computer architectures. For example, the average difference in temperature at the 500 hPa geopotential height was in the range of (−2 °C, +2 °C) after integrating the model over a 4 month timeframe. There is obviously a need to perform regular numerical checks on scientific codes in order to detect the undesirable effect of round-off error propagation.

Several methods and tools have been developed over the years to analyse round-off error propagation. These include direct analysis, inverse analysis [5], methods based on interval arithmetic [6] and randomised interval arithmetic [7]. Compared to other tools, the CADNA (Control of Accuracy and Debugging for Numerical Applications) library [8] appears less intrusive on the original code. It relies on the implementation of discrete stochastic arithmetic (DSA), which is based on the CESTAC (Contrôle et Estimation STochastique des Arrondis de Calculs) method, and is specifically designed for programs written in ADA, C, C++ and Fortran. Where no overflow occurs, the exact result, r, of any non exact floating-point arithmetic operation is bounded by two consecutive floating-point values R^{-} and R^{+}. The basic idea of the method is to perform each arithmetic operation N times, randomly rounding each time, with a probability of 0.5, to R^{-} or R^{+}. A typical value of N is 3. The computer's deterministic arithmetic is therefore replaced by stochastic arithmetic where each operation is performed N times before the next operation is executed, thereby propagating the round-off error differently each time.

Studying round-off error propagation provides an important numerical health check that will give additional confidence in the computed results, which is especially important for mission-critical industrial codes. This problem is exacerbated in today's super-computing environment where trillions of floating-point operations may be performed every second. A parallel program based on domain decomposition, as shown in this paper, could compute slightly different results depending on the number of subdomains. The communication scheme of parallel codes in the TELEMAC suite has to gather local data at all interface nodes with the same global position but belonging to different processors. At the end of each communication step, interface nodes with the same global position must have the same value for each contiguous subdomain. However, this does not always happen due to floating point summation round-off errors. In the case of an interface node shared by four subdomains, for instance, each subdomain receives a value from the three other subdomains during the communication step. These values are successively added to the local value. Each subdomain could compute these sums in parallel in a different order depending on the communication network. Unfortunately, as this involves floating point computations, the result of the sums could be different for each subdomain due to round-off errors since the floating point sum is not associative. To circumvent this problem, one possible solution consists of assigning the *maximum* value of the sums among the subdomains at each interface node. The main drawback of this approach is that the communication volume increases. Moreover, the problem of round-off errors still exists but is effectively hidden. Indeed, the maximum value of the sum does not necessarily correspond to the approximation of the sum with no round-off errors. The communication phase therefore needs to be modified so as to compute the sum in the same order, independent of the number of subdomains. A simple method is to always compute the sum in the ascending order of the MPI processes. This modification avoids having to compute any maximum, and yields a gain of 2.5% (1024 subdomains) to 9% (2048 subdomains) in computing time on the EDF IBM Blue Gene/P [9]. Even if the modified communication scheme ensures the consistency in the values at each interface node belonging to different subdomains, problems associated with round-off errors still remain.

The CADNA library has been implemented in the TELEMAC-3D communication scheme to assess the precision of the floating point summation. The test case consists of a finite element mesh with 6,352,954 grid points and 12,083,160 finite elements. The number of time steps has been set to 1000. For this exercise, the test case has been run from 1024 to 8192 processors on a Blue Gene/P. Table 1 represents the percentage of summations having *nsd* significant digits for the selected communication scheme. Approximately 400,000 summations are performed during the simulation. The number of instabilities does not increase with the number of subdomains. In a large number of cases (more than 95% of the number of sums), there is no round-off error: the contributions to be summed having the same order of magnitude. However, the number of significant digits decreases when the order of magnitude of the operands differs.

One of the short-term objectives is to implement and evaluate error-free transformation for the sum of two floating point numbers [10] using the CADNA library. At this stage, it is not known whether the loss of precision in the summations will have any negative effects on the final results of a simulation. It will obviously depend on the number of significant digits lost by the successive numerical steps. In the near future, each subroutine of the TELEMAC system will be linked to the CADNA library in order to analyse the error propagation behaviour.

4. Hardware

4.1. Cray XT4 (HECToR, Phase 2a [11])

The Cray XT4 comprises 1416 compute blades, each of which has four quad-core processor sockets. This amounts to a total of 22,656 cores, each of which acts as a single CPU. The processor is an AMD 2.3 GHz Opteron. Each quad-core socket shares 8 GB of memory, giving a total of 45.3 TB over the whole XT4 system. The theoretical peak performance of the system is 208 Tflop/s. There are 24 service blades, each with two dual-core processor sockets; these act as login nodes and controllers for I/O and for

Table 1
Number of significant digits obtained during a test summation using different numbers of subdomains.

nsd	Ascending order of MPI processes Percentage of summations having *nsd* significant digits (%)
1024 subdomains	
15	97.1
14	2.68
13	0.23
12	0.03
11	0.003
10	0.0002
2048 subdomains	
15	96.7
14	2.96
13	0.3
12	0.03
11	0.004
10	0.0002
4096 subdomains	
15	95.6
14	3.8
13	0.91
12	0.04
11	0.002
10	0.0002
8192 subdomains	
15	97.4
14	2.23
13	0.21
12	0.02
11	0.002
10	0.0002

the network. Each quad-core socket controls a Cray SeaStar2 chip router. This has six links which are used to implement a 3-D torus of processors. The point-to-point bandwidth is 2.17 GB/s, and the minimum bi-section bandwidth is 4.1 TB/s. The latency between two nodes is around 6 μs. The system is held in 60 cabinets and was ranked number 17 in the Top 500 supercomputer list in November 2007.

4.2. IBM Blue Gene/L and Blue Gene/P [9]

A Blue Gene/L (BGL) and a Blue Gene/P (BGP) have also been used for the simulations. Both single rack Blue Gene systems contain 1024 chips, with two processor cores per chip in the L system and four processor cores per chip in the P system, giving a total of 2048 and 4096 cores, respectively. Memory is provided at 512 MB per core in both systems. The processor in the L system is the POWER 440 running at 700 MHz, whilst the P system uses a POWER 450 processor running at 850 MHz. One rack of a BGL has a theoretical peak performance of 5.7 Tflop/s whereas one rack of a BGP has a peak performance of 13.9 Tflop/s.

4.3. HP cluster (Clamart2 [12])

This HP cluster was ranked number 457 in the Top 500 in November 2009. Its theoretical peak performance is 25 Tflop/s and has 272 nodes or 2176 Nehalem cores (Intel EM64T Xeon X55xx (Nehalem-EP) 2.93 GHz (11.72 Gflop/s)). The cluster uses a Lustre file system and InfiniBand interconnect.

5. Scaling performance of TELEMAC-2D

TELEMAC-2D is currently being used by EDF to help determine the optimal locations for the turbines at the proposed Paimpol-Bréhat demonstration marine turbine farm off the coast of Brittany in France (see Fig. 1) and to evaluate the anticipated energy yield. The most realistic approach for assessing the energy yield is based on simulating the tides over an entire year (∼706 tides). Some preliminary studies [13,14] have been carried out on a relatively coarse grid of about 30,000 elements using TELEMAC-2D. However, a better representation of the seabed can now be used because more accurate bathymetry data are available. In addition, better resolution of the flow around each of the turbines is desirable in order to capture the effects of turbine-wake interactions.

To demonstrate TELEMAC-2D's ability to handle larger mesh sizes, a 2-million element mesh has been built. The mesh was generated by splitting each of the elements from the initial 30,000-element mesh into 64 smaller elements. This provides a higher resolution grid but, unfortunately, does not provide any improvement to the seabed description due to the fact that the new bathymetry has to be linearly interpolated from the original (coarse) bathymetry. The 2-million element mesh has been used to demonstrate the scaling performance of the code and to estimate how much CPU time would be required to simulate one calendar year (∼706 tides). It is envisaged that the 2-million element mesh has sufficient resolution that it could be used, in the future, to estimate turbine wake effects by comparing simulations with and without the turbines in place.

The performance of TELEMAC-2D for the refined 2-million element mesh has been obtained using a simulation of 1000 s with a time step of 0.1 s. Fig. 2 shows the time necessary for TELEMAC-2D to complete the 1000 s test case while Fig. 3 shows the speed-up on the BGL, the BGP and the Cray. The speed-up can be defined as $T_{reference}/T_{N_C}$, where $T_{reference}$ is the total CPU time on the smallest core count for a given simulation, and T_{N_C} is the total CPU time on N_C cores. Overall, very good performance is observed on both Blue Genes; the performance remains almost linear up to 4096 cores on the BGP, and up to 2048 cores on the Cray XT4. A total of 113 CPU seconds are required on 4096 cores (1024 nodes) of the BGP to complete the 1000 s simulation, which should lead to about 40 CPU days for a whole year's simulation (∼706 tides).

6. Scaling performance of TELEMAC-3D

For many years, TELEMAC-3D has been used to study the impact of fresh water releases from a hydro-electric power plant in the

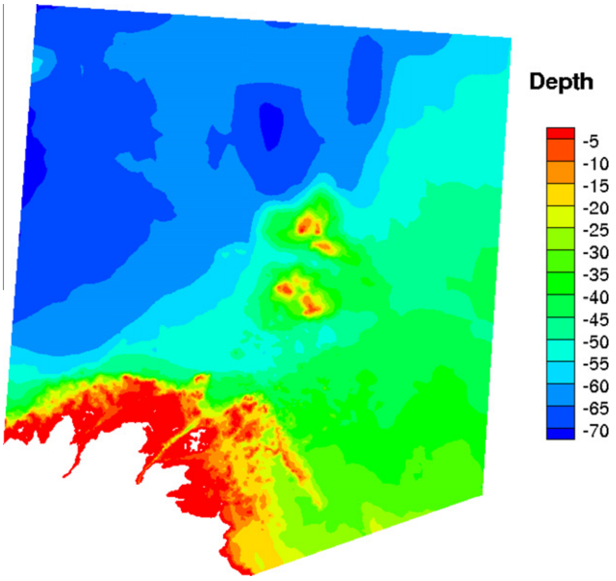

Fig. 1. Bathymetry of the Paimpol–Bréhat region. The spatial extent is approximately 64 km × 72 km.

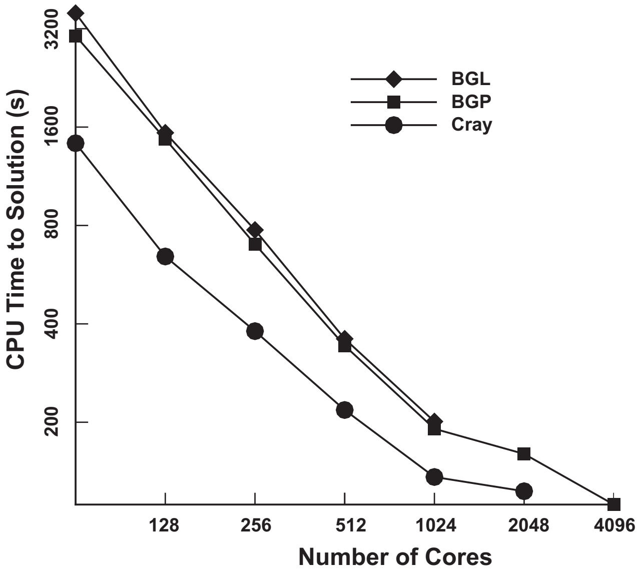

Fig. 2. TELEMAC-2D performance for a 2-million element mesh.

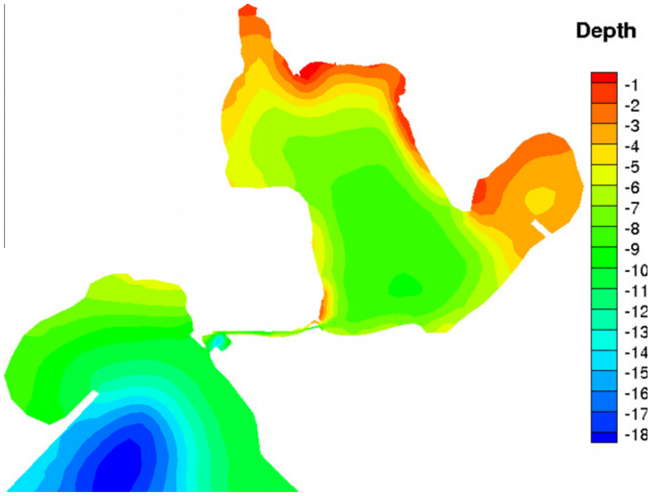

Fig. 4. Bathymetry of the Berre Lagoon. The modelled flow domain extends approximately 30 km × 25 km.

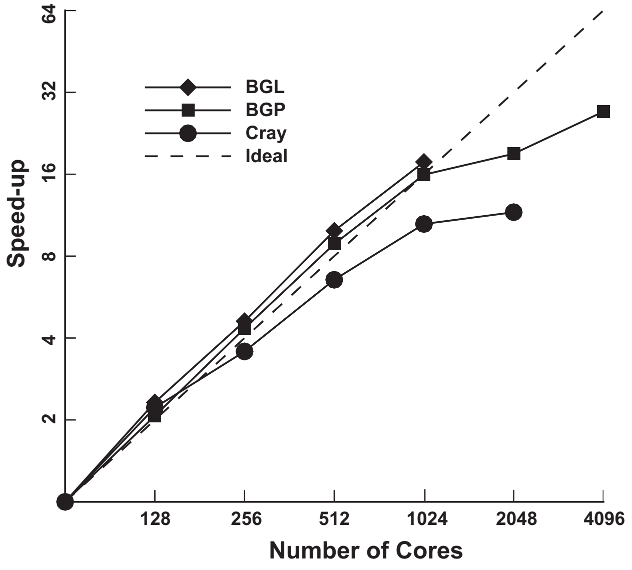

Fig. 3. Scaling performance of TELEMAC-2D. The speed-up is related to the performance on 64 cores on all the machines.

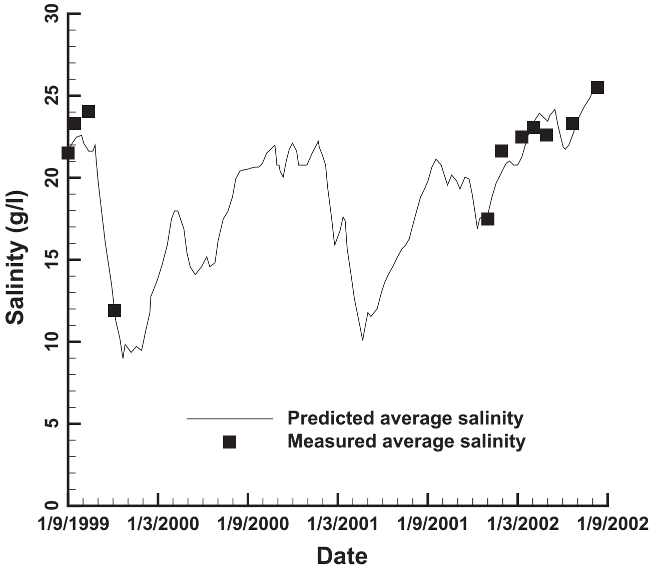

Fig. 5. Evolution of the average salinity in the Berre Lagoon over a 3 year period. The simulation was carried out with a coarse grid composed of 50,000 elements.

Berre Lagoon (see Fig. 4) in the south of France. The lagoon is connected to the Mediterranean Sea through a narrow channel and its salinity is influenced by the quantity of fresh water released during the operation of the power plant [15,16].

The simulations of the Berre Lagoon have been based, up until now, on relatively coarse meshes with typically around 50,000 elements. Fig. 5 shows the evolution of the average salinity in the lagoon obtained using a 50,000-element mesh. The predictions of the average salinity compare well with experimental data but the local salinity distribution is not predicted particularly well. The coarseness of the grid has a major influence on the applicability of turbulence models. For example, the original mesh of the Berre Lagoon considered 11 layers in the vertical direction and the simulations used a simple mixing length model for the turbulence. However, the use of 11 layers is probably insufficient to properly capture the stratified conditions that occur in the lagoon. Moreover, it would be beneficial to use a two-equation turbulence model (for example, a k-ϵ or k-ω model), to provide a much better description of the mixing. However, this would imply having to use a much finer resolution mesh in the vertical direction.

A 3-D numerical model based on 0.4 million horizontal elements has recently been built to improve the representation of both the bathymetry in the lagoon and the canal connecting the Berre Lagoon with the Mediterranean Sea. This should help provide better predictions of the salinity levels at the location where the canal enters the lagoon. It is estimated that 31 vertical layers (12 million elements) would give a better representation of the stratification, but ideally up to 101 layers (40 million elements) might eventually have to be used.

The scaling performance of TELEMAC-3D is presented in Fig. 6 for a 3-D model using 12 million elements (31 vertical layers). The strong scaling is good up to 4096 cores on the BGL, 1024 cores on the Cray, and 256 cores on the HP cluster; the performance then starts to level off. This can be explained by the smaller number of elements per processor on the BGL and the Cray, and probably by

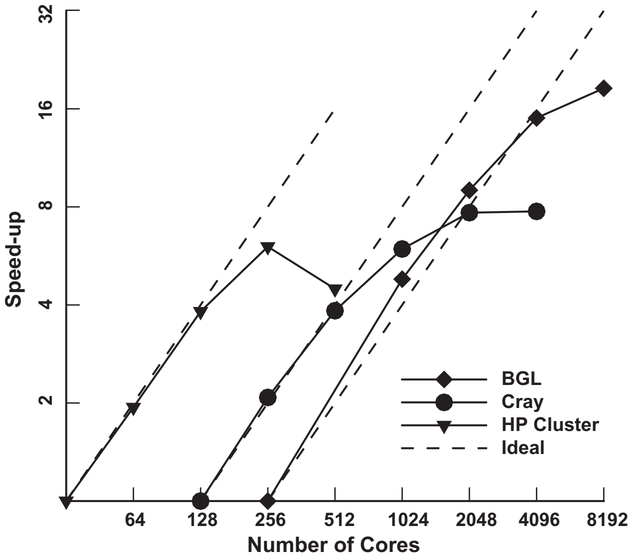

Fig. 6. Scaling performance of TELEMAC-3D. The speed-up is related to the performance on 256 cores on the BGL, 128 cores on the Cray, and 32 cores on the HP cluster.

the high processor speed on the HP cluster, which makes communications more costly than calculations on this particular machine. The poor scaling on the HP might also be explained by the use of an InfiniBand network. It is interesting to note that one year's simulation of the Berre Lagoon would require about 26 days on 2048 cores of the Cray XT4 for the 12-million element case.

7. Concluding remarks

This paper has investigated the scaling performance of TELEMAC-2D and TELEMAC-3D on massively parallel computer architectures. TELEMAC-2D has been used to study the tidal currents around a renewable energy marine turbine farm, and TELEMAC-3D has been used to investigate salinity levels in a coastal lagoon. Both codes have demonstrated good scaling performance on several high-end machines. Future work will include linking both codes to an error analysis tool (CADNA) to provide a better idea of error propagation in massively parallel simulations. In addition, studies involving substantially larger meshes will be performed to evaluate the role that HPC can play in providing increased fidelity for industrial applications of TELEMAC.

Acknowledgments

The authors would like to thank the UK Engineering and Physical Sciences Research Council (EPSRC) for their support of Collaborative Computational Project 12 (CCP12).

References

[1] Hervouet J-M. Hydrodynamics of free surface flows: modelling with the finite element method. Wiley; 2007.
[2] The TELEMAC system. <http://www.telemacsystem.com>.
[3] METIS. <http://glaros.dtc.umn.edu/gkhome/views/metis>.
[4] Goel S, Dash SK. Response of model simulated weather parameters to round-off-errors on different systems. Environ Modell Softw 2007;22:1164–74.
[5] Denis C. Numerical health check of industrial simulation codes from HPC environments to new hardware technologies. Parallel Process Appl Math. Lect Notes Comput Sci 2010:330–9.
[6] Rump SM. Algorithms for verified inclusions – theory and practice. In: Moore RE, editor. Reliability in computing: the role of interval methods in scientific computing. Perspectives in computing, vol. 19. Academic Press; 1988. p. 109–26.
[7] Alt R, Lamotte J-L. Experiments on the evaluation of functional ranges using random interval arithmetic. Math Comput Simul 2001;56(1):17–34.
[8] CADNA: control of accuracy and debugging for numerical applications. Université Pierre et Marie Curie, Paris. <http://www.lip6.fr/cadna>.
[9] <http://www-03.ibm.com/systems/deepcomputing/bluegene>.
[10] Rump SM, Ogita T, Oishi S. Accurate floating-point summation. Part I: faithful rounding. SIAM J Sci Comput (SISC) 2008;31(1):189–224.
[11] HECToR. <http://www.hector.ac.uk>.
[12] <http://www.top500.org/system/10387>.
[13] Pham C-T, Martin VA. Tidal current turbine demonstration farm in Paimpol-Bréhat (Brittany): tidal characterisation and energy yield evaluation with Telemac. In: 8th European wave and tidal energy conference. Uppsala; 2009.
[14] Pham C-T, Pinte K. Paimpol-Bréhat tidal turbine demonstration farm (Brittany): optimisation of the layout, wake effects and energy yield evaluation using Telemac. In: 3rd International conference on ocean energy. Bilbao; 2010.
[15] Razafindrakoto E, Martin LS, Hervouet J-M. 3D modelling of long-term hydrodynamics and water quality in the Berre Lagoon. In: 33rd IAHR congress. Vancouver; 2009.
[16] Martin LS, Gouze E, Razafindrakoto E, Hervouet J-M, Durand N, Pham C-T. 3D coupled modeling of hydrodynamics and water quality in the Berre Lagoon (France). In: Proceedings of 6th international symposium on environmental hydraulics. Athens; 2010.