

HPC for sensitivity studies: simulations with TOMAWAC and TELEMAC-3D

Noémie DURAND, Sébastien BOURBAN
HR Wallingford Limited, UK
n.durand@hrwallingford.co.uk

Charles MOULINEC, Robert BARBER,
Yoann AUDOUIN, Xiaojun GU, David EMERSON
STFC Daresbury Laboratory, UK

Abstract— High Performance Computing (HPC) is useful in a range of scientific applications, from running computationally-intensive high-resolution models to running a large number of smaller simulations either to calibrate the model (for example, using genetic algorithms) or to assess a range of design options or model scenarios. One particularly promising area for HPC is when carrying out *sensitivity studies*. In this case, a large number of simulations are performed using a systematic variation in the values of certain input parameters in order to determine the effects of each of the parameters on the model results. This type of application is *non-intrusive* in the sense that no modifications are required to the model source code. Instead, the only changes that need to be made are to user-generated subroutines and to the files that control the values of the input parameters.

In this paper, we describe a script which has been developed for use on an iDataPlex cluster (STFC Daresbury Laboratory) and a Cray XE6 supercomputer (UK National Supercomputing Service). The script enables concurrent simulations to be submitted to the system via the queuing facility available on parallel clusters. This is not only relevant for serial instances of the code but is also valid for parallel simulations. Using the script for parallel instances allows the elapsed time for sensitivity studies to be reduced by a factor of approximately $\text{NCSIMS} \times \text{ICORES}_{\text{OPT}}$, where NCSIMS is the number of concurrent instances and $\text{ICORES}_{\text{OPT}}$ is the optimum number of cores used for running each parallel instance.

I. INTRODUCTION

TOMAWAC [1,2] is a third generation spectral wave model which accounts for wave generation, refraction, shoaling, non-linear wave interactions and energy dissipation due to white-capping, bottom friction and depth-induced breaking. It was developed by EDF R&D as part of the TELEMAC finite element hydrodynamic modelling system [3-5]. TELEMAC-3D [6] is the 3D hydrodynamic module within TELEMAC which solves the complete Navier-Stokes equations.

In many instances, it is necessary to run TOMAWAC or TELEMAC-3D as part of a sensitivity study. This requires the code to be run many times using a systematic variation in the input parameters. This paper describes the development of a script to enable concurrent simulations to be submitted to the job queuing facility on parallel clusters. Most of the script is used to control the simulations, i.e. creation of temporary

system directories from which the model can be run locally, creation of the local executables, and retrieval of the simulation results. The script is primarily independent of the queuing system. Only the last step of running the model is dependent on local environment variables and other platform-specific commands. The main entries for the script are the number of simulations, the names of the steering files, the name of the TELEMAC executable and the number of MPI tasks per parallel simulation.

II. HARDWARE/SOFTWARE

A. IBM iDataPlex cluster [7]

STFC Daresbury Laboratory hosts an IBM iDataPlex system (see Fig. 1) composed of a total of 76 nodes (912 cores), distributed as follows; 38 nodes for the exclusive use by STFC Daresbury Laboratory, 20 nodes for external consultancy and 18 nodes for the University of Huddersfield. Each node has 12 Westmere cores, i.e. two Intel X5650 six Core 2.66 GHz processors. A total of 24 GB RAM is available per node. A Qlogic QDR InfiniBand system is used as the switch.

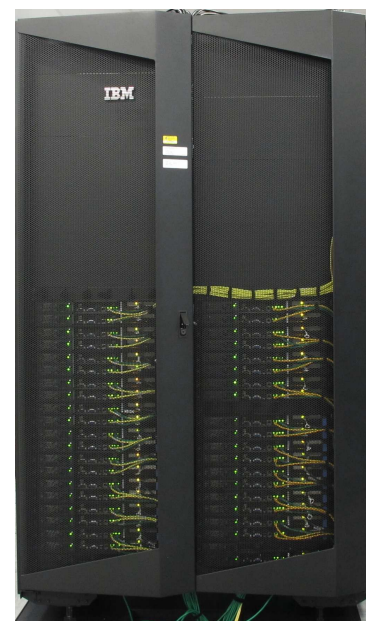


Figure 1. The iDataPlex cluster at STFC Daresbury Laboratory.

The iDataPlex cluster uses the Portable Batch System (PBS) to manage the queues. The Fortran compiler employed on the iDataPlex system is version 11 of the Intel compiler and the TELEMAC system is compiled with the `-O3` optimisation option. Parallel communications are handled in TELEMAC using MPI; version 1.4.3 of OpenMPI is currently used on the iDataPlex system.

B. Cray XE6 [8]

HECToR Phase 2b (Cray XE6) (see Fig. 2) is the UK National Supercomputing Service hosted at the University of Edinburgh. The system is contained in 20 cabinets and comprises a total of 464 compute blades. Each blade contains four compute nodes composed of two 12-core AMD Opteron 2.1 GHz Magny Cours processors (24 cores per node); this amounts to 44,544 cores in total. Each 12-core socket is coupled using a Cray Gemini routing and communications chip, and each 12-core processor shares 16 GB of memory, giving a system total of 59.4 TB. The theoretical peak performance of the Phase 2b system is over 360 Tflop/s.



Figure 2. HECToR (Cray XE6) hosted at the University of Edinburgh.

The Cray XE6 also uses the Portable Batch System (PBS) to manage the queues. The compiler employed on the Cray XE6 is GNU `gcc/4.5.3` from the GNU project and the TELEMAC system is again compiled with the `-O3` option. MPICH2 version 5.3.0 manages the parallel communications.

III. DESCRIPTION OF THE SCRIPT

A bash script has been developed that builds a second script which is referred to, hereafter, as the *job submission script*. The latter is submitted to the queuing system in order to run the concurrent simulations. These may be run in serial, or in parallel, depending upon the user's specific requirements and the size of problem being considered. A typical scenario for running concurrently is when performing a large number of similar simulations in the context of a sensitivity analysis.

The bash script contains seven entries as follows:

- the name of the *job submission script*,
- the name of the module of the TELEMAC system to be run (in this case, *tomawac* or *telemac3d*). The name of the module is referred to as *telmodule*,
- the number of concurrent simulations (or *telmodule* instances), NCSIMS. These are referenced by the variable, ICSIM, which runs from 1 to NCSIMS,
- the number of processes per node on the targeted cluster. In theory, this number should be 12 for the iDataPlex cluster and 24 for the Cray XE6, but might be reduced because of memory constraints,
- the total number of cores used by each *telmodule* simulation. This is equal to 1 if each *telmodule* simulation is run in serial; otherwise, this is set to the number of sub-domains that the grid is to be partitioned into,
- the estimated number of elapsed hours that the job will take, so as to inform and help with the booking process on the cluster, and finally,
- the estimated number of minutes (again to inform and help with the booking process on the cluster).

A pre-processing stage precedes the job submission and is performed as follows. The names of the parameter files for all concurrent simulations are listed in a file called *casfile.dat* (NCSIMS lines). A loop over these filenames is then performed that creates NCSIMS temporary directories and files by running *telmodule casname(ICSIM)*. If different user-subroutines are used from one ICSIM instance to another, a new executable is created.

The names of the temporary *telmodule* directories are then written to a file named *dirfile.dat* in an attempt to simplify the rest of the submission process.

A loop over the directory names in *dirfile.dat* creates symbolic links to the temporary files in DIR(ICSIM) with ICSIM varying between 1 and NCSIMS. This completes the pre-processing stage.

The *job submission script* is then built from the bash script. The operating principle of the *job submission script* is described in the broadest of terms, without going into the details of queuing-system-dependent settings.

A loop over the NCSIMS simulations is performed:

- While ICSIM is less than NCSIMS+1
- Change to directory DIR(ICSIM)
- Run the ICSIMth executable
- End of the While loop

The *job submission script* is then submitted to the queuing system. The specified number of hours and minutes are used by the queuing system to determine which queue should be used to run the job, as the queues are different for short and long runs.

IV. COSTING THE SIMULATIONS

The tendering process for consultancy projects and the preparation of research proposals for computing time on High-End facilities often require an accurate estimate to be made of the total elapsed time necessary to complete the simulations and also the cost of the total CPU time (the cost being based upon the number of core-hours). In the present study, a series of formulae have been developed that help determine the cost of the numerical simulations. Before detailing the cost formulae, it is necessary to list the notation and the assumptions that have been made in the analysis. It is assumed that:

- ICSIM denotes the ICSIMth of the NCSIMS *telmodule* instances.
- each *telmodule* instance is assumed to run on ICORES, where ICORES ranges from ICORES_{REF} to NCORES. Here, ICORES_{REF} is the number of cores used by the *reference* simulation, with ICORES_{REF} = 1 for serial instances. The reference simulation is defined as the one using the smallest number of cores that will allow the *telmodule* instance to run.
- all NCSIMS instances are run on the same number of cores, ICORES.
- T(ICSIM, ICORES) is the time for a given ICSIMth instance to complete.
- T_{ICORES} is the time for the *slowest* of all the NCSIMS instances to complete and is defined as

$$T_{\text{ICORES}} = \max(T(\text{ICSIM}, \text{ICORES})) \quad (1)$$

for $1 \leq \text{ICSIM} \leq \text{NCSIMS}$.

The total time for the job to complete can then be estimated as

$$\text{TOT}_{\text{ICORES}} = T_{\text{ICORES}} + T_{\text{SYSTEM}}, \quad (2)$$

where T_{SYSTEM} is the time required by the system to manage the job involving the concurrent simulations.

The overall speed-up of the *telmodule* instances run on ICORES for the NCSIMS simulations can therefore be defined as

$$\text{SPEED-UP}(\text{ICORES}) = \text{TOT}_{\text{ICORES}_{\text{REF}}} / \text{TOT}_{\text{ICORES}}. \quad (3)$$

The best performance, SPEED-UP_{OPT}, is achieved when the instances are run on ICORES_{OPT} cores and can be defined as

$$\begin{aligned} \text{SPEED-UP}_{\text{OPT}} &= \text{SPEED-UP}(\text{ICORES}_{\text{OPT}}) \\ &= \max(\text{SPEED-UP}(\text{ICORES})) \quad (4) \\ &\textit{for } \text{ICORES}_{\text{REF}} \leq \text{ICORES} \leq \text{NCORES}, \end{aligned}$$

where ICORES_{OPT} is the optimum number of cores for the *telmodule* runs for the specified grid size, input parameters and parallel cluster. The value of SPEED-UP assesses the performance of *telmodule* for each particular configuration.

Now if the reference simulations were run *consecutively* on ICORES_{REF} cores, and assuming that T_{SYSTEM} is negligible, then the total elapsed time, TOT_EL_{CONSEC-REF}, to complete all the consecutive simulations would be given by

$$\text{TOT_EL}_{\text{CONSEC-REF}} = \sum_{\text{ICSIM}=1}^{\text{NCSIMS}} (T(\text{ICSIM}, \text{ICORES}_{\text{REF}})). \quad (5)$$

Assuming that all the values of T(ICSIM, ICORES_{REF}) are of the same order of magnitude, then the summation can be approximated by

$$\text{TOT_EL}_{\text{CONSEC-REF}} \approx \text{NCSIMS} \times T_{\text{ICORES}_{\text{REF}}}. \quad (6)$$

If all NCSIMS instances were run *concurrently* on ICORES_{OPT} cores, then it would take TOT_{ICORES-_{OPT}} seconds to complete the simulations. The duration of the project would then be reduced by approximately TOT_EL_{CONSEC-REF} - TOT_{ICORES-_{OPT}}.

Moreover, it is also possible to estimate the total CPU requirement when NCSIMS *concurrent* runs are each performed on ICORES:

$$\text{TOT_CPU}_{\text{CONCURR}} = \text{NCSIMS} \times \text{ICORES} \times T_{\text{ICORES}}. \quad (7)$$

It is again assumed that all the timings for T(ICSIM, ICORES) are of the same order of magnitude. The CPU cost of the concurrent simulations, relative to the cost of the *reference* concurrent simulations can then be expressed as

$$\begin{aligned} \text{RELATIVE_COST} &= \frac{\text{TOT_CPU}_{\text{CONCURR_OPT}}}{\text{TOT_CPU}_{\text{CONCURR_REF}}} \quad (8) \\ &= \frac{\text{ICORES}_{\text{OPT}} \times T_{\text{ICORES}_{\text{OPT}}}}{\text{ICORES}_{\text{REF}} \times T_{\text{ICORES}_{\text{REF}}}}. \end{aligned}$$

Equation (8) can be used to estimate the cost of the optimised concurrent simulations compared to the reference concurrent simulations.

V. TESTS USING TOMAWAC

A. Description of the problem

The first test case considers the simulation of the morphological processes that affect dredging spoil that has been used to create an artificial island. Although the island is not intended for infrastructural development, there is concern that the spoil material could suffer erosion, with sediment being eventually transported back to the original dredged channel. HR Wallingford has developed a series of numerical models to study the coastal morphological processes affecting the site and to determine the potential for sediment to affect the dredged channel. Waves contribute to the re-suspension and transport of sediments in the shallows and breaking zone. The test case described here only considers the wave modelling aspect of the overall study.

A TOMAWAC model was set up to account for the wave transformation processes (e.g. refraction, shoaling and depth-induced breaking) as the waves propagate from deep to shallow water. Currents can also affect the waves as they travel nearshore, depending on whether the currents follow or

counter the predominant wave direction, and therefore wave-current interactions have been accounted for in the present model. The TOMAWAC mesh used in the study was composed of 57,177 elements and is shown in Fig. 3.

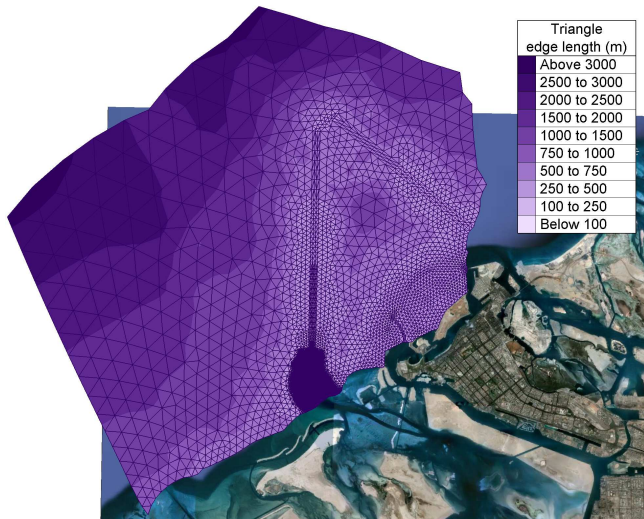


Figure 3. Computational domain for the TOMAWAC study.

Offshore model data were obtained for a point on the TOMAWAC boundary. These data cover a period from January 1983 to December 2002. The offshore wave climate is presented as a wave rose in Fig. 4, indicating the frequency of occurrence of certain wave conditions, discretised into directional sectors and wave height bins. In this case, each ring on the wave rose represents a frequency of occurrence of 10%. Referring to Fig. 4, it is clear that north-westerly waves are the predominant offshore waves in the region of interest. The significant wave heights are below 1.5 m for about 90% of the 20-year record but extreme wave heights can be greater than 4 m.

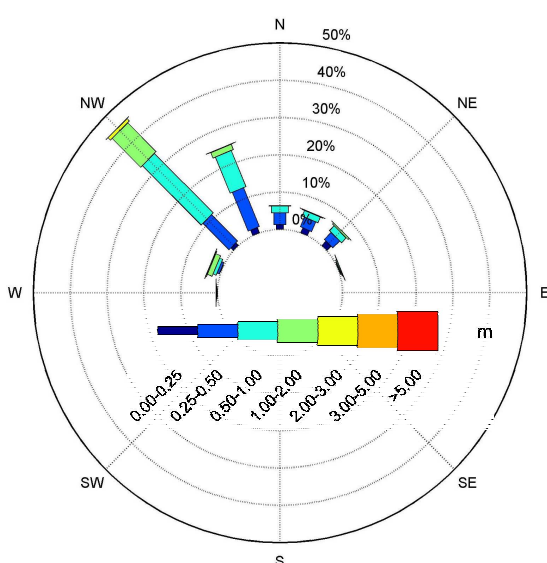


Figure 4. Wave rose at the offshore boundary of the model, computed from wave data covering a period from January 1983 to December 2002.

B. Analysis of the results

It is usually not practical to run the wave model directly for the duration of the offshore wave record. Instead, a look-up table approach is often preferred, whereby the wave model is run for a range of discrete wave and wind conditions, representative of the offshore wind and wave climates. Multi-dimensional look-up tables are then created relating the modelled offshore wave and wind conditions to the nearshore wave conditions at the site. These look-up tables are applied to the offshore time series to provide corresponding long-term nearshore time series. Ideally, the model would be run for thousands of conditions in order to cover all realistic combinations. Added complexity is introduced when the state of the tide (water level and current variations) plays a significant role in the wave conditions at the site.

Due to constraints with HR Wallingford's computational resources, the original study using TOMAWAC was run for 41 conditions representative of the offshore wave climate, considering wave directions between 270°N and 60°N and a water level of Mean High Water. Local wave generation for directions between 60°N and 270°N was included, based on fetch length and wind speeds, and follows the method proposed by Hasselmann et al. [9]. The nearshore wave climate resulting from this analysis is presented in Fig. 5 and was then used in conjunction with flow model results and sediment and borehole data to assess the long term stability of the sea bed around the island, and the potential for infilling of the dredged channel as a result of erosion on the island. Fig. 6 displays a typical result from one of the simulations, and shows the spatial variations of predicted significant wave height at the site. In this figure, the grey arrows are a visual representation of the mean wave direction. For clarity, the vectors are not shown for all grid points.

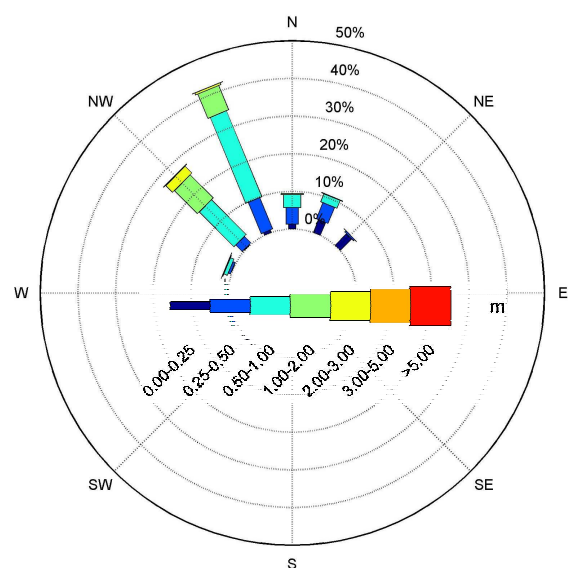


Figure 5. Predicted nearshore wave rose.

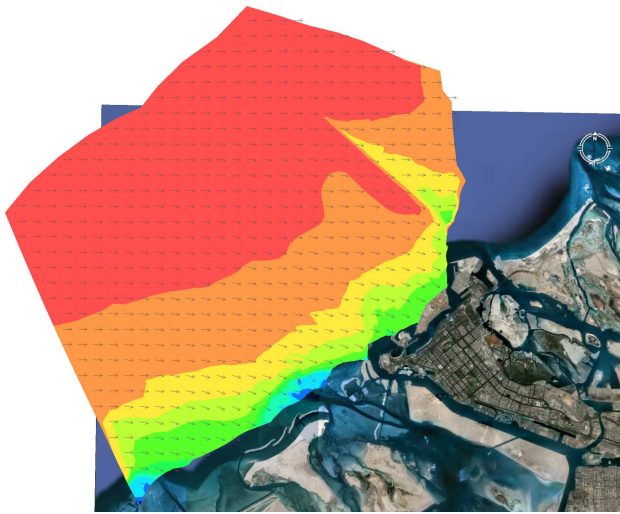


Figure 6. Predicted contour map of significant wave height and mean wave direction at the site.

C. Use of the script on the iDataPlex cluster and the Cray XE6

For the purposes of this investigation into the wider use of HPC in consultancy and research, the same wave model was run on the iDataPlex cluster and the Cray XE6. All 41 instances of the TOMAWAC study (NCSIMS = 41) were set to run for 2000 time steps. In this particular case, $ICORES_{REF} = 1$. Running all the instances concurrently, albeit in serial, takes approximately 25,700 s and 33,358 s on the iDataPlex cluster and the Cray XE6, respectively. The simulations on the iDataPlex cluster are faster for two reasons; firstly its processor is faster than the Cray XE6 processor, and secondly, the internal vectorising options in the Intel compiler generate faster executables, in general, than the GNU compiler.

Figs. 7 and 8 present the time to solution for each simulation (full squares) as well as the time required for the jobs to complete (open circles). When using a small number of cores per simulation, the total time required for the job to complete is very close to the time for the slowest simulation. However, when $ICORES$ becomes large (see for example, $ICORES = 192$ or $ICORES = 384$ in Fig. 8), T_{SYSTEM} is no longer negligible. On the Cray XE6, comparison between $ICORES = 192$ and $ICORES = 384$ shows that the speed-up based on the simulation time is acceptable (1.42) whereas the speed-up based on the job time is poor (1.09).

The value of $SPEED-UP_{OPT}$ is equal to 5.47 for the iDataPlex cluster compared to a theoretical value of 6 assuming a linear speed-up. The corresponding value for the Cray XE6 is 110.09 compared to the theoretical value of 384. This lower than theoretical performance can mainly be attributed to the small size of the grid.

For the iDataPlex cluster, the total elapsed time, $TOT_{EL_{CONSEC_{REF}}}$, to complete all the consecutive simulations is approximately equal to 1,053,700 s (about 12 days); the corresponding figure for the Cray XE6 is 1,367,678 s (about 16 days). In contrast, running concurrent simulations

on the optimum number of cores ($ICORES_{OPT} = 6$ for the iDataPlex cluster and 384 for the Cray XE6) leads to $TOT_{ICORES_{OPT}} = 4700$ s and 303 s, respectively.

Since $TOT_{ICORES_{OPT}}$ is extremely small compared to the total elapsed time, $TOT_{EL_{CONSEC_{REF}}}$, the reduction in time when running concurrent parallel simulations is about 12 days on the iDataPlex cluster and about 16 days on the Cray XE6. The fact that $ICORES_{OPT}$ is, for both machines, the highest number of cores that could be used per simulation, demonstrates excellent performance of TOMAWAC, even for such a relatively small problem size (57,177 elements).

Figs. 9 and 10 show the relative cost of running each simulation on more than $ICORES_{REF}$ cores, and show that there is a benefit of running TOMAWAC with concurrent parallel simulations on the iDataPlex cluster but there is a penalty when running on the Cray XE6. However, the vastly reduced time to solution is thought to more than compensate for the increased computational costs.

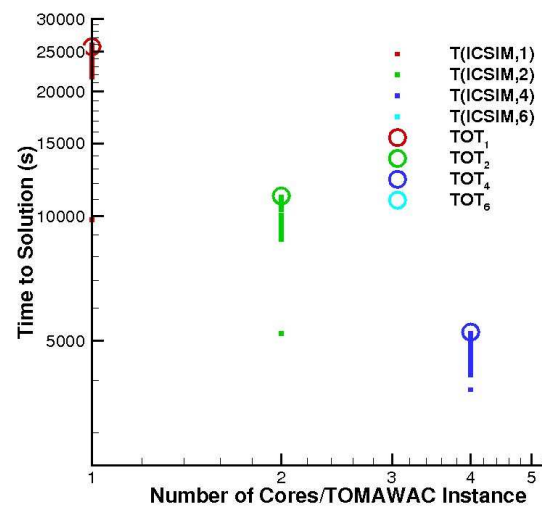


Figure 7. Time to solution for the 41 concurrent TOMAWAC simulations on the iDataPlex cluster.

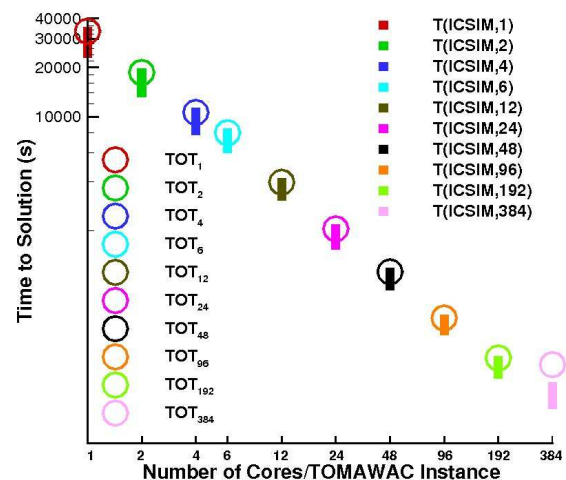


Figure 8. Time to solution for the 41 concurrent TOMAWAC simulations on the Cray XE6.

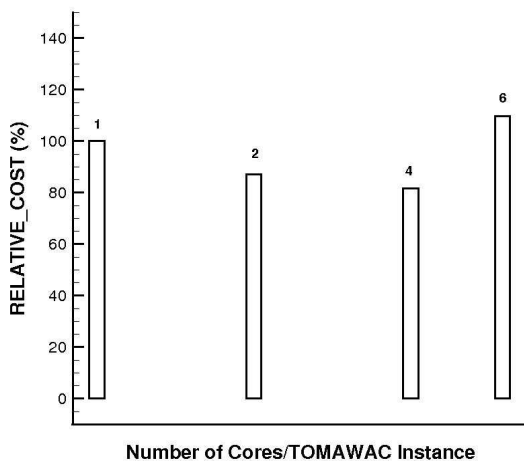


Figure 9. Relative cost of the TOMAWAC simulations on the iDataPlex cluster.

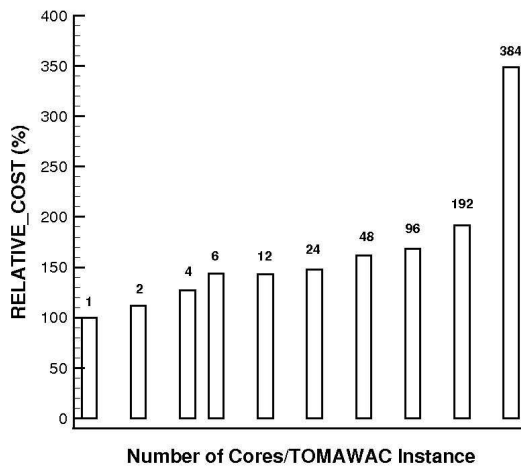


Figure 10. Relative cost of the TOMAWAC simulations on the Cray XE6.

VI. TESTS USING TELEMAC-3D

A. Description of the problem

In this case study, HR Wallingford had an advisory role in support of the design and the acquisition of planning permission for a proposed coastal development. Since the project remains confidential, references to names, locations and model values have been modified or removed; thus any resemblance with any particular site is purely coincidental. The bathymetry of the site is shown in Fig. 11.

For this project, HR Wallingford carried out physical model tests aimed at measuring wave overtopping along a new seawall under a range of waves with different return periods and different states of the tide. It was noticed that the combination of strongly oblique waves and a (very nearly) vertical seawall give rise to a local wave travelling along the seawall, termed for this study a ‘stem-wave’ but sometimes also referred to as a ‘Mach-stem wave’. These stem-waves result from the non-linear interaction between the incident

and reflected wave along vertical breakwaters for a specific range of oblique incident angles.

A range of numerical models have been developed for the site in order to study more closely the phenomenon of ‘stem wave’ generation along the protected coastline. The numerical study considered the severity of any ‘stem wave’ enhancement, whether that effect might continue to increase beyond the area represented in the physical model, as well as related effects which might cause problems for the adjoining site (in the direction of wave travel). The numerical modelling work was carried out using the TELEMAC-3D module of the TELEMAC system.

The TELEMAC-3D model was set up in non-hydrostatic and wave resolving modes in order to predict non-linear wave propagation, refraction and shoaling in front of the seawall. In this model, the size of the elements varies from 1 m near the seawall to approximately 5 m at the offshore boundary.

Extreme offshore wave conditions were derived based on 20 years of model wind and wave data. The first step was to propagate these conditions closer to the site (not detailed here). Sensitivity tests to the state of the tide demonstrated that extreme nearshore wave conditions were well represented by modelling the conditions at high water. Selected outputs from the wave transformation model were in turn used as input to TELEMAC-3D. Were it not for the time constraints of this short-term project, it would have been desirable to simulate a long record of randomly generated waves. Instead, only a small number of regular, yet non-linear waves, were generated by imposing the water level and the vertical pressure and velocity profiles at the offshore boundary.

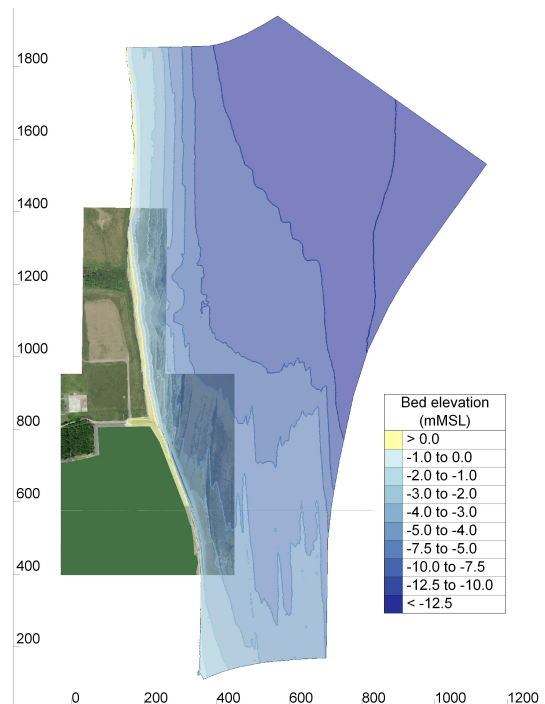


Figure 11. Bathymetry for the TELEMAC-3D study.

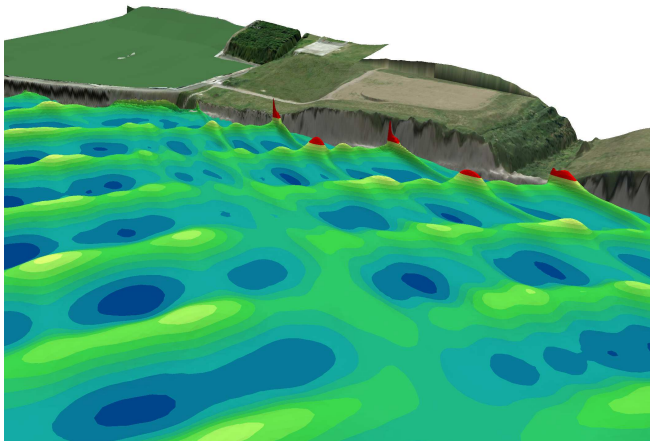


Figure 12. Predicted wave field for the 1-in-200 year event, illustrating stem-wave like effects close to the seawall.

B. Analysis of the results

The preliminary results presented here are for waves of amplitude 5 m and of period 10 s, approximating the conditions with a 1-in-200 year return period (see Fig. 12). It is shown that the waves undergo refraction and shoaling as they propagate towards the coast, reaching the seawall with an angle that is less oblique than the incident angle. Local stem-wave like effects were also observed in the model although the effect reduces as the depth of water at the seawall toe decreases. TELEMAC-3D was able to confirm physical model observations. Significantly more computer time would be required to analyse the output from the numerical model statistically.

C. Use of the script on the Cray XE6

A computationally more demanding test has been performed running 12 parallel concurrent TELEMAC-3D simulations, each composed of a horizontal grid of 376,158 elements and 10 layers (3,761,580 prisms in total). This test could only be carried out on the Cray XE6 since the iDataPlex cluster does not have enough cores for a relevant demonstration. All 12 instances of TELEMAC-3D (NCSIMS = 12) were set to run for 6000 time steps. Each simulation was run on at least 24 cores (i.e. $ICORES_{REF} = 24$). This also corresponds to the number of cores on each node of the Cray XE6.

Fig. 13 presents the time to solution for each simulation (full squares) together with the time required for the jobs to complete (open circles). Comparison between $ICORES = 384$ and $ICORES = 768$ shows that the speed-up based on the simulation time (1.71) and the speed-up based on the job time (1.63) are very similar. This is mainly because only 12 concurrent simulations have to be dealt with by the system and therefore T_{SYSTEM} is negligible for such a small number of simulations. The value of $SPEED-UP_{OPT}$ for the TELEMAC-3D jobs on the Cray XE6 is equal to 33.49 compared to a theoretical value of 32.

Finally, Fig. 14 shows the relative cost of running each simulation on more than $ICORES_{REF}$ cores. The analysis

indicates that it is actually still cheaper, in terms of the total CPU time, to run on 768 cores rather than on 24 for this particular test case, highlighting the benefits of running concurrent TELEMAC-3D instances in parallel.

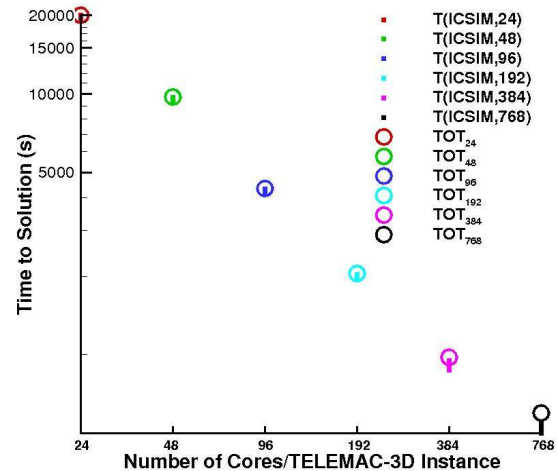


Figure 13. Time to solution for 12 concurrent TELEMAC-3D simulations on the Cray XE6.

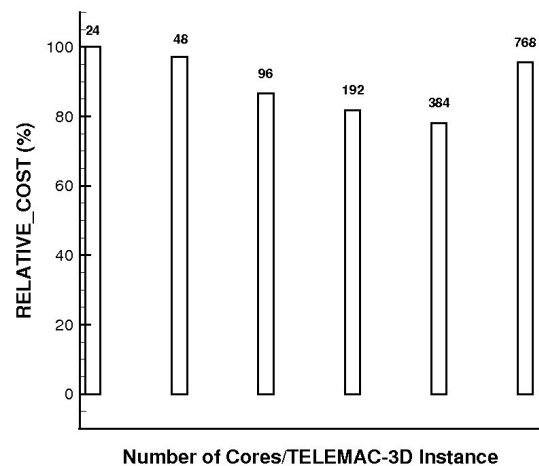


Figure 14. Relative cost of the TELEMAC-3D simulations on the Cray XE6.

VII. CONCLUSIONS AND RECOMMENDATIONS

This paper has presented a script which can be used to run concurrent parallel simulations on High-End computing resources. The script is specifically aimed at sensitivity analyses where it is necessary to run a large number of identical simulations using different input parameters.

The script enables concurrent simulations to be run either in serial or in parallel, depending upon the user's specific requirements and the size of problem being considered. The results show that the time to solution for both TOMAWAC and TELEMAC-3D can be dramatically reduced with the help of HPC facilities. This will not only improve future calibration exercises but will also allow far

more simulations to be run in order to assess a wider range of design options or model scenarios. In turn, this should lead to increased fidelity of wave-climate predictions for both research and consultancy projects.

ACKNOWLEDGEMENTS

This work was initiated by the South-East England Development Agency (SEEDA) which aims to promote strong relationships between academia and businesses in South-East England. In this context, HR Wallingford was granted a SEEDA Voucher covering STFC Daresbury Laboratory staff and computing time.

The authors would also like to thank the UK Engineering and Physical Sciences Research Council (EPSRC) for their support of Collaborative Computational Project 12 (CCP12) and the Distributed Computing Group at STFC Daresbury Laboratory.

REFERENCES

- [1] M. Benoit, F. Marcos, and F. Becq, "Development of a third generation shallow-water wave model with unstructured spatial meshing," Proc. 25th Int. Conf. on Coastal Engineering, ASCE, Orlando, USA, 1996, pp. 465-478.
- [2] M. Benoit, TOMAWAC Software for Finite Element Sea State Modelling - User Manual - Release 5.5, EDF-LNHE, Chatou, France, 2005.
- [3] J.-M. Hervouet, Hydrodynamics of Free Surface Flows: Modelling with the Finite Element Method, Chichester: John Wiley & Sons, 2007.
- [4] <http://www.opentelemac.org>
- [5] C. Moulinec, C. Denis, C.-T. Pham, D. Rougé, J.-M. Hervouet, E. Razafindrakoto, R.W. Barber, D.R. Emerson, X.-J. Gu, "TELEMAC: an efficient hydrodynamics suite for massively parallel architectures," Computers & Fluids (2011), doi:[10.1016/j.compfluid.2011.07.003](https://doi.org/10.1016/j.compfluid.2011.07.003)
- [6] TELEMAC-3D - Operating Manual - Release 5.8, EDF-LNHE, Chatou, France, 2007.
- [7] http://www.nw-grid.ac.uk/STFC_iDataPlex
- [8] <http://www.hector.ac.uk>
- [9] K. Hasselmann et al., "Measurements of wind-wave growth and swell decay during the Joint North Sea Wave Project (JONSWAP)," Dtsch. Hydrogr. Z. A8(12), 1973.