A Performance Study of Applications in the Australian Community Climate and Earth System Simulator

Mark Cheeseman, Ben Evans, Dale Roberts, and Marshall Ward

National Computational Infrastructure, Canberra, Australia
{mark.cheeseman,ben.evans,ds.roberts,marshall.ward}@anu.edu.au

Abstract. A 3-year investigation is underway into the performance of applications used in the Australian Community Climate and Earth System Simulator on the petascale supercomputer Raijin hosted at the National Computational Infrastructure. Several applications have been identified as candidates for this investigation including the UK MetOffice's Unified Model (UM) atmospheric model and Princeton University's Modular Ocean Model (MOM). In this paper we present initial results of the investigation of the performance and scalability of UM and MOM on Raijin. We also present initial results of a performance study on the data assimilation package (VAR) developed by the UK MetOffice and used by the Australian Bureau of Meteorology in its operational weather forecasting suite. Further investigation and optimization is envisioned for each application investigated and will be discussed.

Keywords: climate simulation, Unified Model, performance evaluation.

1 Introduction

1.1 Australian Community Climate and Earth System Simulator Optimization Project

The Australian Community Climate and Earth System Simulator, or ACCESS (Keenan, et al., 2014), is an integrated software system for coupled climate and earth system simulations. It is a joint initiative of the Australian Bureau of Meteorology (Bureau) and the Commonwealth Scientific and Industrial Research Organization (CSIRO) in cooperation with the university community in Australia. It is used extensively by the Australian academic community through the Australian Research Council Centre of Excellence for Climate System Science (ARCCSS) (ARC Centre of Excellence for Climate System Science, 2011). The simulation work performed within ACCESS is so computationally intensive that supercomputers are needed to complete this work at a reasonable rate. One such supercomputer heavily used for ACCESS-related work is Raijin –a petascale supercomputer installed at the National Computational Infrastructure (NCI) in Canberra, Australia. With ACCESS-related research being a core activity, knowing (and improving) the performance of ACCESS-related simulations on Raijin is of great interest. Improved performance will translate into additional and/or larger simulations that achieve greater scientific output.

R. Denzer et al. (Eds.): ISESS 2015, IFIP AICT 448, pp. 589–598, 2015.

To date, no coordinated effort has been done to gauge the current performance of the ACCESS framework at NCI and what could be done to increase it. The ACCESS Optimization project (referred to as ACCESS-Opt henceforth) was created to address the issue. This three-year collaboration between NCI, Bureau and Fujitsu has clear performance-driven goals: 1) determine strategically important ACCESS simulations being run on Raijin (or to be run in the very near future), 2) assess the performance of these configurations on Raijin, and 3) implement upgrades for identified performance issues in these simulations.

ACCESS-Opt provides an opportunity for high performance computing experts to engage with Australian weather/climate researchers. It is hoped that this collaboration will further their scientific deliverables and aspirations with ACCESS.

1.2 Raijin Supercomputer

ACCESS simulations performed at NCI use Raijin -an x86 based cluster comprised of 3592 nodes connected by Infiniband FDR interconnect. Each node contains two 8-core Intel Sandybridge CPUs running at 2.6 MHz clock speed. At least 32 GB of physical memory is available and shared between the two CPUs on a node. A high-speed Lustre-based filesystem containing approximately ten petabytes of space is available for high-speed storage. Intel FORTRAN and C compilers are used for source code compilation. OpenMPI (Gabriel, et al., 2004) is used for message-passing support in the parallel applications.

2 Project Methodology

2.1 Organizational Collaboration

Cooperation and collaboration between the member organizations is key to ACCESS-Opt's success and methodology. NCI specialists first interact with climate and NWP researchers at Bureau, CSIRO and ARCCSS with the goal of determining which ACCESS applications (and corresponding configurations) should be investigated. The same specialists, with input and assistance from staff at the before-mentioned organizations, then conduct performance assessments and possible optimizations on the selected configurations. The results are then relayed back to the same researchers for dissemination. Wider distribution of the results is made through publications and presentations.

2.2 Software Benchmarking

Figure 1 outlines the methodology followed when assessing an application's performance. The first step is to perform what is known as a strong scaling analysis. Here, a particular configuration of an ACCESS simulation is run multiple times. All runs are identical except for the number of CPU cores being used –which is increased at a constant factor (usually two). Ideally, as the number of cores is doubled, the simulation's runtime should be halved. So, if one were to plot the ratio of the observed runtimes at increasing numbers of cores to the observed runtime at the smallest number of cores used, one would get a straight line if the vertical axis is logarithmic with a base of 2. There are multiple examples of such strong scaling plots in the paper stating with Figure 3.

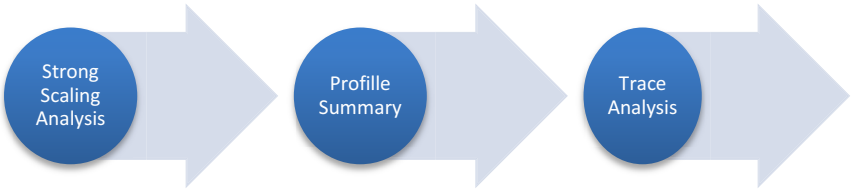

Fig. 1. Performance assessment methodology

Strong scaling plots generally will indicate the point where adding additional CPUs to a given simulation is no longer prudent. At this point, one needs to take a more detailed look at the application's performance by creating a performance profile. A performance profile summarizes where an application's runtime is being spent. The profile's specificity is up to the investigator –one can record the time spent in general types of activity observed (I/O, communication, computational, etc) or in individual subroutines and functions. Figure 2 shows a basic profile for a data assimilation code ran on Raijin using 384 CPU cores –with each core running a single MPI task. One sees that communication-related activity clearly dominates the application's run-time. Profiling was performed with the open-sourced tool, Score-P (Schlutter, Philippen, Morin, Geimer, & Mohr, 2014).

The final step is to generate traces for selected configurations on ACCESS-related jobs. In a trace, snapshots of activity within a running job are gathered at a specified frequency for the duration of the entire job. While a performance profile can indicate which parts of an application take up significant portions of its runtime, a trace can show how and when these "hot-spots" accumulate during a run. Visualization of trace logs is performed using the Vampir software (Overview: Vampir 8.3, 2014).

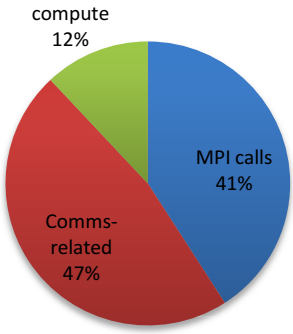

Fig. 2. Profile of the data assimilation application at 384 CPU cores

3 Preliminary Results

3.1 Unified Model

The Unified Model (UM) is a general atmospheric circulation model developed by the United Kingdom MetOffice (UKMO). An overview of its development can be found

in (Brown, Milton, Golding, Mitchell, & Shelly, 2012). UM is used operationally for numerical weather prediction in Australia, New Zealand and other countries. Under ACCESS, it is used for atmosphere-alone simulations and serves as the atmospheric component in coupled climate model configurations.

Initial UM optimization work uses version 8.4 of the UM and has focused on a global N512L70 configuration (eg. ~25km horizontal resolution with 70 model vertical levels) -which we will refer to as UM-N512L70 from this point onwards. This configuration due for use by the Bureau for operational weather forecasting duties and is also well known by a number of supercomputer vendors (including Fujitsu). It uses distributed memory parallelism almost exclusively with the addition of OpenMP threading in certain subroutines and areas such as I/O.

I/O is typically a dominant factor in the performance of any earth-system application. UM has adopted an asynchronous IO server approach to improve its I/O performance –particularly in the output of large datasets. In this approach, a subset of MPI tasks being used for the UM job are dedicated to IO activity only -thereby freeing the remaining MPI tasks to concentrate on "normal" computational work. This approach has been successfully adopted in other codes (Edwards & Roy, 2010) but requires proper configuration of the IO servers for optimal performance. This fine-tuning involves manipulating a number of parameters such as the number of active IO servers, number of MPI tasks dedicated to each IO server, frequency at which data is passed to IO servers and so forth. A strong scaling analysis was performed on UM-N512L70 with and without the use of IO servers as shown in Figure 3. All runs lasted one model day. We define strong scaling factor as the logarithmic base 2 of the ratio of the recorded run time at each number of CPU cores to the lowest number of CPU cores used (256 in this case).

$$\log_2\left(\frac{runtime_i}{runtime_{256\ CPUs}}\right), i = \#\ of\ cores\ used\ \in \{256, 512, 1024, 2048, 4096\}$$

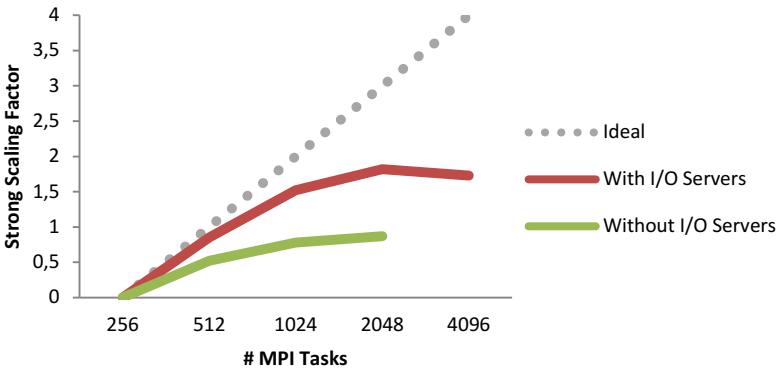

Fig. 3. Strong scaling for the N512L70 global UM configuration

The use of IO servers reduces observed runtimes while also improving scalability as shown in Figure 4.

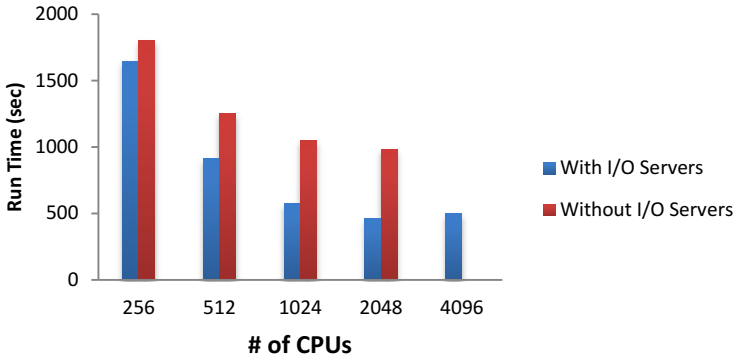

Fig. 4. Observed run times for the N512L70 global UM configuration

Strong scaling significantly improves up to 2048 cores with IO servers. At 4096 cores, observed runtime increases again due to the design of the IO server implementation in UM. Output data is replicated and passed between MPI tasks acting as IO servers in order to balance the workload among them. The "least-busy" IO server will output the next datafield to be written while the remaining IO server MPI tasks will flush data belonging to that field from their memory buffer and concentrate on other output datafields. However, this data replication/sharing can become excessive and can stall the overall writing process. A potential remedy would be incorporating MPI-IO into the IO server design. Instead of passing data between the individual IO servers, all the MPI tasks holding out the parts of the output datafield(s) could concurrently write to the same file.

The existing OpenMP implementation in UM is inefficient in a number of areas in the code –particularly in the convection routines. In such areas, 1 or more OpenMP threads are idle for long periods of time. Most existing OpenMP was added for loop and small-scale data parallelism only. Task-wise parallelism would be a more beneficial use of OpenMP and future work will focus on this. Also, some existing OpenMP use is architecture dependent (for the IBM Power architecture). Removing this dependence should yield significant performance benefits for Raijin.

The CPUs used in Raijin are capable of running more than one thread per physical core by using hyper-threading (Intel Corporation, 2014). In certain circumstances, enabling hyper-threading can increase application performance however this was not the case with the UM. Because of the low utilization of present OpenMP threads, hyper-threading is not necessary to effectively deal with extra processes spun up by OpenMPI and other system software.

Additional UM configurations to be assessed include the N768L85 global configuration that is scheduled to be part of the Bureau's next generation operational forecast suite in 2016/17. Performance issues flagged in UM-N512L70 (I/O, OpenMP) are expected to be more severe at this higher resolution.

3.2 UK MetOffice Data Assimilation Package (VAR)

In 2006, the UK MetOffice developed a 4D variational data assimilation (known simply as VAR) scheme for use in its operational weather forecasting suites (Rawlins, et al., 2007). This scheme has since been adopted by other organizations worldwide including the Bureau. In VAR, time series of observational data for certain parameters (such as temperature and pressure) are fitted to data from a Unified Model weather forecast. Fitting the observations involves repeated running of a simplified low resolution of the UM to assess the impact small changes to the model states has on the fit to the observations and to determine new perturbations to improve this fit. These simplified versions of the model are currently run at the horizontal resolutions of N108 and N216 (approximately 120km and 60km respectively) with the same number of vertical levels (70). Performance tests are conducted on version 30.0 of VAR.

Observed strong scaling is persistently good up to 384 MPI tasks as seen in Figure 5. After this point, performance depended on the number of MPI tasks assigned to each multicore CPU on Raijin. Observed strong scaling and associated runtimes continued favorably up to 1536 MPI tasks (the limit for domain decomposition) when only 4 MPI tasks were assigned to each 8-corPU (while still reserving the entire CPU). However, when the CPUs are fully populated with 8 MPI tasks, observed runtimes increased significantly at all MPI task counts as seen in Figure 6. At lower counts (eg, 192 and less), memory bandwidth contention is the main culprit for the increased runtimes. With large computational sub-domains and the use of three and four-dimensional data arrays, cache stalling is prevalent. At larger MPI task counts (over 384), OS jitter becomes apparent. The main control daemon for PBS (the job scheduling software used on Raijin) frequently interrupts the execution on a single CPU core on each node as it checks resource usage. It was observed that OpenMPI would spawn 4 hardware threads per MPI task it created. These threads are used to enable OpenMPI's asynchronous non-blocking communication calls as explained in (Wittmann, Hager, Zeiser, & Wellein, 2013). While they do overload physical CPU cores, we do not believe these additional OpenMPI threads contribute significantly to the observed OS jitter. Enabling hyper-threading (Intel Corporation, 2014) on the Intel CPUs alleviated the OS jitter issue by efficiently scheduling the non-compute processes onto logical threads. Observed runtimes dropped dramatically -at 1536 MPI tasks, there is now only a 12% difference in runtime between using full and half populated CPUs. Table 1 shows the observed run times for VAR as the number of MPI tasks is increased. It also highlights the OS jitter effect apparent when fully-allocated CPUs are utilized on the Raijin supercomputer (and how hyperthreading and under-allocating CPUs are effective workarounds).

Future VAR work includes the assessment of a higher resolution configuration using N320L70 of the outermost loop. No OpenMP was used in the current assessment. Addressing its effective use in VAR will be the target of upcoming work.

Table 1. Measured run times for the N216/108L70 configuration of VAR

# of MPI Tasks	Run Time (seconds)		
	4 MPI tasks / CPU (no hyperthreading)	8 MPI tasks / CPU (no hyperthreading)	8 MPI tasks / CPU (hyperthreading)
24	8751	15033	15315
48	4342	8057	7629
96	2180	4002	3828
192	1158	2000	1821
384	705	1425	1013
768	491	1586	941
1536	471	2440	539

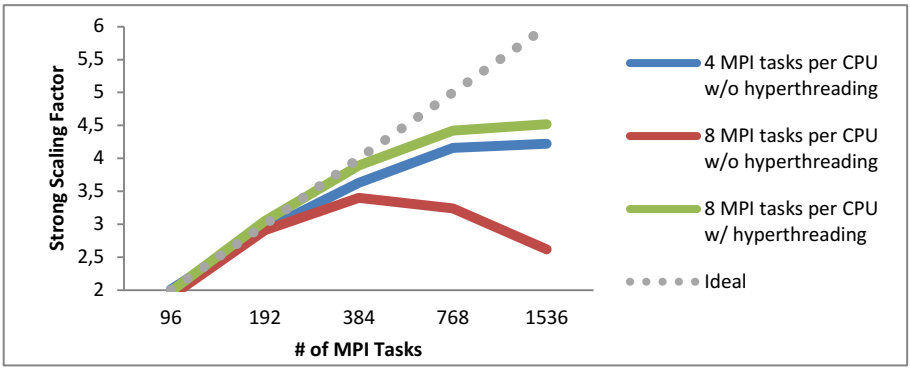

Fig. 5. Observed strong scaling for the N216/108L70 global VAR configuration

3.3 Modular Ocean Model

The Modular Ocean Model (MOM) (Griffies, Harrison, Pacanowski, & Rosati, 2004) is a general ocean circulation model developed at the Geophysical Fluids Dynamics Laboratory (GFDL) at Princeton University. It is a finite difference application written in FORTRAN90/95 using MPI distributed parallelization. Domain decomposition is performed along the horizontal dimensions only (eg. longitude and latitude). MOM is one of the main earth science models used by the ACCESS user community. It is employed for standalone ocean simulations and serves as the ocean component for coupled climate model configurations. Performance assessment work has targeted version 5.1 of MOM using a global configuration with a 0.25^{o} horizontal resolution and 50 model levels. Sea-ice representation is done by GFDL's Sea Ice Simulator model (Winton, 2001) that is coupled to MOM. Observed strong scaling of the 0.25^{o} global configuration is shown in Figure 6. The issue of an observed difference in

performance between using fully and partially populated CPUs on Raijin is present just like with VAR. The cause of the inflated runtimes (see Figure 6) when using 8 MPI tasks per CPU is the same as with VAR. The observed differences in performance at lower core counts is smaller than that observed with VAR. We believe that this is due to less memory bandwidth contention in MOM. At 1920 CPU cores, OS jitter becomes a serious impediment to performance. The same interrupting behavior from the PBS scheduler is still present. Enabling hyper-threading on the CPUs alleviated the problem just like with VAR.

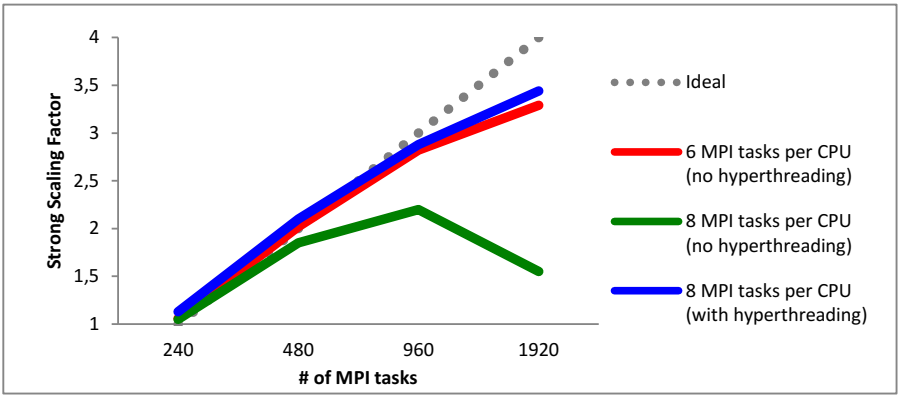

Fig. 6. Strong scaling results for the 0.25° global MOM configuration

Profiling showed that a significant number of calls to the MPI_Allreduce collective operation. Code modifications have reduced the frequency of these calls leading to lower runtimes. Introducing land masking (eg. ignoring land-filled grid cells during the integration steps) dropped observed run times another 20%. The final observed run times are given in Table 2 below.

Table 2. Observed run times for the 0.25o global MOM configuration

# of MPI Tasks	Run Time (seconds)		
	6 MPI tasks / CPU (no hyperthreading)	8 MPI tasks / CPU (no hyperthreading)	8 MPI tasks / CPU (hyperthreading)
120	4112	5298	5220
240	1970	2558	2393
480	1013	1469	1221
960	583	1152	707
1920	419	1810	480

Additional profiling on Raijin revealed that approximately 50% of all observed runtime is concentrated in three areas of the model: a) tracer advection, b) horizontal viscosity determination, and c) vertical mixing. These areas will be the focus of renewed optimization efforts in the upcoming year.

4 Future Work

Optimization work will continue on the applications and configurations previously discussed. NCI, Bureau, ARCCSS and Fujitsu staff frequently meet to discuss the project's status and to explore new goals. Higher resolution configurations are to be assessed for UM (N768L85), MOM (0.1^o horizontal resolution) and VAR (N320L70).

Emphasis will be placed on the performance assessment and optimization of version 1.4 of the coupled climate model ACCESS-CM (Bi, et al., 2013). This model couples UM 9.1, MOM 5.1 the sea ice mode, CICE 5.0 (Los Alamos National Laboratory, 2013) and the land surface model, CABLE 2.0 (Kowalczyk, Wang, Law, Davies, McGregor, & Abramowitz, 2006) via the OASIS3-MCT library (Valcke, Craig, & Coquart, 2013).

Acknowledgements. We wish to thank all the Bureau and CSIRO staff who have assisted us –in particular, Dr. Ilia Bermous, Dr Justin Freeman, Dr Martin Dix, Dr. Simon Marsland, Dr. Vicki Steinle and Dr Peter Steinle. We would also like to acknowledge several individuals from ARCCSS: Dr. Scott Wales, Prof. Andy Hogg and Dr. Nic Hannah.

Finally we extend a special thanks to ACCESS-Opt's steering committee for their continued guidance and support (Ben Evans and Mark Cheeseman of NCI, Robin Bowen and Tim Pugh of the Bureau, Ross Nobes and Tomohiro Yamada of Fujitsu).

References

Altair: PBS Works, `http://www.pbsworks.com` (retrieved October 23, 2014)

Home: ARC Centre of Excellence for Climate System Science, `http://www.climatescience.org.au` (retrieved October 23, 2014)

Bi, D., Dix, M., Marsland, S., O'Farrell, S., Rashid, H., Uotila, P., et al.: The ACCESS coupled model: description, control climate and evaluation. Australian Meteorological and Oceanographic Journal 63(1), 41–64 (2013)

Brown, A., Milton, S., Golding, B., Mitchell, J., Shelly, A.: Unified Modeling and Prediction of Weather and Climate A 25-Year Journey. American Meteorological Society, 1865–1877 (2012)

Cullen, M.J.: The unified forecast/climate model. Meteorological Magazine 122, 81-94

Edwards, T., Roy, K.: Using I/O Servers to Improve Application Performance on Cray XT Technology, Cray Users Group, pp. 1–4. Edinburgh (2010)

Gabriel, E., Fagg, G.E., Bosilca, G., Angskun, T., Dongarra, J.J., Squyres, J.M., et al.: Open MPI: Goals, Concept and Design of a Next Generation MPI Implementation. In: Proceedings 11th European PVM/MPI Users' Group Meeting, pp. 97–104. Budapest (2004)

Griffies, S.M., Harrison, M.J., Pacanowski, R.C., Rosati, A.: A Technical Guide to MOM4. Technical Report, Princeton University, Geophysical Fluids Dynamics Laboratory, Princeton (2004)

Intel Corporation, Intel Hyper-Threading Technology, from Intel Corporation website: `http://www.intel.com/content/www/us/en/architecture-and-technology/hyper-threading/hyper-threading-technology.html` (retrieved October 25, 2014)

Keenan, T., Puri, K., Pugh, T., Evans, B., Dix, M., Pitman, A., et al.: Next Generation Australian Community Climate and Earth-System Simulator (NGACCESS) - A Roadmap 2014-2019. Technical, Centre for Australian Weather and Climate Research, CAWCR (2014)

Kowalczyk, E., Wang, Y., Law, R., Davies, H., McGregor, J., Abramowitz, G.: The CSIRO Atmosphere Biosphere Land Exchange (CABLE) model for use in climate models and as an offline model. Technical, Commonwealth Scientific and Industrial Research Organisation (2006)

Los Alamos National Laboratory, The Los Alamos sea ice model (CICE) (2013), From COSIM: The Climate, Ocean and Sea Ice Modeling Group:
`http://oceans11.lanl.gov/drupal/CICE` (retrieved November 7, 2014)

Rawlins, F., Ballard, S.P., Bovis, K.J., Clayton, A.M., Li, D., Inverarity, W., et al.: The Met Office global four-dimensional variational data assimilation scheme. Quarterly Journal of the Royal Meteorological Society 133, 347–362 (2007)

Schlutter, M., Philippen, P., Morin, L., Geimer, M., Mohr, B.: Profiling Hybrid HMPP Applications with Score-P on Heterogeneous Hardware. In: Bader, M., Bode, A., Bungartz, H.-J., Gerndt, M., Joubert, G.R., Peters, F.J. (eds.) Parallel Computing: Accelerating Computational Science and Engineering (CSE), vol. 25, pp. 773–782. IOS Press (2014)

Valcke, S., Craig, T., Coquart, L.: OASIS3-MCT User Guide. Technical Report, CERFACS, CNRS, Toulouse (2013)

Winton, M.: FMS Sea Ice Simulator. Princeton University, Geophysical Fluid Dynamics Laboratory, Princeton (2001)

Wittmann, M., Hager, G., Zeiser, T., Wellein, G.: Asynchronous MPI for the Masses. Cornell University, Department of Computer Science, Ithaca (2013)