SEVENTH FRAMEWORK PROGRAMME
Research Infrastructures

INFRA-2011-2.3.5 – Second Implementation Phase of the European High Performance Computing (HPC) service PRACE

PRACE-2IP

PRACE Second Implementation Phase Project

Grant Agreement Number: RI-283493

D7.4
Unified European Applications Benchmark Suite

Final

Version: 1.0
Author(s): Mark Bull, EPCC
Date: 26.07.2013

Project and Deliverable Information Sheet

PRACE Project	**Project Ref. №: RI-283493**	
	Project Title: PRACE Second Implementation Phase Project	
	Project Web Site: http://www.prace-project.eu	
	Deliverable ID: < **D7.4**>	
	Deliverable Nature: <DOC_TYPE: Report / Other>	
	Deliverable Level: PU *	**Contractual Date of Delivery:** 31 / July / 2013
		Actual Date of Delivery: 31 / July / 2013
	EC Project Officer: Leonardo Flores Añover	

* - The dissemination level are indicated as follows: **PU** – Public, **PP** – Restricted to other participants (including the Commission Services), **RE** – Restricted to a group specified by the consortium (including the Commission Services). **CO** – Confidential, only for members of the consortium (including the Commission Services).

Document Control Sheet

Document	**Title: Unified European Applications Benchmark Suite**	
	ID: D7.4	
	Version: 1.0	**Status:** ***Final***
	Available at: http://www.prace-project.eu	
	Software Tool: Microsoft Word 2007	
	File(s): D7.4.docx	
Authorship	**Written by:**	Mark Bull, EPCC
	Contributors:	Stefanie Janetzko (FZJ) Dimitris Dellis, Marios Chatziangelou (GRNET) Ake Sandgren, Rickard Nilsson, Rossen Apostolov, Jing Gong, Jeff Ko, Chandan Basu (SNIC) Jorge Rodriguez, Christian Simarro (BSC) Rene Kalus, Vit Vondrak (VSB-TUO) Andrew Emerson (CINECA) Jan Christian Meyer (NTNU) Xu Guo (EPCC) Martti Louhivuori (CSC) Jeroen Engelberts (SURFsara) Eric Boyer, Tyra Van Olmen (CINES) Tilman Dannert, Andreas Marek (RZG) Maciej Szpindler (PSNC) Plamena Borovska (NCSA) Josip Jakic (IPB) Charles Moulinec (STFC)
	Reviewed by:	Giovanni Erbacci, CINECA Florian Berberich, JUELICH
	Approved by:	MB/TB

Document Status Sheet

Version	Date	Status	Comments
0.1	9th July 2013	Draft for internal review	
1.0	26th July 2013	Final version for approval	

Document Keywords

Keywords:	PRACE, HPC, Research Infrastructure, Benchmark Suite

Disclaimer

This deliverable has been prepared by the responsible Work Package of the Project in accordance with the Consortium Agreement and the Grant Agreement n° RI-283493. It solely reflects the opinion of the parties to such agreements on a collective basis in the context of the Project and to the extent foreseen in such agreements. Please note that even though all participants to the Project are members of PRACE AISBL, this deliverable has not been approved by the Council of PRACE AISBL and therefore does not emanate from it nor should it be considered to reflect PRACE AISBL's individual opinion.

Copyright notices

Table of Contents

Project and Deliverable Information Sheet i
Document Control Sheet i
Document Status Sheet ii
Document Keywords iii
Table of Contents iv
List of Figures v
List of Tables vi
References and Applicable Documents vi
List of Acronyms and Abbreviations vii
Executive Summary 1
1 Introduction 2
2 Selection Process 3
2.1 Shortlisting process and results 3
2.2 Final Selection Process and Results 10
3 Suite contents and sample results 12
3.1 ALYA 12
3.1.1 *Application and test cases* *12*
3.1.2 *Sample results* *13*
3.2 Code_Saturne 14
3.2.1 *Application and test cases* *14*
3.2.2 *Sample results* *15*
3.3 CP2K 16
3.3.1 *Application and test cases* *16*
3.3.2 *Sample results* *16*
3.4 GADGET 18
3.4.1 *Application and test cases* *18*
3.4.2 *Sample results* *19*
3.5 GENE 20
3.5.1 *Application and test cases* *20*
3.5.2 *Sample results* *20*
3.6 GPAW 22
3.6.1 *Application and test cases* *22*
3.6.2 *Sample results* *23*
3.7 GROMACS 25
3.7.1 *Application and test cases* *25*
3.7.2 *Sample results* *26*
3.8 NAMD 28
3.8.1 *Application and test cases* *28*
3.8.2 *Sample results* *29*
3.9 NEMO 30
3.9.1 *Application and test cases* *30*
3.9.2 *Sample results* *31*
3.10 QCD 32
3.10.1 *Application and test cases* *32*

3.10.2 *Sample results* 33
3.11 Quantum Espresso **37**
3.11.1 *Application and test cases* 37
3.11.2 *Sample results* 38
3.12 SPECFEM3D **40**
3.12.1 *Application and test cases* 40
3.12.2 *Sample results* 41
4 Conclusions and Future Work **42**

List of Figures

Figure 1 Execution time of ALYA for Test Case B 13
Figure 2 Performance per Peak-TFlop/s of ALYA for Test Case B 14
Figure 3 Execution time of Code_Saturne for Test Case A, SFC partitioner 15
Figure 4 Performance per Peak-TFlop/s of Code_Saturne for Test Case A, SFC partitioner 15
Figure 5 Execution time of CP2K for Test Case A 16
Figure 6 Performance per Peak-TFlop/s of CP2K for Test Case A 17
Figure 7 Execution time of CP2K for Test Case B 17
Figure 8 Performance per Peak-TFlop/s of CP2K for Test Case B 18
Figure 9 Execution time of GADGET for Test Case A 19
Figure 10 Performance per Peak-TFlop/s of GADGET for Test Case A 19
Figure 11 Execution time of GENE for Test Case A 20
Figure 12 Performance per Peak-TFlop/s of GENE for Test Case A 21
Figure 13 Execution time of GENE for Test Case B 21
Figure 14 Performance per Peak-TFlop/s of GENE for Test Case B 22
Figure 15 Execution time of GPAW for Test Case A 23
Figure 16 Performance per Peak-TFlop/s of GPAW for Test Case A 24
Figure 17 Execution time of GPAW for Test Case B 24
Figure 18 Performance per Peak-TFlop/s of GPAW for Test Case B 25
Figure 19 Execution time of GROMACS for Test Case A 26
Figure 20 Performance per Peak-TFlop/s of GROMACS for Test Case A 27
Figure 21 Execution time of GROMACS for Test Case B 27
Figure 22 Performance per Peak-TFlop/s of GROMACS for Test Case B 28
Figure 23 Execution time of NAMD for Test Case A 29
Figure 24 Performance per Peak-TFlop/s of NAMD for Test Case A 29
Figure 25 Execution time of NAMD for Test Case B 30
Figure 26 Performance per Peak-TFlop/s of NAMD for Test Case B 30
Figure 27 Execution time of NEMO for Test Case A 31
Figure 28 Performance per Peak-TFlop/s of NEMO for Test Case A 32
Figure 29 Execution time of QCD for Kernel A 33
Figure 30 Performance per Peak-TFlop/s of QCD for Kernel A 33
Figure 31 Execution time of QCD for Kernel B 34
Figure 32 Performance per Peak-TFlop/s of QCD for Kernel B 34
Figure 33 Execution time of QCD for Kernel C 35
Figure 34 Performance of QCD for Kernel C 35
Figure 35 Execution time of QCD for Kernel D 36
Figure 36 Performance per Peak-TFlop/s of QCD for Kernel D 36
Figure 37 Execution time of QCD for Kernel E 37
Figure 38 Performance per Peak-TFlop/s of QCD for Kernel E 37
Figure 39 Execution time of Quantum Espresso for Test Case A 38
Figure 40 Performance per Peak-TFlop/s of Quantum Espresso for Test Case A 39
Figure 41 Execution time of Quantum Espresso for Test Case B 39
Figure 42 Performance per Peak-TFlop/s of Quantum Espresso for Test Case B 40

List of Tables

Table 1 Voting Results 11
Table 2 List of systems used for benchmarking tests 12

References and Applicable Documents

[1] PRACE PP Deliverable D6.3.2 “Final Benchmark Suite”.

[2] DEISA Benchmark Suite http://www.deisa.eu/science/benchmarking

[3] PRACE 1IP Deliverable D7.4.2 “Benchmarking and Performance Modelling on Tier-0 Systems”.

[4] Chao Mei, Yanhua Sun, Gengbin Zheng, Eric J. Bohm, Laxmikant V. Kale, James C.Phillips and Chris Harrison, "Enabling and Scaling Biomolecular Simulations of 100 Million Atoms on Petascale Machines with a Multicore-optimized Message-driven Runtime", Proceedings of the ACM/IEEE Supercomputing Conference 2011.

[5] “The Millennium-XXL Project: Simulating the Galaxy Population of dark Energy Universes” http://inside.hlrs.de/htm/Edition_02_10/article_06.html

[6] “CP2K: High Performance Computing” http://www.nanosim.mat.ethz.ch/research/CP2K

[7] Roland Schulz, Benjamin Lindner, Loukas Petridis and Jeremy C. Smith, “Scaling of Multimillion-Atom Biological Molecular Dynamics Simulation on a Petascale Supercomputer”, *J. Chem. Theory Comput.*, 2009, 5 (10), pp 2798–2808.

[8] Jussi Enkovaara et al., “Electronic structure calculations with GPAW: a real-space implementation of the projector augmented-wave method”, *J. Phys.: Condens. Matter* **22** 253202, 2010.

[9] Joseba Alberdi Rodriguez et al., “Improving Octopus Towards the new Generation of HPC Systems”, in Technical Report on Jülich Blue Gene/P Extreme Scaling Workshop, 2011. http://www2.fz-juelich.de/jsc/docs/printable/ib/ib-11/ib-2011-02.pdf

[10] T. Dannert, T. Gorler, F. Jenko and F. Merz, “Gyrokinetic Turbulence Simulation with GENE”, in Technical Report on Jülich Blue Gene/P Extreme Scaling Workshop, 2009. http://www2.fz-juelich.de/jsc/docs/printable/ib/ib-10/ib-2010-02.pdf

List of Acronyms and Abbreviations

AVX	Advanced Vector Extensions
BSC	Barcelona Supercomputing Center (Spain)
CEA	Commissariat à l'Energie Atomique (represented in PRACE by GENCI, France)
CeCILL	CEA CNRS INRIA Logiciel Libre
CFD	Computational Fluid Dynamics
CINECA	Consorzio Interuniversitario, the largest Italian computing centre (Italy)
CINES	Centre Informatique National de l'Enseignement Supérieur (represented in PRACE by GENCI, France)
CPU	Central Processing Unit
CSC	Finnish IT Centre for Science (Finland)
CSCS	Swiss National Computing Centre (Switzerland)
DEISA	Distributed European Infrastructure for Supercomputing Applications. EU project by leading national HPC centres.
DFT	Density Functional Theory
ECMWF	European Centre for Medium-Range Weather Forecasts
EDF	Électricité de France
EP	Efficient Performance, e.g., Nehalem-EP (Intel)
EPCC	Edinburgh Parallel Computing Centre (represented in PRACE by EPSRC, United Kingdom)
EPSRC	The Engineering and Physical Sciences Research Council (United Kingdom)
ESSL	Engineering and Scientific Subroutine Library (IBM)
ETG	Electron Temperature Gradient
FP	Floating-Point
FZJ	Forschungszentrum Jülich (Germany)
GB	Giga (= 2^{30} ~ 10^{9}) Bytes (= 8 bits), also GByte
Gb/s	Giga (= 10^{9}) bits per second, also Gbit/s
GB/s	Giga (= 10^{9}) Bytes (= 8 bits) per second, also GByte/s
GENCI	Grand Equipement National de Calcul Intensif (France)
GFlop/s	Giga (= 10^{9}) Floating point operations (usually in 64-bit, i.e. DP) per second, also GF/s
GHz	Giga (= 10^{9}) Hertz, frequency =10^{9} periods or clock cycles per second
GPGPU	General Purpose Graphic Processing Unit
GPL	GNU General Public Licence
GPU	Graphic Processing Unit
GRNET	Greek Research and Technology Network (Greece)
HPC	High Performance Computing; Computing at a high performance level at any given time; often used synonym with Supercomputing
IBM	Formerly known as International Business Machines
I/O	Input/Output
IPB	Institute of Physics Belgrade (Serbia)
IS-ENES	InfraStructure for the European Network for Earth System Modelling
JKU	Johannes Kepler University of Linz (Austria)
JSC	Jülich Supercomputing Centre (FZJ, Germany)
KB	Kilo (= 2^{10} ~10^{3}) Bytes (= 8 bits), also KByte
KTH	Kungliga Tekniska Högskolan (represented in PRACE by SNIC, Sweden)
LES	Large Eddy Simulation

LRZ	Leibniz Supercomputing Centre (Garching, Germany)
MB	Mega (= 2^{20} ~ 10^{6}) Bytes (= 8 bits), also MByte
MB/s	Mega (= 10^{6}) Bytes (= 8 bits) per second, also MByte/s
MD	Molecular Dynamics
MFlop/s	Mega (= 10^{6}) Floating point operations (usually in 64-bit, i.e. DP) per second, also MF/s
MHD	Magneto-hydro dynamics
MHz	Mega (= 10^{6}) Hertz, frequency =10^{6} periods or clock cycles per second
MKL	Math Kernel Library (Intel)
MPI	Message Passing Interface
MPP	Massively Parallel Processing (or Processor)
NCF	Netherlands Computing Facilities (Netherlands)
NCSA	National Centre for Supercomputing Applications (Bulgaria)
NTNU	Norwegian University of Science and Technology (Norway)
OpenMP	Open Multi-Processing
OS	Operating System
PAW	Projector-Augmented Wave
PRACE	Partnership for Advanced Computing in Europe; Project Acronym
PSNC	Poznan Supercomputing and Networking Centre (Poland)
QCD	Quantum Chromodynamics
QR	QR method or algorithm: a procedure in linear algebra to compute the eigenvalues and eigenvectors of a matrix
RZG	Rechenzentrum Garching (Germany)
SARA	Stichting Academisch Rekencentrum Amsterdam (Netherlands)
SNIC	Swedish National Infrastructure for Computing (Sweden)
SP	Single Precision, usually 32-bit floating point numbers
SPH	Smooth Particle Hydrodynamics
SSE	Streaming SIMD Extensions
STFC	Science and Technology Facilities Council (represented in PRACE by EPSRC, United Kingdom)
STMV	Satellite Tobacco Mosaic Virus
SURFsara	Dutch National High Performance Computing & e-Science Support Center (Netherlands)
TB	Tera (= 240 ~ 1012) Bytes (= 8 bits), also TByte
TD-DFT	Time Dependent Density Functional Theory
TFlop/s	Tera (= 1012) Floating-point operations (usually in 64-bit, i.e. DP) per second, also TF/s
Tier-0	Denotes the apex of a conceptual pyramid of HPC systems. In this context the Supercomputing Research Infrastructure would host the Tier-0 systems; national or topical HPC centres would constitute Tier-1
UC-LCA	University of Coimbra Advanced Computing Laboratory (Portugal)
UEABS	Unified European Application Benchmark Suite
UYBHM	Ulusal Yuksek Basarimli Hesaplama Merkezi, Istanbul Technical University (Turkey)
VSB-TUO	Technical University of Ostrava (Czech Republic)

Executive Summary

This deliverable describes the Unified European Application Benchmark Suite (UEABS), a set of 12 application codes taken from the existing PRACE and DEISA application benchmark suites to form a single suite, with the objective of providing a set of scalable, currently relevant and publically available codes and datasets, of a size which can realistically be run on large systems, and maintained into the future. This work has been undertaken by Task 7.4 “Unified European Applications Benchmark Suite for Tier-0 and Tier-1” in the PRACE Second Implementation Phase (PRACE-2IP) project.

The selection process, which was determined by the consensus of partners in Workpackage 7, and which is described here in detail, took into account the criteria of open licensing, public availability of suitable datasets, support from developers, portability, scalability and relevance to science communities.

The selected codes are: **ALYA**, **Code_Saturne**, **CP2K**, **GADGET**, **GENE**, **GROMACS**, **GPAW**, **NAMD**, **NEMO**, **QCD**, **Quantum Espresso** and **SPECFEM3D**.

For each code either one or two test case datasets have been selected. These are described in this document, along with a brief introduction to the application codes themselves. For each code some sample results are presented, from several PRACE Tier-0 and Tier-1 systems.

1 Introduction

Both PRACE [1] and DEISA [2] projects have previously produced application benchmark suites. However, maintaining both of these suites, with a combined total of 29 application codes, requires a significant amount of effort. Furthermore, several of the application codes are out of date, in that they are no longer used nor supported by developers, or they are not sufficiently scalable to be useful benchmarks for current or future HPC hardware. Several of the applications and datasets in the PRACE suite, which was only intended for project internal use, do not have public licenses, or are not freely available.

The main purposes of the unified benchmark suite are the following:

- To provide a resource of application codes and datasets that PRACE partners can draw on for procurement purposes.
- To provide performance data on existing PRACE systems to assist users when choosing which system to apply for time on.
- To provide data for “currency conversion” of CPU hours between PRACE systems.

With these purposes in mind, the role of Task 7.4 in PRACE-2IP was therefore to select a subset of these applications for inclusion in a unified suite that would avoid the drawbacks noted above. The number of applications was chosen to be in the range 10-12 in order to provide a good coverage of application areas without being excessively burdensome on human resources for maintenance, or on machine resources for collecting data.

Section 2 describes in detail the selection process used to arrive at the final list of applications. Section 3 gives a brief description of each of the selected applications, together with the test case datasets, and presents some sample results from a number of PRACE systems. Section 4 outlines future work on, and using, the suite.

2 Selection Process

This section describes the process of selecting the contents of the benchmark suite.

2.1 Shortlisting process and results

The starting point for the shortlisting was to consider the 29 distinct application codes that are contained in the existing PRACE [1] and DEISA [2] benchmark suites. For each of these codes information was collected about the licensing status, availability of suitable datasets, portability and scalability of the codes.

The codes were evaluated against the following criteria:

- The code must be publically available, either unlicensed or with a suitable open license agreement.
- Suitable datasets must be publically available.
- The code must not have any significant barriers to portability.
- The code must demonstrate good scalability.
- The code must have active support by the developers.

In order to avoid difficulties comparing systems with different types of CPU core, scalability is defined in peak Tflop/s: a code is considered to scale to X Tflop/s if its performance on a machine partition with a peak of X Tflops is at least 1.7 times the performance on a partition of the same machine of half the size (i.e. X/2 Tflop/s peak). This definition is the same as used as in PRACE-PP and PRACE-1IP work on benchmarking. For benchmarks to be relevant to current and future Tier-0 systems, applications which scale to less than 100 Tflop/s (equivalent to 10% of the cores of a Petaflop machine) under the above definition should not be included in the suite unless there are other compelling reasons to do so. Much of the scalability data for codes in the existing PRACE suite was taken from benchmark runs conducted on the JUGENE and CURIE Tier-0 systems, and reported in PRACE-1IP Deliverable D7.4.2 [3]. However, where this data was recognised not to represent the highest scalability known for the application, other sources have been used and referenced.

It was decided at this stage not to consider accelerator-enabled versions of codes, since there is currently no widely adopted vendor-neutral programming model for these devices.

Each application is now considered in turn, reporting the information gathered and the reasons for the subsequent shortlisting decision.

Application code: QCD
Origin: PRACE
Science Area: QCD
License type: Open
Portability issues: None
Scalability: Up to 50 Tflop/s on BlueGene/P [3]
Coverage: European
Comments: Consists of 5 separate kernels: some scale up to 220 Tflop/s
Shortlisting decision: Accept into shortlist, though the kernels with poorer scaling, including

one that is a weak scaling test, should be removed.

Application code: Quantum Espresso
Origin: DEISA, PRACE
Science Area: Quantum MD
License type: Open (GPL)
Portability issues: None
Scalability: 220 Tflop/s on BlueGene/P [3]
Coverage: European
Comments: none
Shortlisting decision: Accept into shortlist.

Application code: NAMD
Origin: DEISA, PRACE
Science Area: Classical MD
License type: Open (registration required)
Portability issues: None
Scalability: 1 Pflop/s on Cray XT5 [4]
Coverage: Global
Comments: Installation can be time consuming: depends on CHARM++ library.
Shortlisting decision: Accept into shortlist.

Application code: CPMD
Origin: DEISA, PRACE
Science Area: Quantum MD
License type: Open
Portability issues: None
Scalability: 20 Tflop/s on Bull x86 cluster [3]
Coverage: European
Comments: No longer under active support/development.
Shortlisting decision: Reject: code is obsolete and scalability is not good enough.

Application code: Code_Saturne
Origin: PRACE
Science Area: CFD
License type: Open
Portability issues: None
Scalability: 55 TFlop/s on BlueGene/P (110 TFlop/s with factor 1.5) [3]
Coverage: European
Comments: None
Shortlisting decision: Accept into shortlist: scaling is marginal, but can be addressed with a larger dataset.

Application code: GADGET
Origin: DEISA, PRACE
Science Area: Astrophysics
License type: Open (see comments below)
Portability issues: None
Scalability: 140 TFlop/s on Sun x86 Cluster [5]
Coverage: European

Comments: The version in the PRACE and DEISA suites is GADGET 2, which does not show good scalability. The scalability quoted above is for GADGET 3, which is currently restricted. However, GADGET 3 was expected to be available as open source in the first half of 2012, and the developers indicated a willingness to make a benchmarking version available for distribution in UEABS.
Shortlisting decision: Accept into shortlist (as GADGET 3).

Application code: EUTERPE
Origin: PRACE
Science Area: Fusion Physics
License type: Available on application to developers. Developers have indicated that they are willing to make code available for UEABS.
Portability issues: None
Scalability: Version in PRACE scales to 160 TFlop/s on BlueGene/P [3] but this is obsolete. Scalability tests on current version by BSC are ongoing, but suggest that the 100 Tflop/s criterion can be met.
Coverage: European
Comments: See above
Shortlisting decision: Accept into shortlist.

Application code: WRF
Origin: PRACE
Science Area: Earth Sciences
License type: Open with registration
Portability issues: None
Scalability: 20 TFlop/s on BlueGene/P [3]
Coverage: Global
Comments: None
Shortlisting decision: Accept into shortlist: scaling is poor, but there is no better candidate in weather/climate science area. See Section 3 for subsequent update.

Application code: NEMO
Origin: DEISA, PRACE
Science Area: Earth Sciences
License type: Open (CeCILL) but dataset used for PRACE benchmark suite is not public domain.
Portability issues: None
Scalability: 10 TFlop/s on Bull x86 cluster [3]
Coverage: European
Comments: None
Shortlisting decision: Reject: scalability is poor and datasets are restricted, but see Section 3 for subsequent update.

Application code: CP2K
Origin: DEISA, PRACE
Science Area: Quantum MD
License type: Open (GPL)
Portability issues: None
Scalability: 120 TFlop/s on Cray XT5 [6]
Coverage: European
Comments: PRACE benchmark suite dataset does not scale well and needs replacing.

Shortlisting decision: Accept into shortlist.

Application code: GROMACS
Origin: PRACE
Science Area: Classical MD
License type: Open (GPL)
Portability issues: None
Scalability: ~170 TFlop/s on Cray XT5 [7]
Coverage: European
Comments: None
Shortlisting decision: Accept into shortlist.

Application code: NS3D
Origin: PRACE
Science Area: CFD
License type: Restricted
Portability issues: Primarily designed for vector systems
Scalability: 23 TFlop/s on Bull x86 cluster [3]
Coverage: European
Comments: None
Shortlisting decision: Reject: code not publically available.

Application code: AVBP
Origin: PRACE
Science Area: CFD
License type: Restricted
Portability issues: None
Scalability: No recent data
Coverage: European
Comments: None
Shortlisting decision: Reject: code not publically available.

Application code: HELIUM
Origin: DEISA, PRACE
Science Area: Quantum Physics
License type: Restricted: use requires developers' permission
Portability issues: None
Scalability: 40 TFlop/s on BlueGene/P [3]
Coverage: European
Comments: None
Shortlisting decision: Reject: code not publically available.

Application code: TRIPOLI_4
Origin: PRACE
Science Area: Computational Engineering
License type: Restricted: code is owned by CEA and not distributed
Portability issues: None
Scalability: No recent data
Coverage: European
Comments: None
Shortlisting decision: Reject: code not publically available.

Application code: PEPC
Origin: DEISA, PRACE
Science Area: Plasma Physics
License type: Open
Portability issues: None
Scalability: 110 TFlop/s on BlueGene/P [3]
Coverage: European
Comments: None
Shortlisting decision: Accept into shortlist.

Application code: GPAW
Origin: PRACE
Science Area: Quantum MD
License type: Open
Portability issues: None
Scalability: 140 TFlop/s on BlueGene/P [3], 210 TFlop/s on Cray XT5 [8]
Coverage: European
Comments: None
Shortlisting decision: Accept into shortlist.

Application code: ALYA
Origin: PRACE
Science Area: CFD
License type: Available on request to BSC. Became open in Q2 2012.
Portability issues: None
Scalability: 160 Tflop/s on BlueGene/P [3]
Coverage: European
Comments: None
Shortlisting decision: Accept into shortlist.

Application code: OCTOPUS
Origin: PRACE
Science Area: Quantum MD
License type: Open
Portability issues: None
Scalability: 110 TFlop/s on BlueGene/P [9]
Coverage: European
Comments: None
Shortlisting decision: Accept into shortlist.

Application code: BSIT
Origin: PRACE
Science Area: Earth Sciences
License type: Open
Portability issues: Originally designed for Cell BE, though a CPU port does exist.
Scalability: No recent data
Coverage: European
Comments: PRACE benchmark case is a weak-scaling task farm test.
Shortlisting decision: Reject: task farm style of parallelism is not appropriate.

Application code: ELMER
Origin: PRACE
Science Area: Computational Engineering
License type: Open (GPL)
Portability issues: None
Scalability: 20 TFlop/s on Bull x86 cluster [3]
Coverage: European
Comments: Has potential for better scalability but this has not yet been fully demonstrated. Serial mesh pre-processing is a problem for large benchmark runs.
Shortlisting decision: Reject.

Application code: SPECFEM3D
Origin: PRACE
Science Area: Earth Sciences
License type: Open (GPL)
Portability issues: None
Scalability: 55 TFlop/s on Bull x86 cluster
Coverage: European
Comments: Scalability is marginal
Shortlisting decision: Accept into shortlist.

Application code: RAMSES
Origin: DEISA
Science Area: Astrophysics
License type: Open (CeCILL)
Portability issues: None
Scalability: Known to run on several thousand cores, but lacking definitive published scalability data.
Coverage: European
Comments: None
Shortlisting decision: Reject.

Application code: FENFLOSS
Origin: DEISA
Science Area: CFD
License type: Restricted
Portability issues: Code tuned for vector systems
Scalability: No recent data
Coverage: European
Comments: None
Shortlisting decision: Reject: code not publically available.

Application code: IFS
Origin: DEISA
Science Area: Earth Sciences
License type: Use requires NDA contract with ECMWF
Portability issues: None
Scalability: No recent data
Coverage: European
Comments: None
Shortlisting decision: Reject: code not publically available.

Application code: IQCS
Origin: DEISA
Science Area: Informatics
License type: Open
Portability issues: None
Scalability: 33 TFlop/s on Cray XE6
Coverage: European
Comments: Code no longer active
Shortlisting decision: Reject: obsolete code.

Application code: GENE
Origin: DEISA
Science Area: Plasma Physics
License type: Open
Portability issues: None
Scalability: ~1 PFlop/s on BlueGene/P [10]
Coverage: European
Comments: None
Shortlisting decision: Accept into shortlist.

Application code: BQCD
Origin: DEISA
Science Area: QCD
License type: Open (GPL)
Portability issues: None
Scalability: N/A
Coverage: European
Comments: None
Shortlisting decision: Reject: kernel is included in PRACE QCD benchmark.

Application code: SU3_AHIGGS
Origin: DEISA
Science Area: QCD
License type: Open
Portability issues: None
Scalability: N/A
Coverage: European
Comments: None
Shortlisting decision: Reject: kernel is included in PRACE QCD benchmark.

The following list summarises the 15 benchmarks applications accepted into the shortlist, by science area:

Particle Physics (1): QCD
Classical MD (2): NAMD, GROMACS
Quantum MD (4): Quantum Espresso, CP2K, GPAW, OCTOPUS
CFD (2): Code_Saturne, ALYA
Earth Sciences (2): WRF, SPECFEM3D
Plasma Physics (3), GENE, EUTERPE, PEPC
Astrophysics (1): GADGET

2.2 Final Selection Process and Results

This section describes the final selection process that reduced the shortlist of 15 application codes to the final list. The intention was to produce a final list of 10-12 codes that are representative of the various science areas. At the WP7 Face-to-face meeting in Stockholm (29th/30th March 2012), it was decided to accept QCD and GADGET into the final list without further discussion, since they are the only shortlisted applications in their respective science areas. For the remaining codes, a voting process was undertaken to gauge the level of support for codes across the PRACE project partners. Each PRACE partner country was invited to allocate 100 points between the remaining 13 codes, weighted to express their preferences. This system was chosen to allow partner countries with multiple institutions to allocate subsets of the 100 points to different institutions. The voting process was open for a two-week period ending on 20th April 2012.

Table 1 shows the results of the voting process. Of the 20 partner countries, 16 returned votes. Austria (JKU) recorded an abstention, and no votes were received from Switzerland (CSCS), Turkey (UYBHM) or Portugal (UC-LCA).

At this point, following some discussions with the InfraStructure for the European Network for Earth System Modelling (IS-ENES) consortium, which represents climate application users in Europe, it became apparent that the criteria used to exclude NEMO from the shortlist for the benchmark suite were no longer valid. Recent improvements to the code showed reasonable scalability up to 8,000 cores on an SGI x86 cluster (around 95 TFlop/s peak), and the consortium indicated that the relevant dataset (ORCA12, which is the same as used in the PRACE-1IP benchmark tests) could be made publically available. In the light of this new information, it was proposed to reinstate NEMO in the benchmark suite, and remove WRF, which now showed poorer scaling than NEMO, is much less relevant to the European climate community, and was only shortlisted because there appeared at the time to be no other option in the same scientific area.

	Total Points	Non-zero votes	Germany (GCS)	France (GENCI)	UK (EPCC)	Spain (BSC)	Finland (CSC)	Netherlands (SARA)	Sweden (SNIC)	Italy (CINECA)	Poland (PSNC)	Norway (SIGMA)	Greece (GRNET)	Ireland (ICHEC)	Cyprus (CASTORC)	Bulgaria (NCSA)	Czech Republic (VSB)	Serbia (IPB)
Quantum Espresso	**261**	**13**	1	10	2				17	50	3	13	10	50	20	5	10	70
GROMACS	**215**	**14**	22		8		10	40	40	10	6	14	15	10	15	5	10	10
CP2K	**168**	**11**	22		18		10	30	7	10	18	13	20	10			10	
NAMD	**164**	**11**	7		2					5	50	20	15	10	25	10	10	10
SPECFEM3D	**145**	**9**	4	50	6	10		10		15			10			35	5	
Code_Saturne	**144**	**9**	4	15	52				7	5	1		15		15		30	
WRF	**129**	**7**			10			20	9			20	15		25	30		
GPAW	**129**	**6**	6				80		12		1	20						10
ALYA	**103**	**4**		10		60			8								25	
GENE	**78**	**6**	21	15	2					5	20					15		
EUTERPE	**30**	**1**				30												
OCTOPUS	**20**	**1**												20				
PEPC	**14**	**2**	13								1							

Table 1 Voting Results

A Workpackage 7 telephone conference was held on 26th April 2012 to finalize the list. At this meeting, the replacement of WRF with NEMO was approved, and it was decided to drop the two codes with the least support according to Table 1, namely PEPC and OCTOPUS. It was also decided that only one of EUTERPE and GENE should be included, but that the fusion physics community should be consulted as to their preference. This consultation, which included the heads of the groups that develop GENE and EUTERPE, resulted in a decision in favour of GENE.

The final list of 12 codes to form the initial version of UEABS is therefore:

Particle Physics (1): QCD
Classical MD (2): NAMD, GROMACS
Quantum MD (3): Quantum Espresso, CP2K, GPAW
CFD (2): Code_Saturne, ALYA
Earth Sciences (2): NEMO, SPECFEM3D
Plasma Physics (1): GENE
Astrophysics (1): GADGET

This final list was approved at a meeting of the PRACE Executive Board on 3rd May 2012.

3 Suite contents and sample results

In this section we give a brief description of the application codes and the datasets. For each code we have selected either one or two test case datasets. Where there are two test cases, the smaller one (Test Case A) is designed to run on Tier-1 systems and scale up to around 1000 x86 cores, or equivalent. The larger one (Test Case B) is designed to run on Tier-0 systems and scale up to 10000 x86 cores or equivalent. Where there is only one test case, it is suitable for both classes of system, and should also scale to 10000 x86 cores or equivalent.

We also present some sample results of running the codes on a variety of PRACE Tier-1 and Tier-0 systems. At this stage, due to constraints on staff effort and CPU resources, we have not attempted anything approaching complete coverage by running all the codes on some set of systems. This is work in progress, which will be continued under Task 7.3 in PRACE-3IP. Table 2 lists the systems used to obtain the results presented here. The first two systems (JUQUEEN and CURIE) are Tier-0 systems the remainder are all Tier-1.

For each code we present two different figures for each test case: first, a scaling plot, which is the execution time as a function of the number of CPUs (where “CPU” is used synonymously for “core”). Secondly, we show the performance (the reciprocal of execution time) per Peak-TFlop/s as function of the partition size in Peak-TFlop/s. In this figure the y-axis values are actually not meaningful but there are two reasons to show the data in this way:

(i) the shape of the curves is of interest: scaling is better if this curve does not decline too much, and ideal scaling is represented by a horizontal line, and

(ii) the performance of codes can be compared between systems.

Note that it is not possible to compare different codes on one platform using this metric.

Name	Partner	System	Total # of cores	Processor	Cores per node	Clock rate (GHz)	Peak Gflop/s per core
JUQUEEN	FZJ	IBM BlueGene/Q	458752	IBM PowerPC A2	16	1.60	12.8
CURIE (thin nodes)	GENCI	Bull X cluster	80640	Intel Sandy Bridge EP	16	2.70	21.6
HECToR	EPCC	CRAY XE6	90112	AMD Interlagos	32	2.30	9.2
Huygens	SARA	IBM p575 P6	3328	IBM Power5+	32	4.70	18.8
boreasz	ICM	IBM Power 775	2432	IBM Power7	32	4.14	33.12
HYDRA	RZG	IBM iDataPlex	9904	Intel Sandy Bridge EP	16	2.60	20.8
Abisco	Umea	Supermicro H8QG6	15456	AMD Interlagos	48	2.60	10.4
PLX	CINECA	IBM Dataplex C	3288	Intel Westmere	12	2.40	9.6
Lindgren	KTH	Cray XE6 MC	36384	AMD Magny-Cours	24	2.10	8.4
halo2	ICM	Sun Blade 6048 cluster	6912	AMD Barcelona	16	2.30	6.9

Table 2 List of systems used for benchmarking tests

3.1 ALYA

3.1.1 *Application and test cases*

The Alya System is a Computational Mechanics code capable of solving different physics, each one with its own modelization characteristics, in a coupled way. Among the problems it solves are: convection-diffusion reactions, incompressible flows, compressible flows, turbulence, bi-phasic flows and free surface, excitable media, acoustics, thermal flow, quantum mechanics (DFT) and solid mechanics (large strain). ALYA is written in fortran 90/95 and parallelised using MPI and OpenMP.

Web site: http://www.bsc.es/computer-applications/alya-system

Test Case A A 27 million element mesh representing the respiratory system.

Test Case B A 552.9 million element mesh of generic elements.

3.1.2 *Sample results*

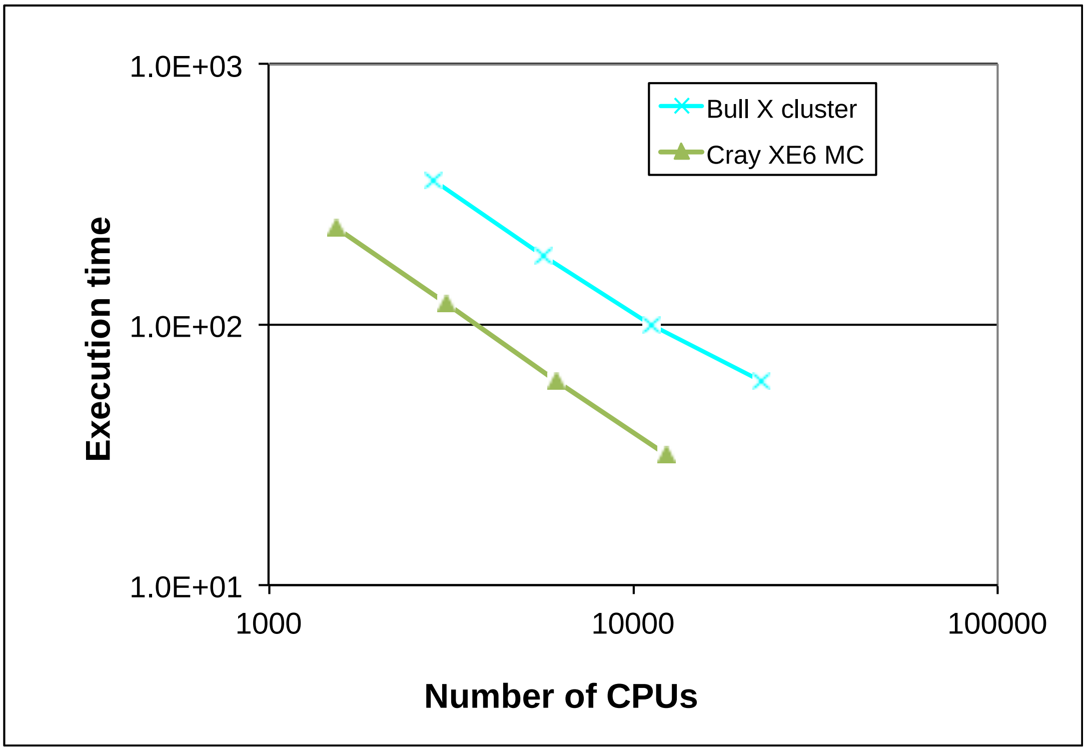

Figure 1 Execution time of ALYA for Test Case B

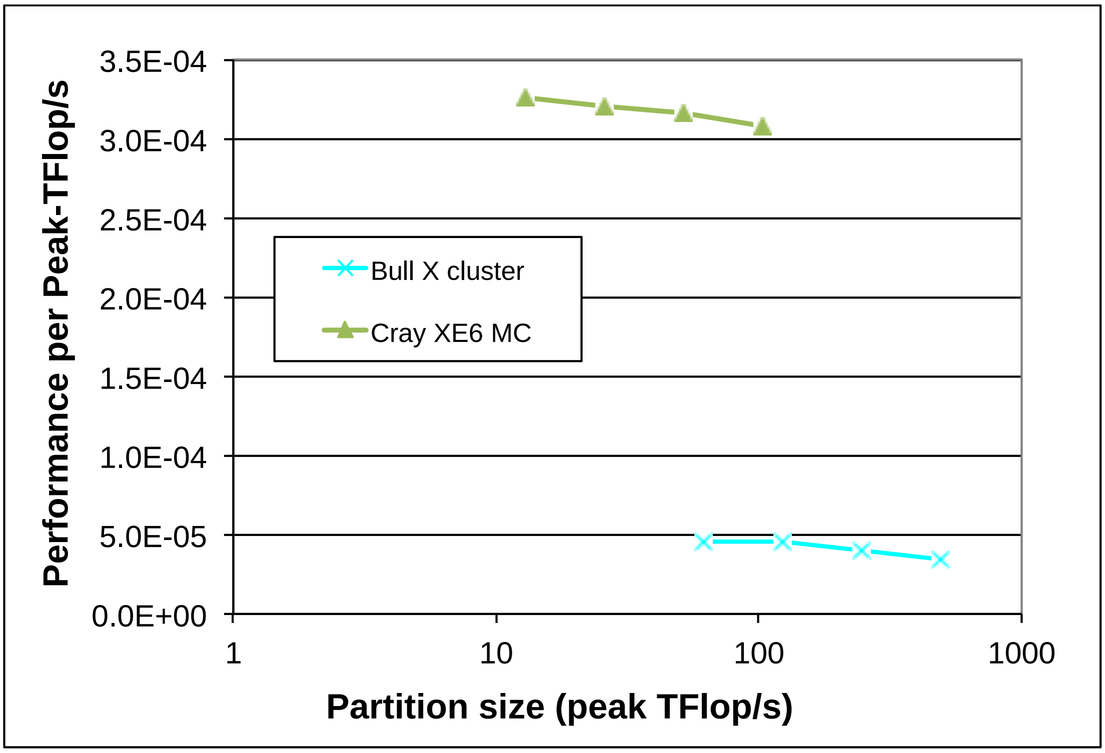

Figure 2 Performance per Peak-TFlop/s of ALYA for Test Case B

3.2 Code_Saturne

3.2.1 *Application and test cases*

Code_Saturne® is a multipurpose Computational Fluid Dynamics (CFD) software package, which has been developed by EDF (France) since 1997. The code was originally designed for industrial applications and research activities in several fields related to energy production; typical examples include nuclear power thermal-hydraulics, gas and coal combustion, turbo-machinery, heating, ventilation, and air conditioning. In 2007, EDF released the code as open-source and this provides both industry and academia to benefit from its extensive pedigree. *Code_Saturne*®'s open-source status allows for answers to specific needs that cannot easily be made available in commercial "black box" packages. It also makes it possible for industrial users and for their subcontractors to develop and maintain their own independent expertise and to fully control the software they use.

Code_Saturne® is based on a co-located finite volume approach that can handle three-dimensional meshes built with any type of cell (tetrahedral, hexahedral, prismatic, pyramidal, polyhedral) and with any type of grid structure (unstructured, block structured, hybrid). The code is able to simulate either incompressible or compressible flows, with or without heat transfer, and has a variety of models to account for turbulence. Dedicated modules are available for specific physics such as radiative heat transfer, combustion (e.g. with gas, coal and heavy fuel oil), magneto-hydro dynamics, and compressible flows, two-phase flows.
The software comprises of around 500 000 lines of source code, with around 50% written in Fortran90, 40% in C and 10% in Python. The code is parallelised using MPI.

Web site: http://code-saturne.org

Test Case A There is one dataset, representing a staggered bundle of five tubes. The mesh has 51 million of hexahedral cells, and the case has been designed to run using large eddy simulation (LES). Three different mesh partitioners are available: Space-Filling Curve (Morton), ParMETIS and PT-SCOTCH.

3.2.2 *Sample results*

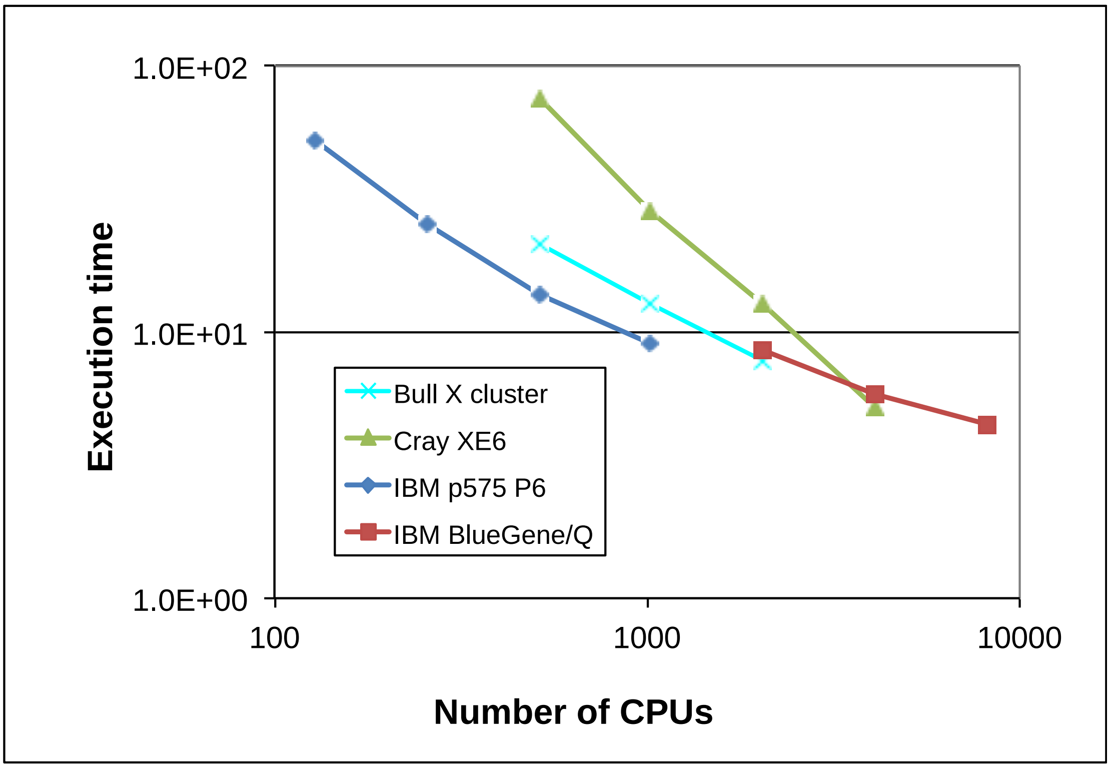

Figure 3 Execution time of Code_Saturne for Test Case A, SFC partitioner

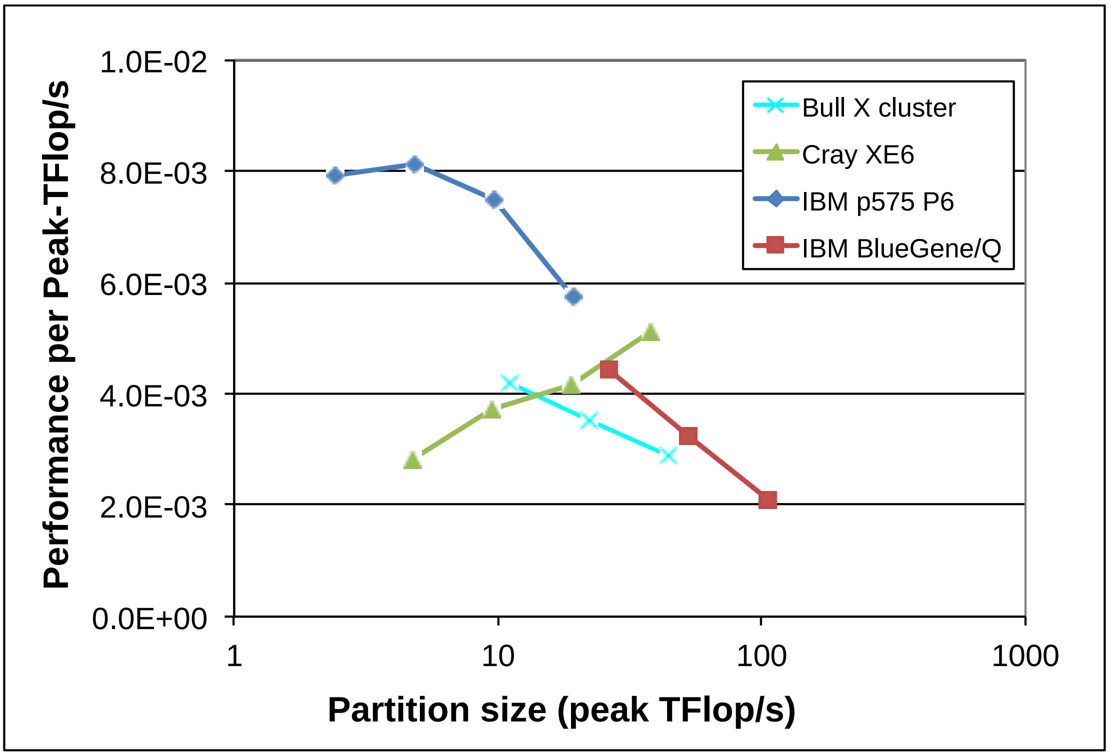

Figure 4 Performance per Peak-TFlop/s of Code_Saturne for Test Case A, SFC partitioner

3.3 CP2K

3.3.1 *Application and test cases*

CP2K is a freely available (GPL) program to perform atomistic and molecular simulations of solid state, liquid, molecular and biological systems. It provides a general framework for different methods such as e.g. density functional theory (DFT) using a mixed Gaussian and plane waves approach (GPW), and classical pair and many-body potentials. It is very well and consistently written, standards-conforming Fortran 95, parallelized with MPI and in some parts with hybrid OpenMP+MPI as an option.

CP2K provides state-of-the-art methods for efficient and accurate atomistic simulations, sources are freely available and actively improved. It has an active international development team, with the unofficial head quarters in the University of Zürich.

Web site: http://www.cp2k.org/

Test Case A The Tier-1 dataset is a single step energy calculation of 1024 Waters. This case is included in the CP2K distribution under cp2k/tests/QS/benchmark. The data file used in UEABS is slightly modified: instead of performing 10 MD steps, only 1 step, i.e. energy calculation, is performed in order to reduce the run time. In addition MAX_SCF, is increased to 200 for the single step energy calculation. The energy calculation takes 48 iterations to converge.

Test Case B The Tier-0 dataset is a 216 LiH system with Hartree-Fock Exchange. This type of calculation requires the libint-1.1.4 library, which is freely available, and whose source is included in the UEABS distribution.

3.3.2 *Sample results*

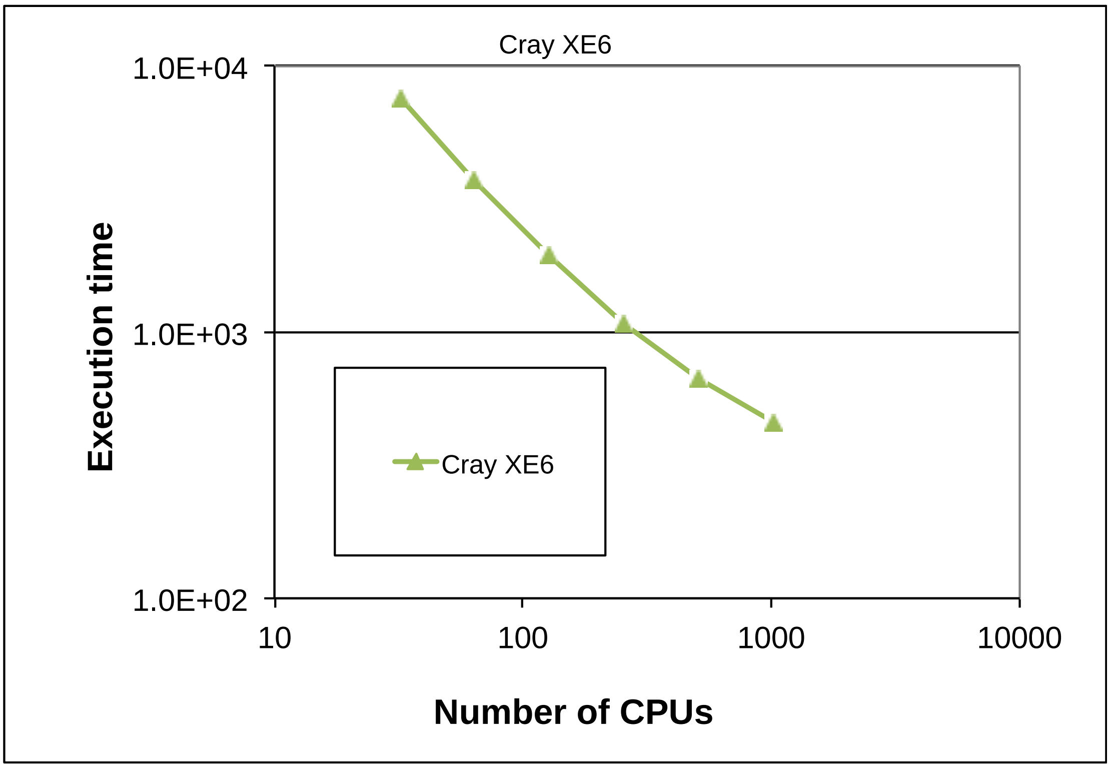

Figure 5 Execution time of CP2K for Test Case A

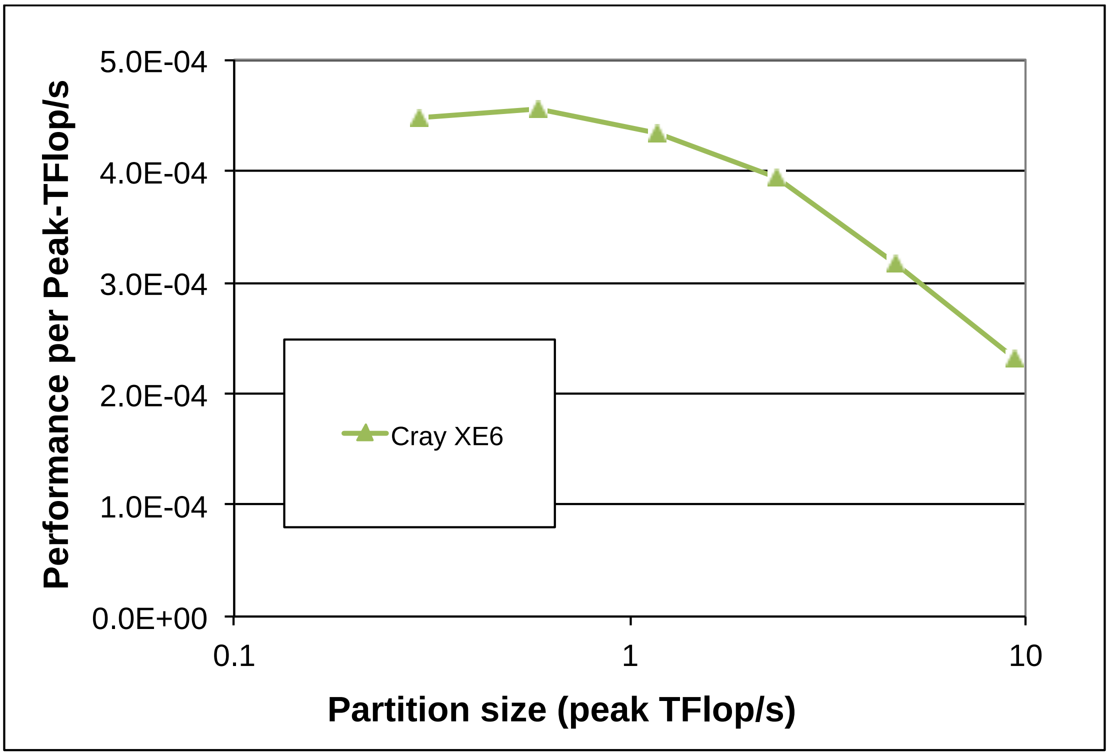

Figure 6 Performance per Peak-TFlop/s of CP2K for Test Case A

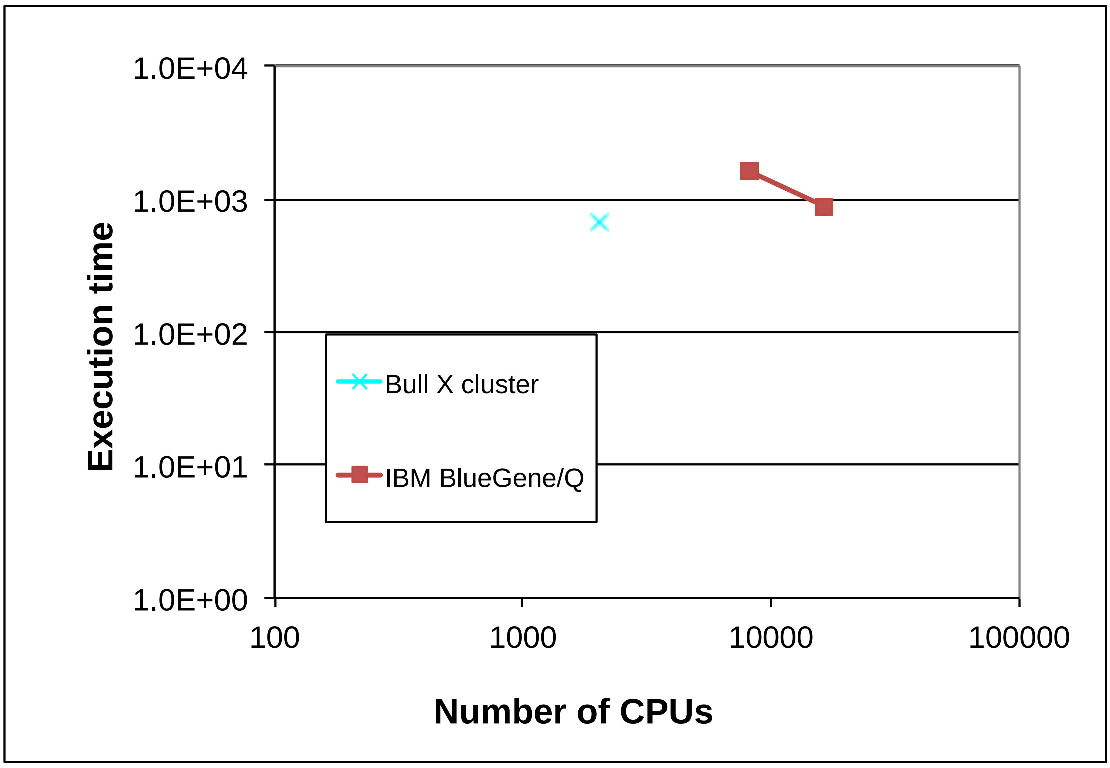

Figure 7 Execution time of CP2K for Test Case B

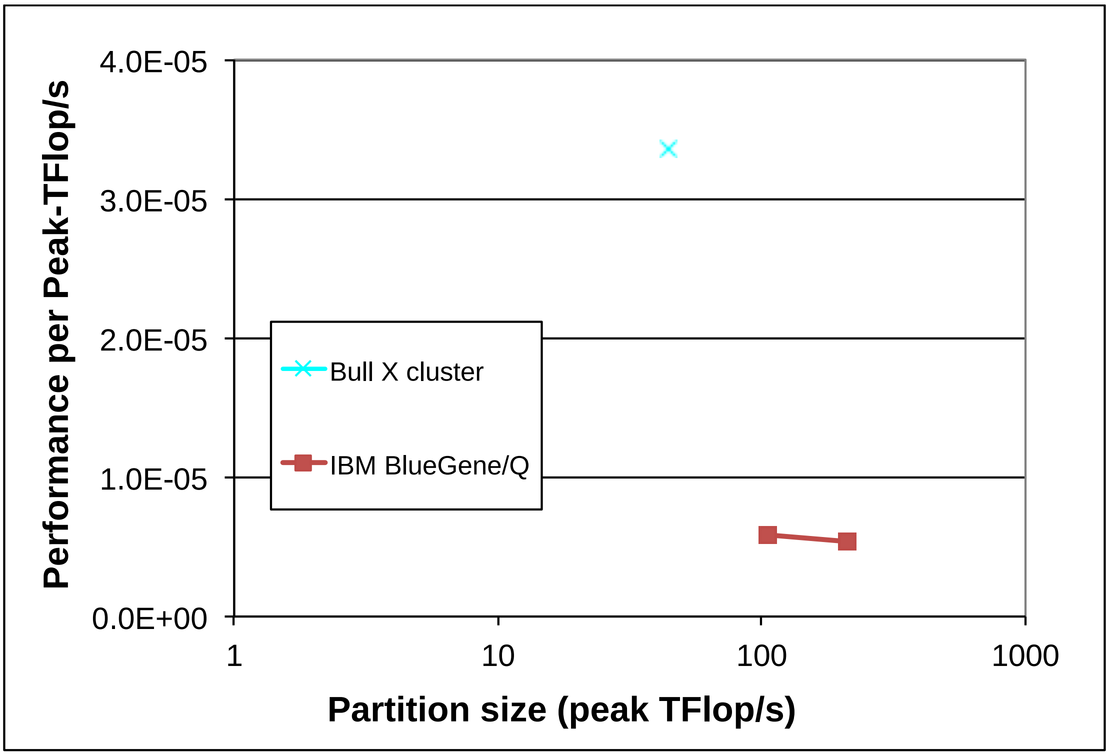

Figure 8 Performance per Peak-TFlop/s of CP2K for Test Case B

3.4 GADGET

3.4.1 *Application and test cases*

GADGET is a freely available code for cosmological N-body/SPH simulations on massively parallel computers with distributed memory written by Volker Springel, Max-Plank-Institute for Astrophysics, Garching, Germany. GADGET is written in C and uses an explicit communication model that is implemented with the standardized MPI communication interface. The code can be run on essentially all supercomputer systems presently in use, including clusters of workstations or individual PCs. GADGET computes gravitational forces with a hierarchical tree algorithm (optionally in combination with a particle-mesh scheme for long-range gravitational forces) and represents fluids by means of smoothed particle hydrodynamics (SPH). The code can be used for studies of isolated systems, or for simulations that include the cosmological expansion of space, either with, or without, periodic boundary conditions. In all these types of simulations, GADGET follows the evolution of a self-gravitating collisionless N-body system, and allows gas dynamics to be optionally included. Both the force computation and the time stepping of GADGET are fully adaptive, with a dynamic range that is, in principle, unlimited. GADGET can therefore be used to address a wide array of astrophysics interesting problems, ranging from colliding and merging galaxies, to the formation of large-scale structure in the Universe. With the inclusion of additional physical processes such as radiative cooling and heating, GADGET can also be used to study the dynamics of the gaseous intergalactic medium, or to address star formation and its regulation by feedback processes.

Web site: http://www.mpa-garching.mpg.de/gadget/

Test Case A: This case is a simulation consisting of 135 million particles.

3.4.2 *Sample results*

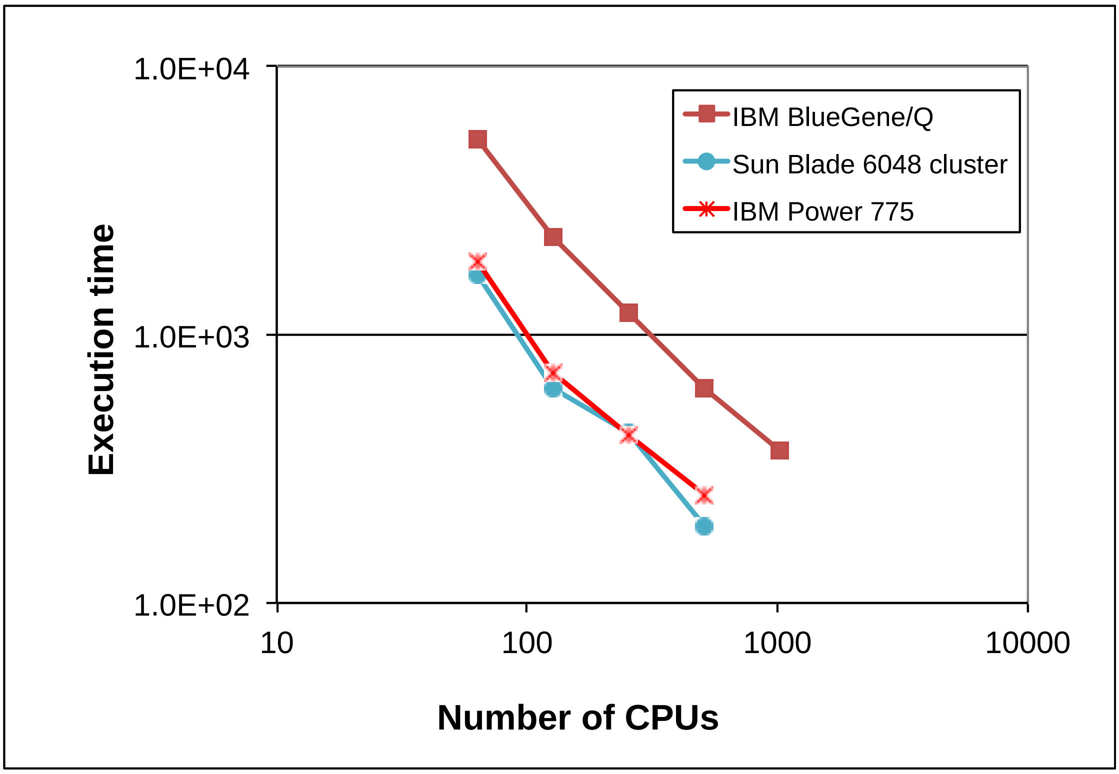

Figure 9 Execution time of GADGET for Test Case A

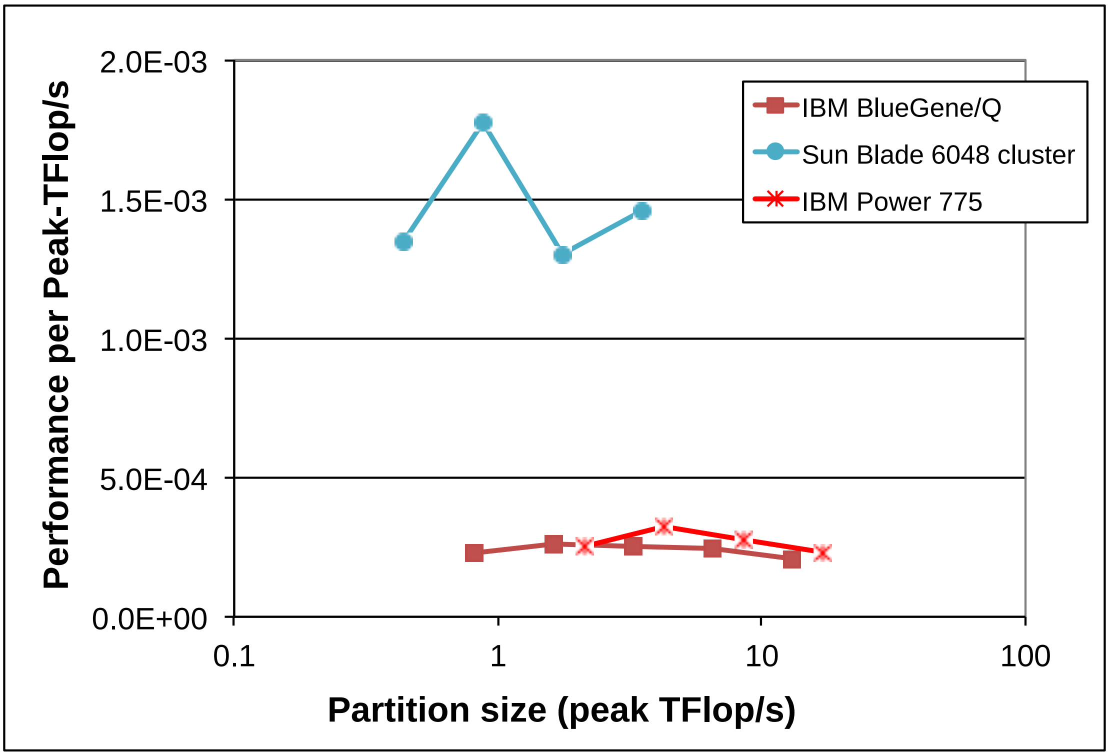

Figure 10 Performance per Peak-TFlop/s of GADGET for Test Case A

3.5 GENE

3.5.1 *Application and test cases*

GENE is a gyrokinetic plasma turbulence code which has been developed since the late 1990's and is physically very comprehensive and flexible as well as computationally very efficient and highly scalable. Originally used for flux-tube simulations, today GENE also operates as a global code, either gradient- or flux-driven. An arbitrary number of gyrokinetic particle species can be taken into account, including electromagnetic effects and collisions. GENE is, in principle, able to cover the widest possible range of scales, all the way from the system size (where nonlocal effects or avalanches can play a role) down to sub-ion-gyroradius scales (where ETG or microtearing modes may contribute to the transport), depending on the available computer resources. Moreover, there exist interfaces to various MHD equilibrium codes. GENE has been carefully benchmarked against theoretical results and other codes.

The GENE code is written in Fortran 90 and C and is parallelized with pure MPI. It strongly relies on a Fast Fourier Transform library and has built-in support for FFTW, MKL or ESSL. It also uses LAPACK and ScaLapack routines for LU decomposition and solution of a linear system of equations of moderate size (up to 1000 unknowns).

Web site: http:// gene.rzg.mpg.de

Test Case A This is a global simulation of ion-scale turbulence in Asdex-Upgrade, requiring 200-500GB total memory, and runnable from 256 to 4096 cores.

Test Case B This is a global simulation of ion-scale turbulence in JET, requiring 3.5-7TB total memory, and runnable from 4096 to 16384 cores.

3.5.2 *Sample results*

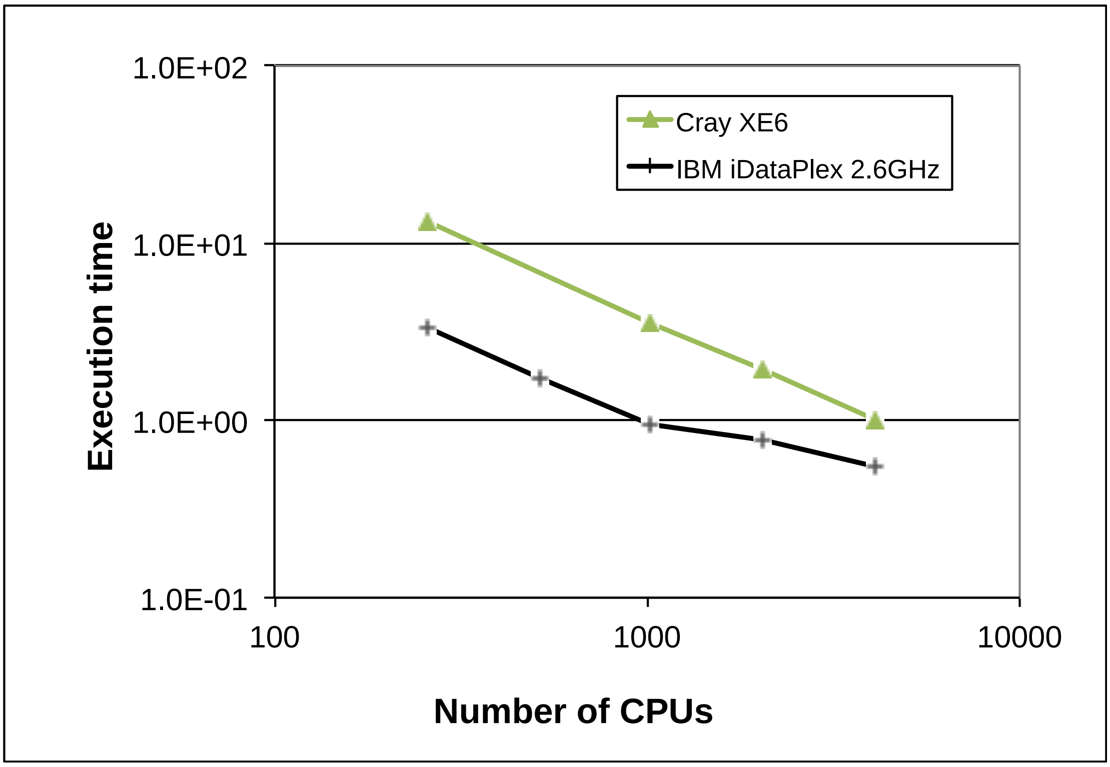

Figure 11 Execution time of GENE for Test Case A

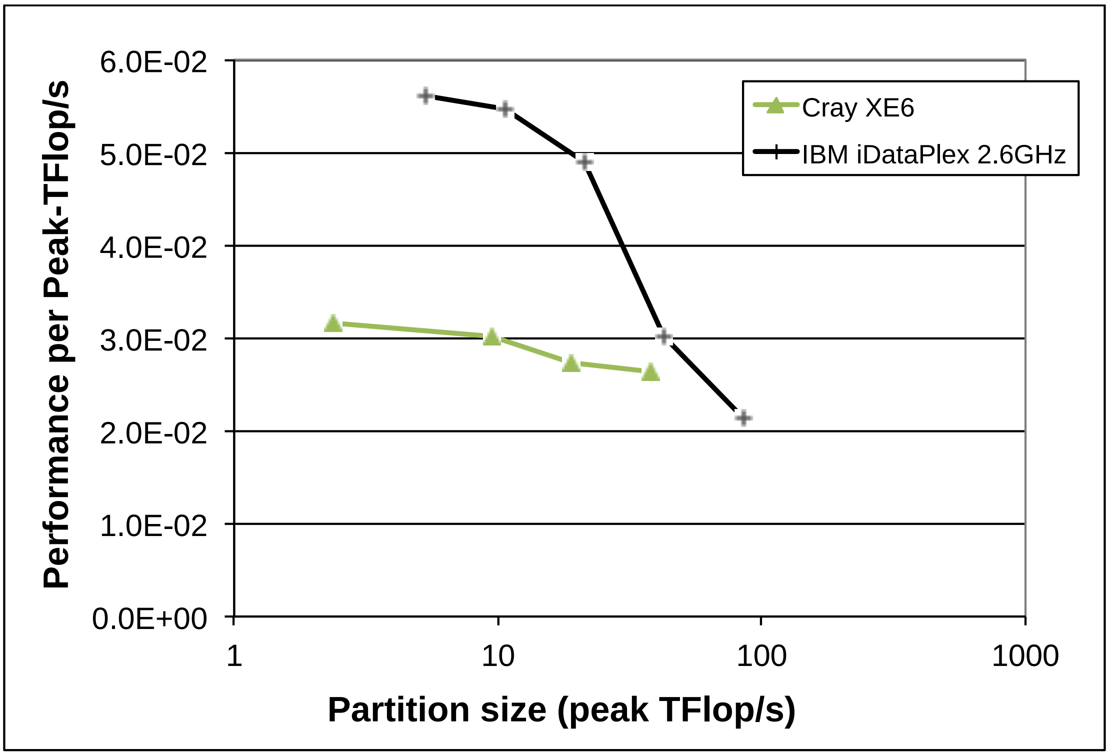

Figure 12 Performance per Peak-TFlop/s of GENE for Test Case A

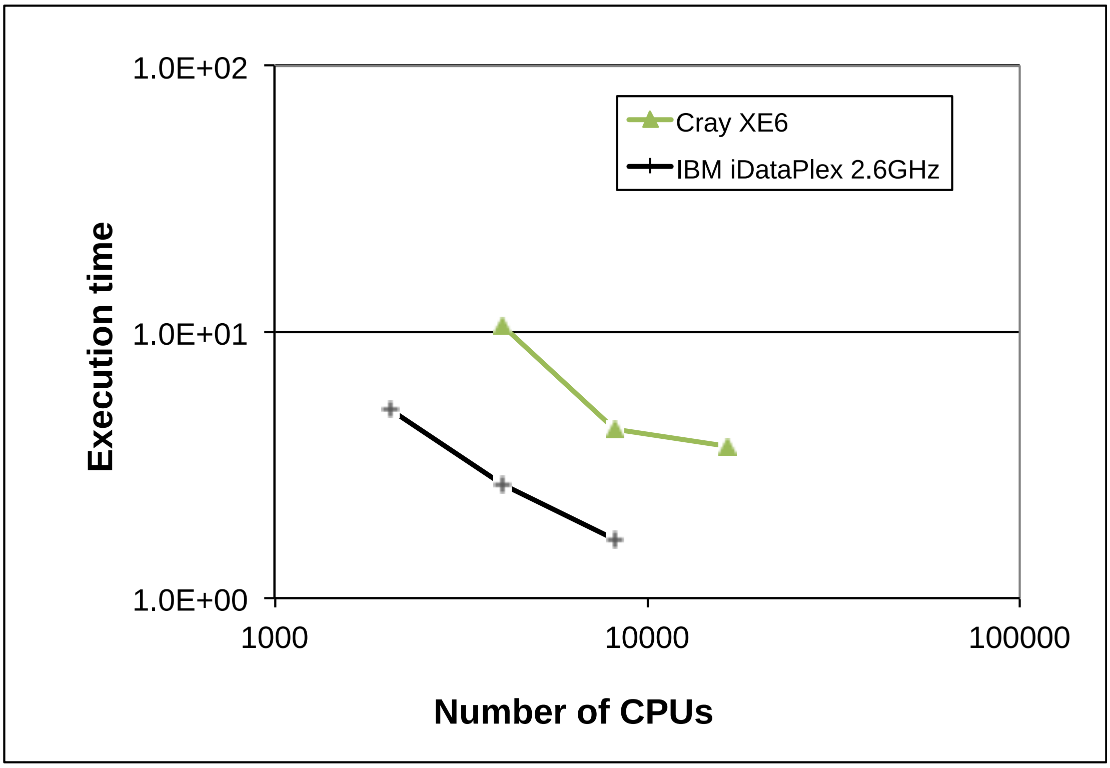

Figure 13 Execution time of GENE for Test Case B

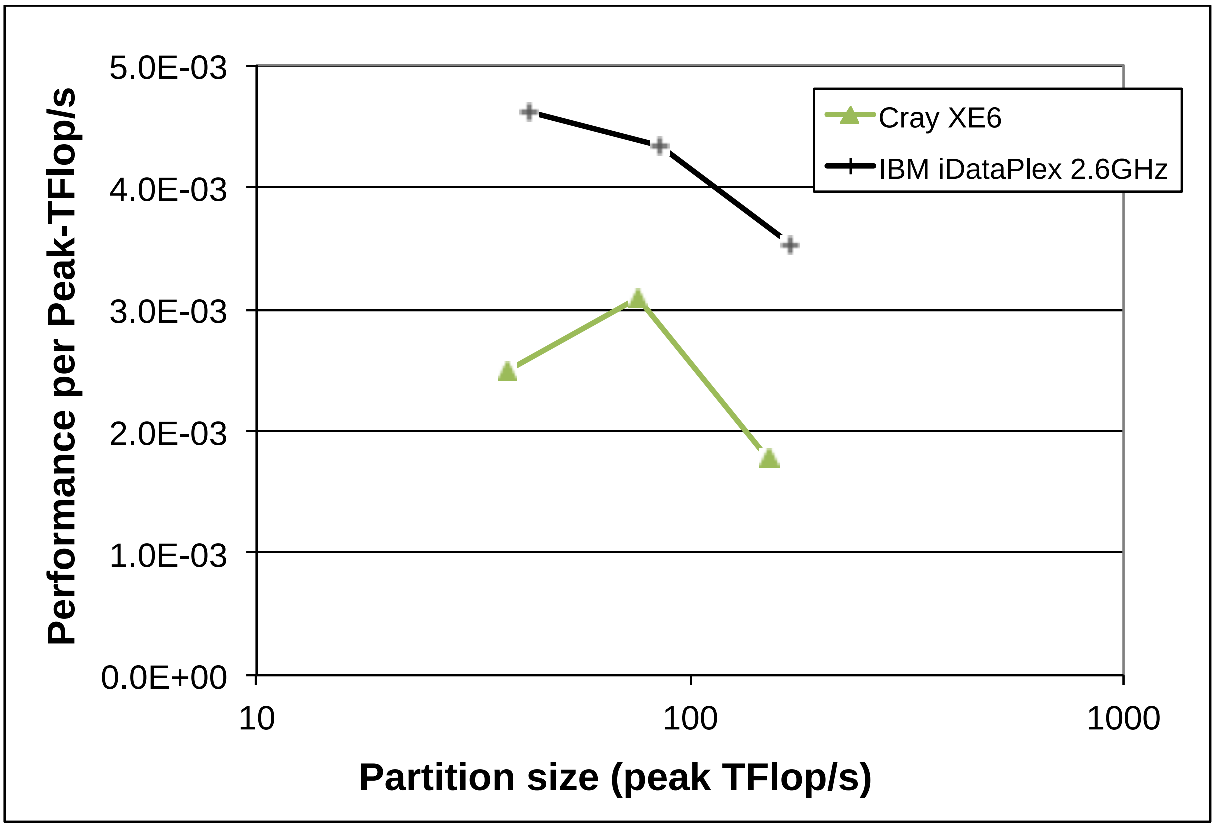

Figure 14 Performance per Peak-TFlop/s of GENE for Test Case B

3.6 GPAW

3.6.1 *Application and test cases*

GPAW is an efficient program package for electronic structure calculations based on the density functional theory (DFT) and the time-dependent density functional theory (TD-DFT). The density-functional theory allows studies of ground state properties such as energetics and equilibrium geometries, while the time-dependent density functional theory can be used for calculating excited state properties such as optical spectra. The program package includes two complementary implementations of time-dependent density functional theory: a linear response formalism and a time-propagation in real time.

The program uses the projector augmented wave (PAW) method that allows one to get rid of the core electrons and work with soft pseudo valence wave functions. The PAW method can be applied on the same footing to all elements, for example, it provides a reliable description of the transition metal elements and the first row elements with open p-shells that are often problematic for standard pseudopotentials. A further advantage of the PAW method is that it is an all-electron method (frozen core approximation) and there is a one to one transformation between the pseudo and all-electron quantities.

The equations of the (time-dependent) density functional theory within the PAW method are discretized using finite-differences and uniform real-space grids. The real-space representation allows flexible boundary conditions, as the system can be finite or periodic in one, two or three dimensions (e.g. cluster, slab, bulk). The accuracy of the discretization is controlled basically by single parameter, the grid spacing. The real-space representation allows also efficient parallelization with domain decomposition.

The program offers several parallelization levels. The most basic parallelization strategy is domain decomposition over the real-space grid. In magnetic systems it is possible to parallelize over spin, and in systems that have k-points (surfaces or bulk systems) parallelization over k-points is also possible. Furthermore, parallelization over electronic states is possible in DFT and in real-time TD-DFT calculations. GPAW is written in Python and C and parallelised with MPI.

Web site: https://wiki.fysik.dtu.dk/gpaw/

The test cases are a linear response time-dependent density-functional theory calculation of an Au_{38} cluster surrounded by CH_3S ligands. Total number of atoms in the system is 158, and there are 730 valence electrons in the system. The benchmark corresponds to the calculation of optical spectra up to ~2.5 eV. The benchmark utilizes domain decomposition with 128 cores, and the rest of cores are used for parallelization over electron-hole pairs in the construction of the Casida matrix.

Test Case A This is a basic DFT computation that scales up to about 2000 cores. As a by-product, it produces wave functions that are required as an input for Test Case B.

Test Case B A large TD-DFT computation that scales up to 10-100 thousand cores.

3.6.2 *Sample results*

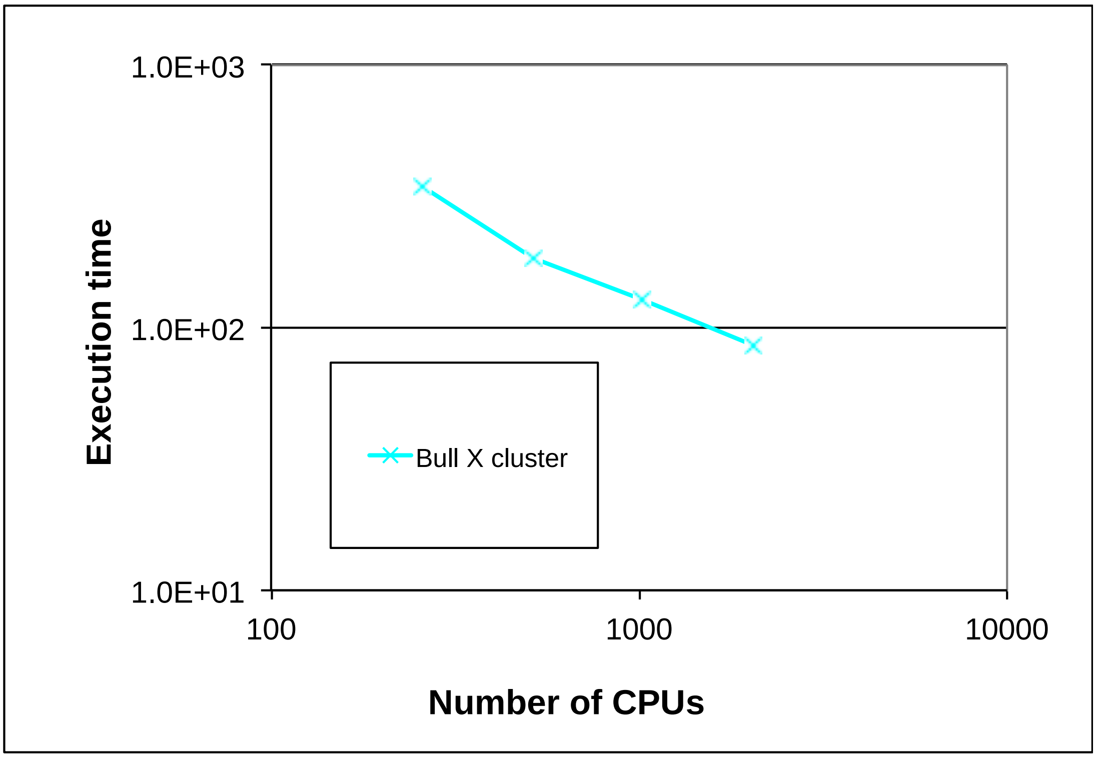

Figure 15 Execution time of GPAW for Test Case A

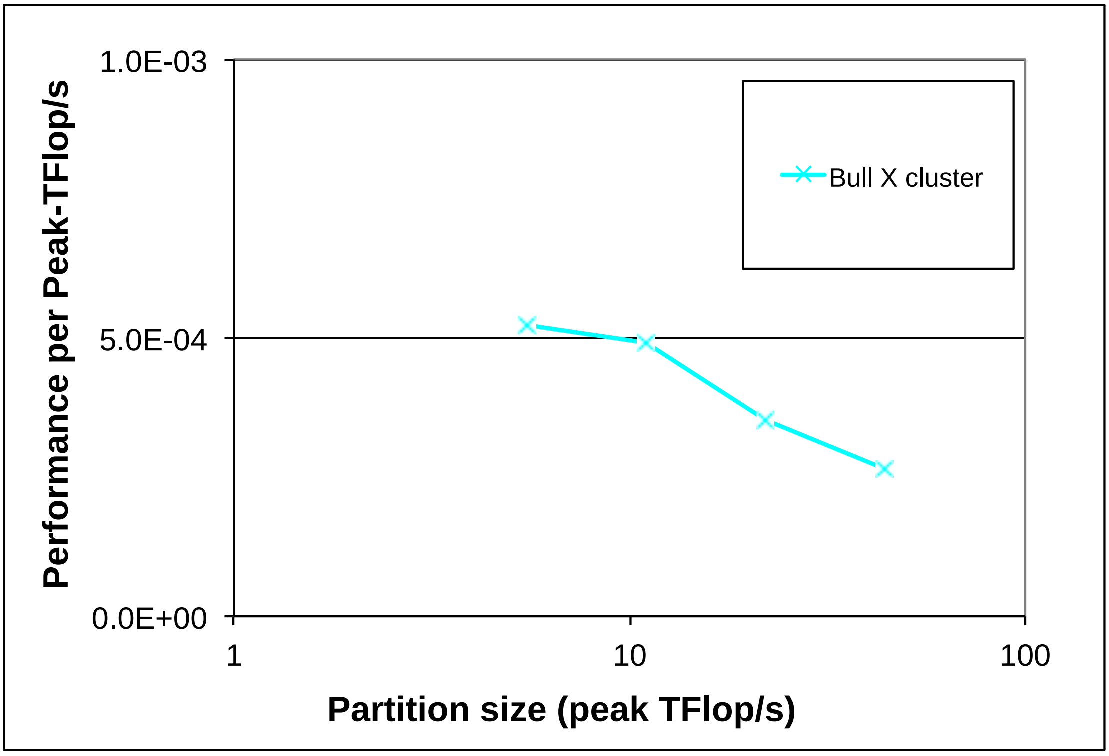

Figure 16 Performance per Peak-TFlop/s of GPAW for Test Case A

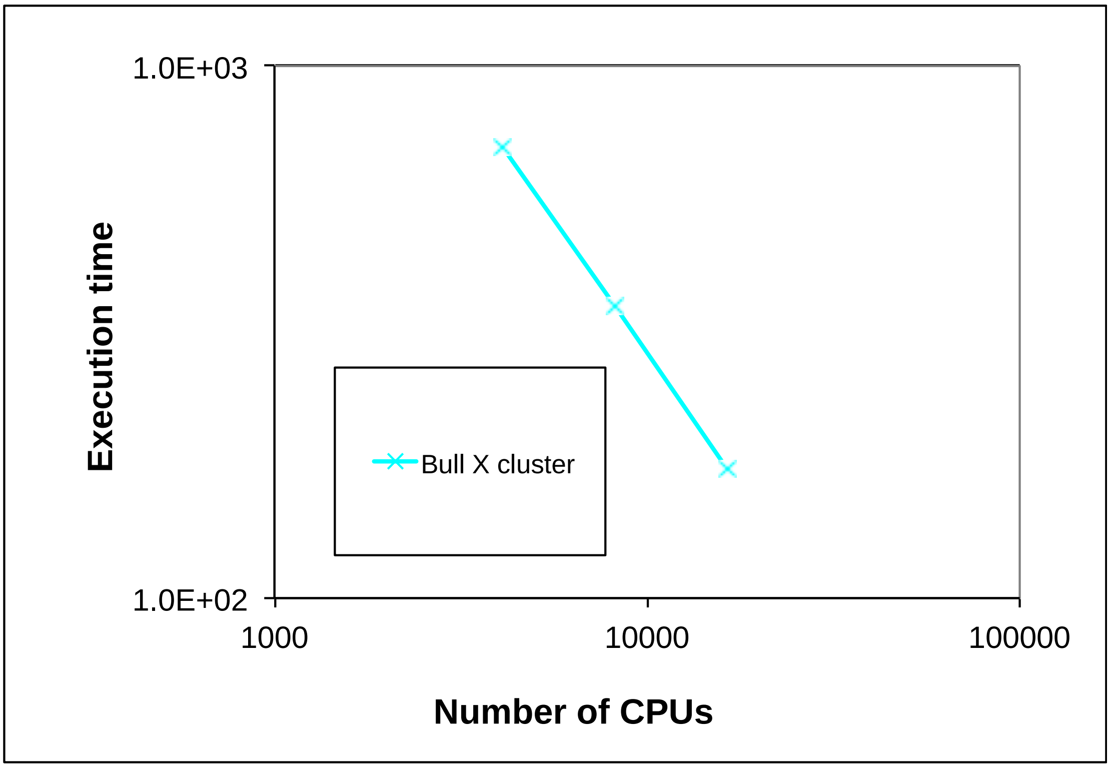

Figure 17 Execution time of GPAW for Test Case B

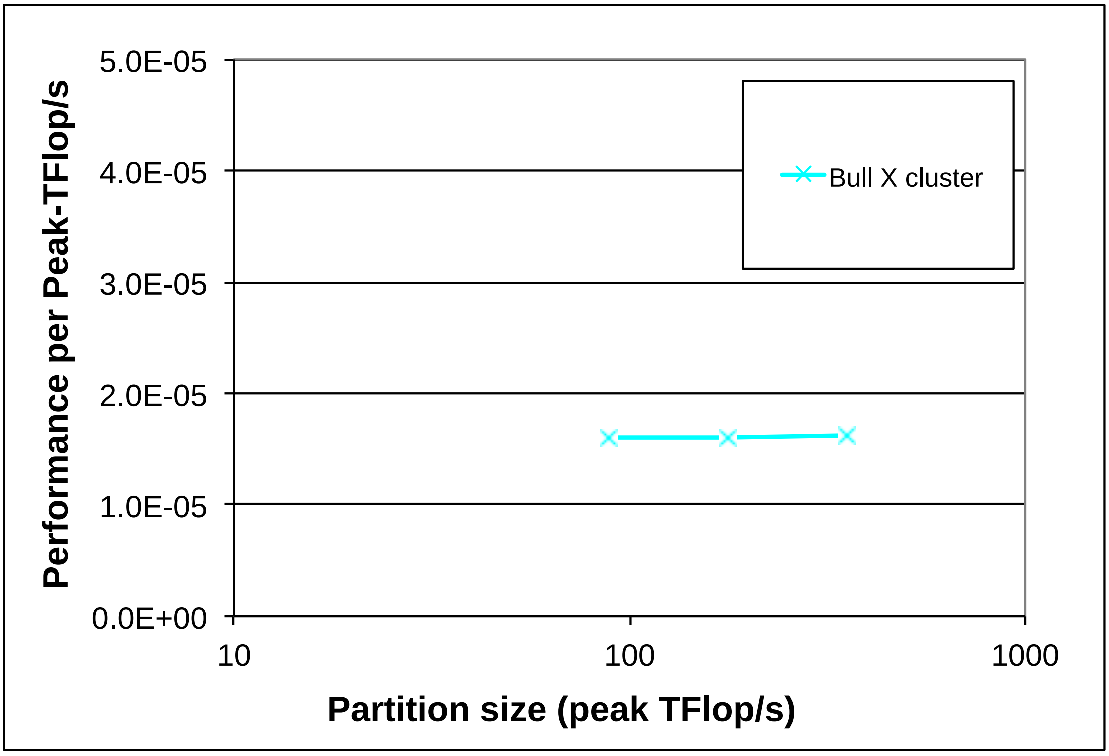

Figure 18 Performance per Peak-TFlop/s of GPAW for Test Case B

3.7 GROMACS

3.7.1 *Application and test cases*

GROMACS is a versatile package to perform molecular dynamics, i.e. simulate the Newtonian equations of motion for systems with hundreds to millions of particles. It is primarily designed for biochemical molecules such as proteins, lipids and nucleic acids that have a lot of complicated bonded interactions, but since GROMACS is extremely fast at calculating the non-bonded interactions (that usually dominate simulations) many groups also use it for research on non-biological systems, e.g. polymers.

GROMACS supports all the usual algorithms one might expect from a modern molecular dynamics implementation, but there are also quite a few features that make it stand out:

- GROMACS provides extremely high performance compared to all other programs. A lot of algorithmic optimizations have been introduced in the code. The innermost loops are written in assembly with optimized kernels for most common CPU extensions such as SSE/SSE2/SSE4 and AVX.
- Hybrid-CPU/GPGPU support for efficient usage of modern GPU hardware.
- GROMACS is user-friendly, with topologies and parameter files written in clear text format.
- There is no scripting language - all programs use a simple interface with command line options for input and output files.
- GROMACS is written in C and can be run in parallel, using standard MPI communication. Hybrid-MPI/OpenMP implementation is able to push the scaling limits for small to medium systems.

- GROMACS contains several state-of-the-art algorithms that make it possible to extend the time steps is simulations significantly, and thereby further enhance performance without sacrificing accuracy or detail.
- GROMACS is Free Software, available under the GNU General Public License.

Web site: http://www.gromacs.org

Test Case A The ion channel system is the membrane protein GluCl, which is a pentameric chloride channel embedded in a lipid bilayer. This system contains roughly 150,000 atoms, and is a quite challenging parallelization case due to the small size. However, it is likely one of the most wanted target sizes for biomolecular simulations due to the importance of these proteins for pharmaceutical applications.

It is particularly challenging due to a highly inhomogeneous and anisotropic environment in the membrane, which poses hard challenges for load balancing with domain decomposition.

Test Case B A model of cellulose and lignocellulosic biomass in an aqueous solution. This system of 3.3M atoms is inhomogeneous, at least with GROMACS 4.5. This system uses reaction-field electrostatics instead of PME and therefore should scale well.

3.7.2 *Sample results*

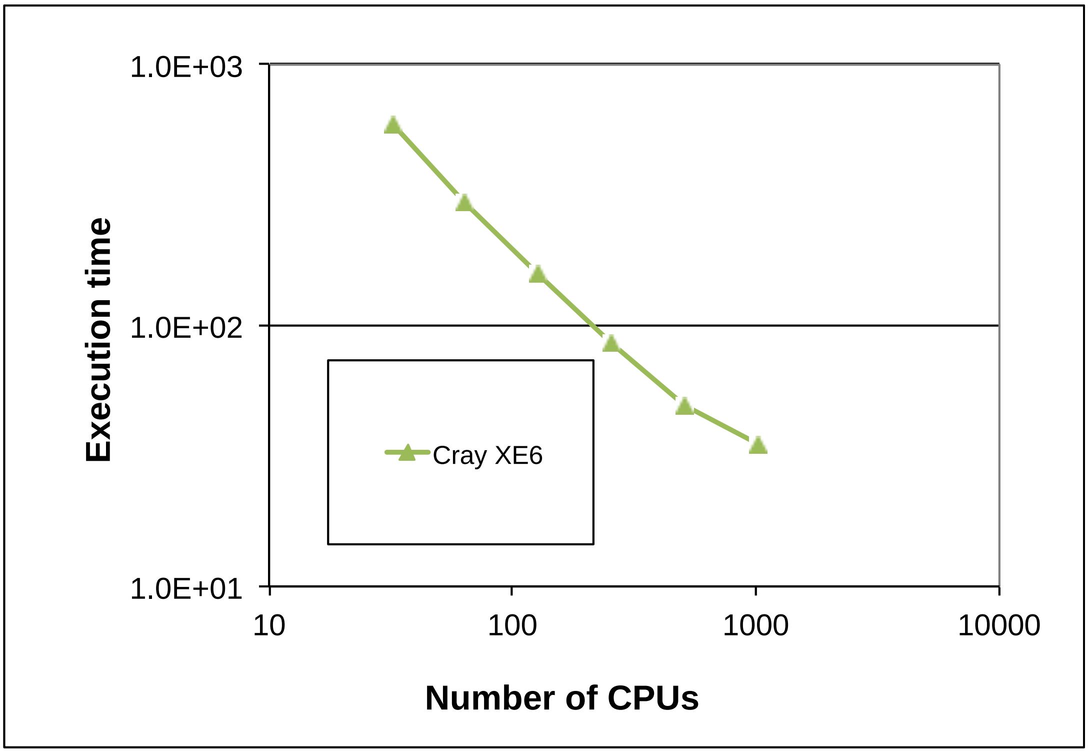

Figure 19 Execution time of GROMACS for Test Case A

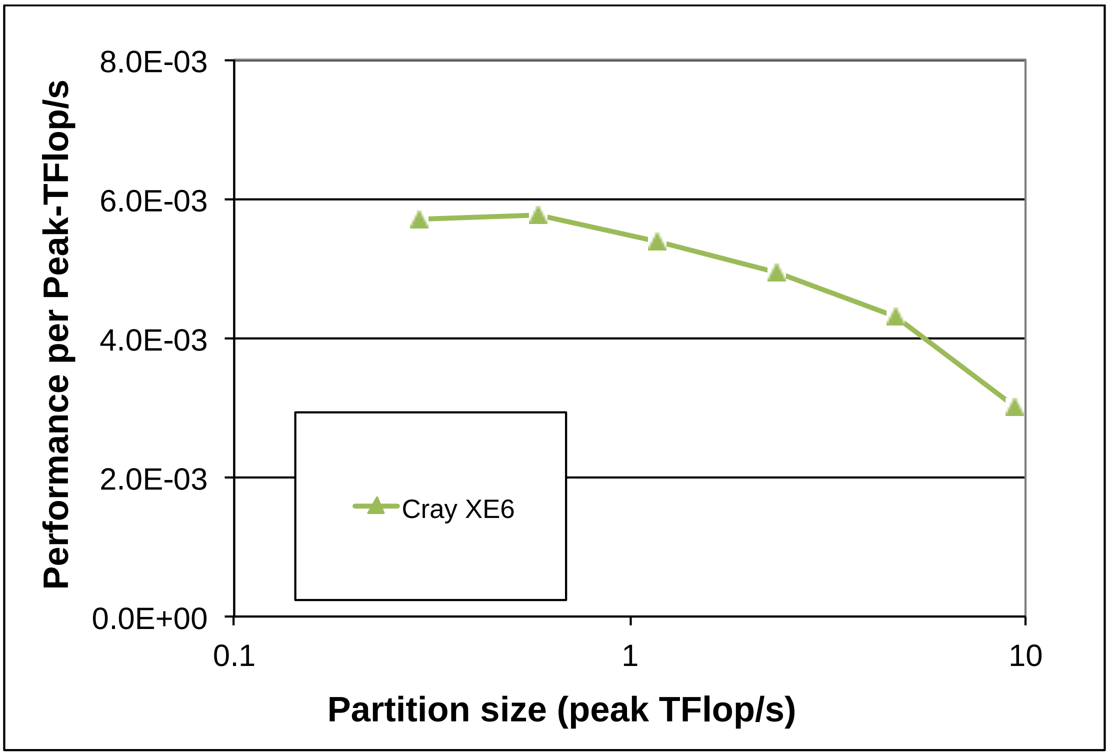

Figure 20 Performance per Peak-TFlop/s of GROMACS for Test Case A

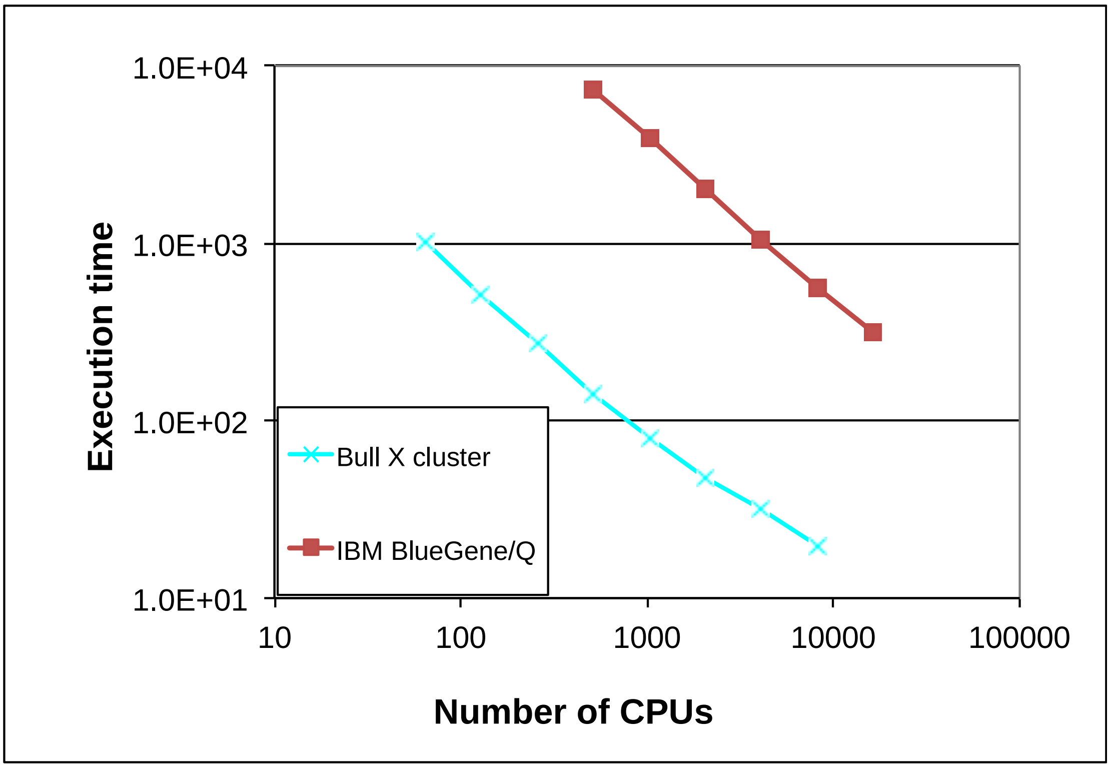

Figure 21 Execution time of GROMACS for Test Case B

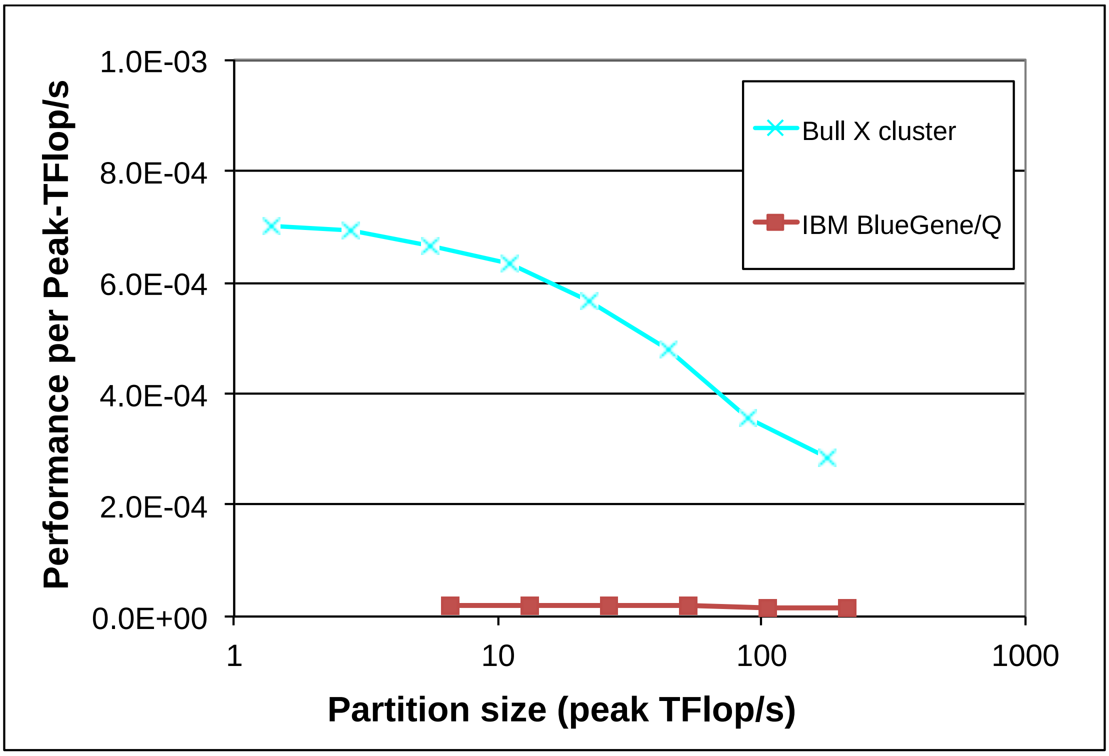

Figure 22 Performance per Peak-TFlop/s of GROMACS for Test Case B

3.8 NAMD

3.8.1 *Application and test cases*

NAMD is a widely used molecular dynamics application designed to simulate bio-molecular systems on a wide variety of compute platforms. NAMD is developed by the “Theoretical and Computational Biophysics Group” at the University of Illinois at Urbana Champaign. In the design of NAMD particular emphasis has been placed on scalability when utilising a large number of processors. The application can read a wide variety of different file formats, for example force fields, protein structure, which are commonly used in bio-molecular science.

A NAMD license can be applied for on the developer’s website free of charge. Once the license has been obtained, binaries for a number of platforms and the source can be downloaded from the website.

Deployment areas of NAMD include pharmaceutical research by academic and industrial users. NAMD is particularly suitable when the interaction between a number of proteins or between proteins and other chemical substances is of interest. Typical examples are vaccine research and transport processes through cell membrane proteins.

NAMD is written in C++ and parallelised using Charm++ parallel objects, which are implemented on top of MPI.

Web site: http://www.ks.uiuc.edu/Research/namd/

The datasets are based on the original "Satellite Tobacco Mosaic Virus (STMV)" dataset from official NAMD site.

Test Case A This is a 2x2x2 replication of the STMV dataset.

Test Case B This is a 3x3x3 replication of the STMV dataset.

3.8.2 *Sample results*

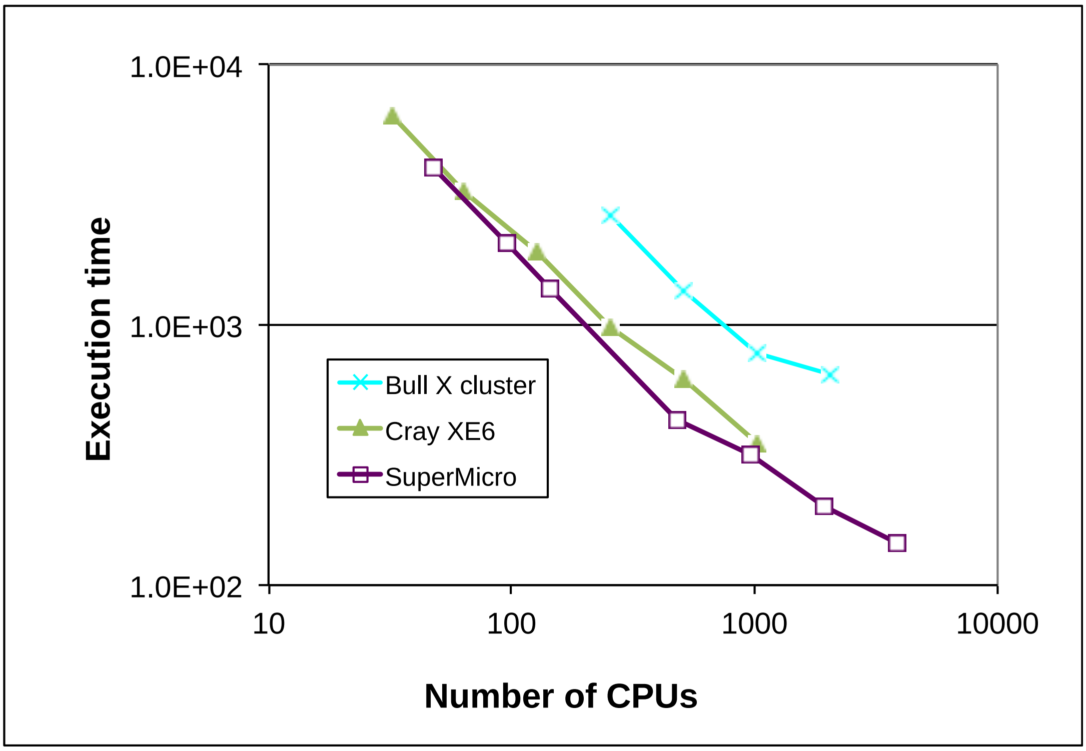

Figure 23 Execution time of NAMD for Test Case A

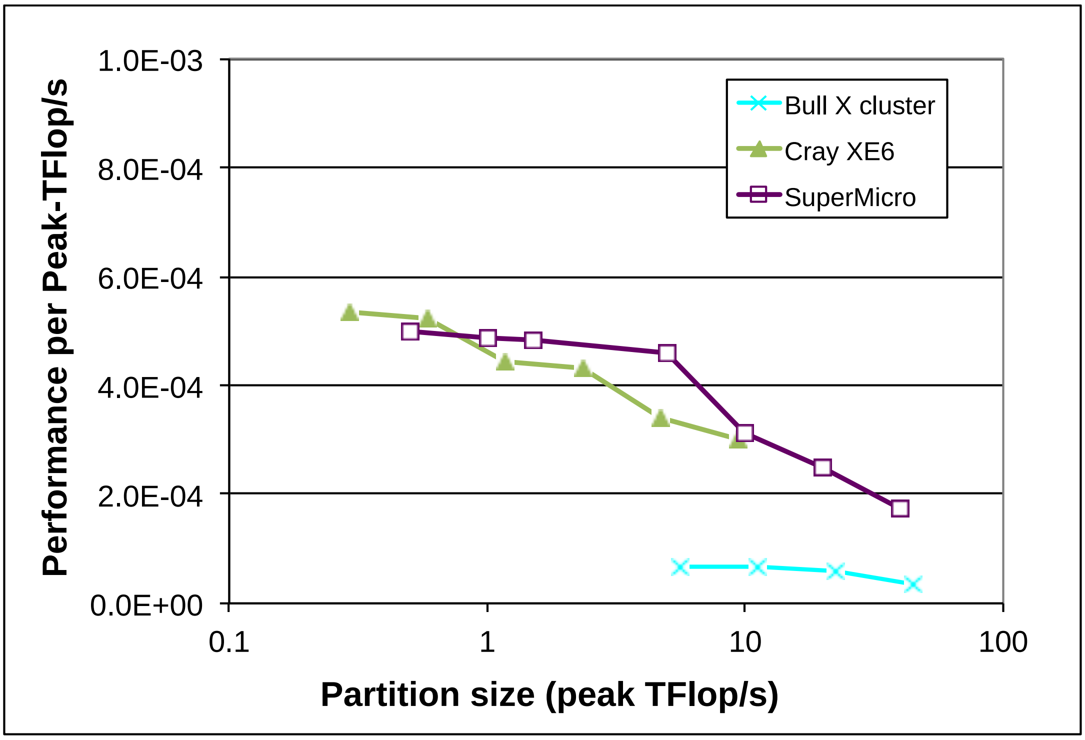

Figure 24 Performance per Peak-TFlop/s of NAMD for Test Case A

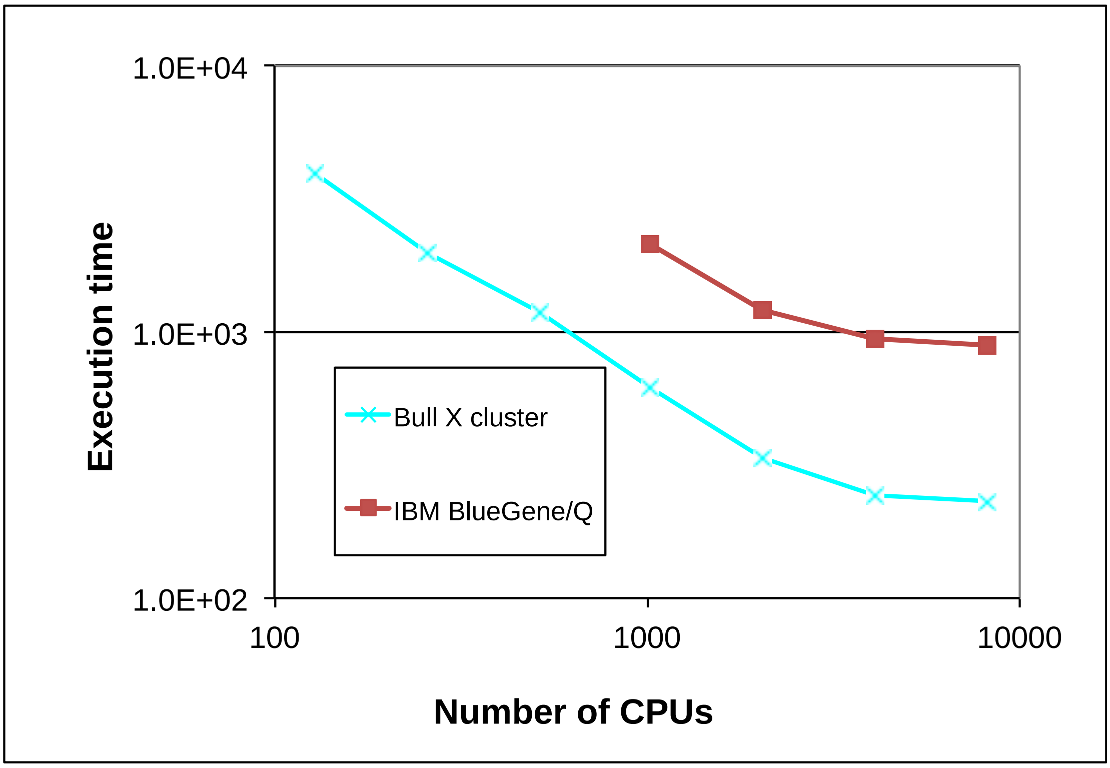

Figure 25 Execution time of NAMD for Test Case B

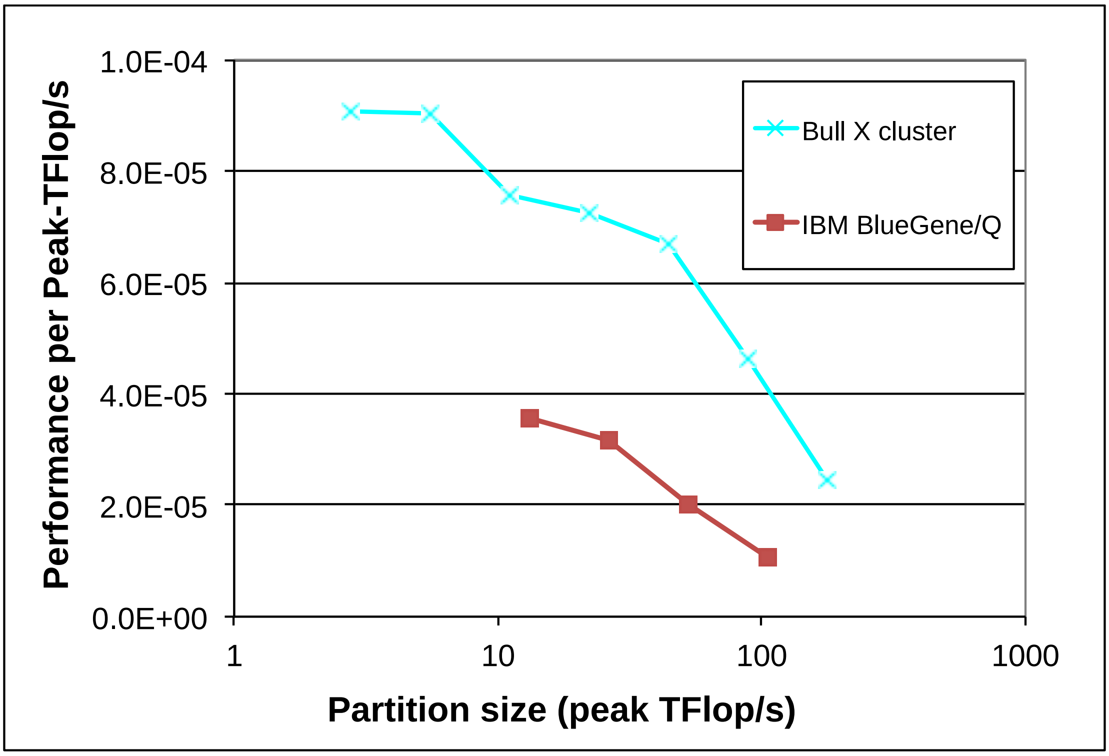

Figure 26 Performance per Peak-TFlop/s of NAMD for Test Case B

3.9 NEMO

3.9.1 *Application and test cases*

NEMO (Nucleus for European Modeling of the Ocean) is a state-of-the-art modeling framework for oceanographic research, operational oceanography seasonal forecast and climate studies. Prognostic variables are the three-dimensional velocity field, a linear or nonlinear sea surface height, the temperature and the salinity. In the horizontal direction, the model uses a curvilinear orthogonal grid and in the vertical direction, a full or partial step z-coordinate, or s-coordinate, or a mixture of the two. The distribution of variables is a three-dimensional Arakawa C-type grid. Within NEMO, the ocean is interfaced with a sea-ice model (LIM v2 and v3), passive tracer and biogeochemical models (TOP) and, via the OASIS coupler, with several atmospheric general circulation models. It also supports two-way grid embedding via the AGRIF software.

The framework includes five major components:

- the blue ocean (ocean dynamics, NEMO-OPA)
- the white ocean (sea-ice, NEMO-LIM)
- the green ocean (biogeochemistry, NEMO-TOP)
- the adaptive mesh refinement software (AGRIF)
- the assimilation component (NEMO_TAM)

NEMO is used by a large community: 240 projects in 27 countries (14 in Europe, 13 elsewhere) and 350 registered users (numbers for year 2008). The code is available under the CeCILL license (public license). The latest stable version is v3_4_1. NEMO is written in Fortran90 and parallelised with MPI.

Web site: http://www.nemo-ocean.eu/

Test Case A This is a 12° global configuration (ORCA12: ORCA tri-polar grid, 1/12°at equator) with 75 vertical levels. Its horizontal resolution is 4322 x 3059 grid points, whose spacing ranges from 10 km at equator down to 3 km at high latitudes. The domain is partitioned along i- and j- (horizontal) direction for parallel simulations.

3.9.2 *Sample results*

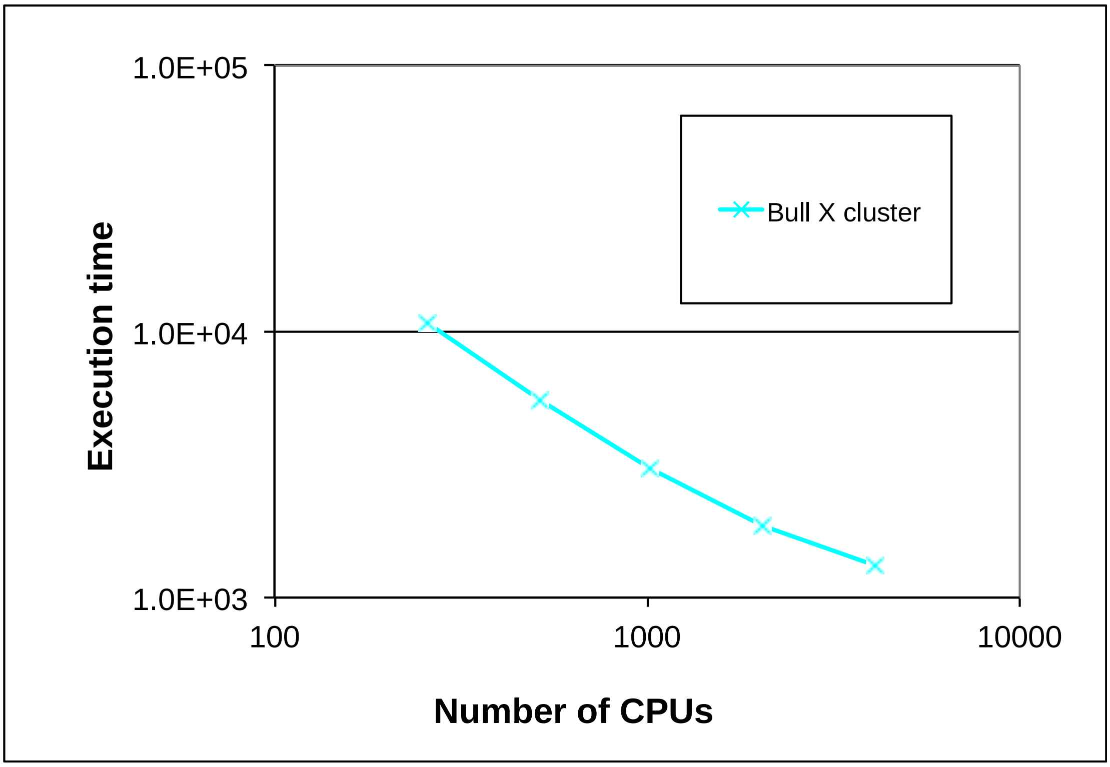

Figure 27 Execution time of NEMO for Test Case A

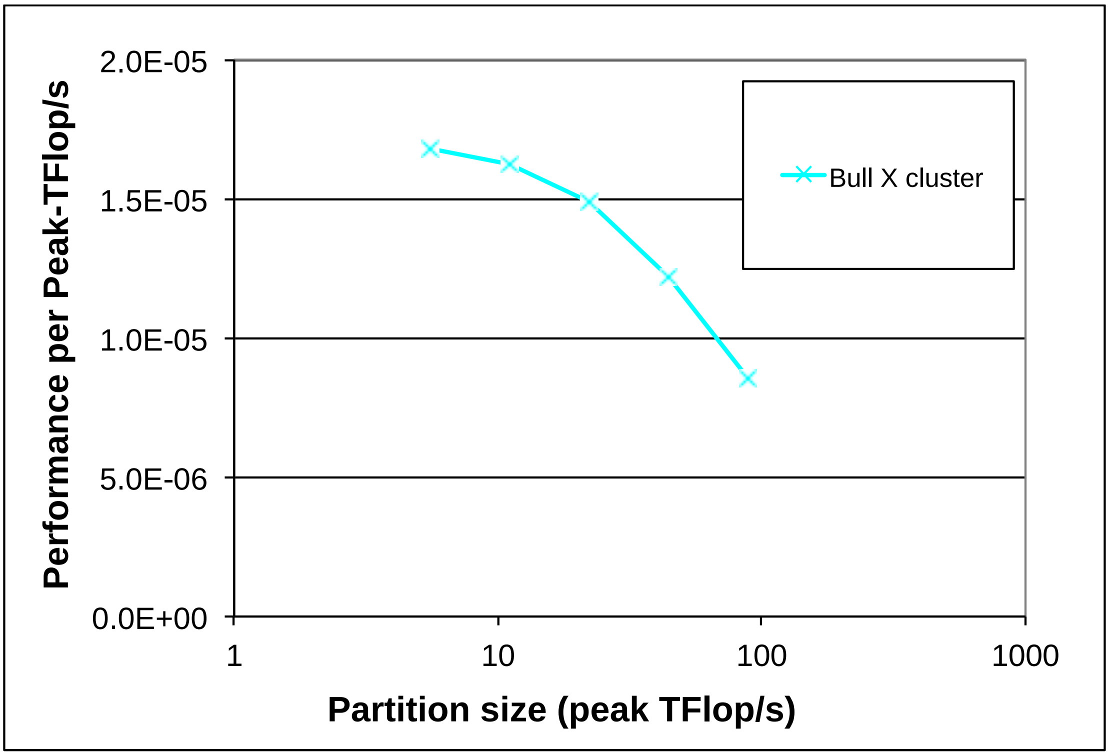

Figure 28 Performance per Peak-TFlop/s of NEMO for Test Case A

3.10 QCD

3.10.1 *Application and test cases*

The QCD benchmark is, unlike the other benchmarks in the PRACE application benchmark suite, not a full application but a set of 5 kernels which are representative of some of the most compute-intensive parts of QCD calculations.

Each of the 5 kernels has one test case:

Kernel A is derived from BQCD (Berlin Quantum ChromoDynamics program), a hybrid Monte-Carlo code that simulates Quantum Chromodynamics with dynamical standard Wilson fermions. The computations take place on a four-dimensional regular grid with periodic boundary conditions. The kernel is a standard conjugate gradient solver with even/odd pre-conditioning. Lattice size is 32^2 x 64^2.

Kernel B is derived from SU3_AHiggs, a lattice quantum chromodynamics (QCD) code intended for computing the conditions of the Early Universe. Instead of "full QCD", the code applies an effective field theory, which is valid at high temperatures. In the effective theory, the lattice is 3D. Lattice size is 256^3.

Kernel C Lattice size is 8^4. Note that Kernel C can only be run in a weak scaling mode, where each CPU stores the same local lattice size, regardless of the number of CPUs. Ideal scaling for this kernel therefore corresponds to constant execution time, and performance is simply the reciprocal of the execution time.

Kernel D consists of the core matrix-vector multiplication routine for standard Wilson fermions. The lattice size is 64^4 .

Kernel E consists of a full conjugate gradient solution using Wilson fermions. Lattice size is 64^3 x 3.

3.10.2 *Sample results*

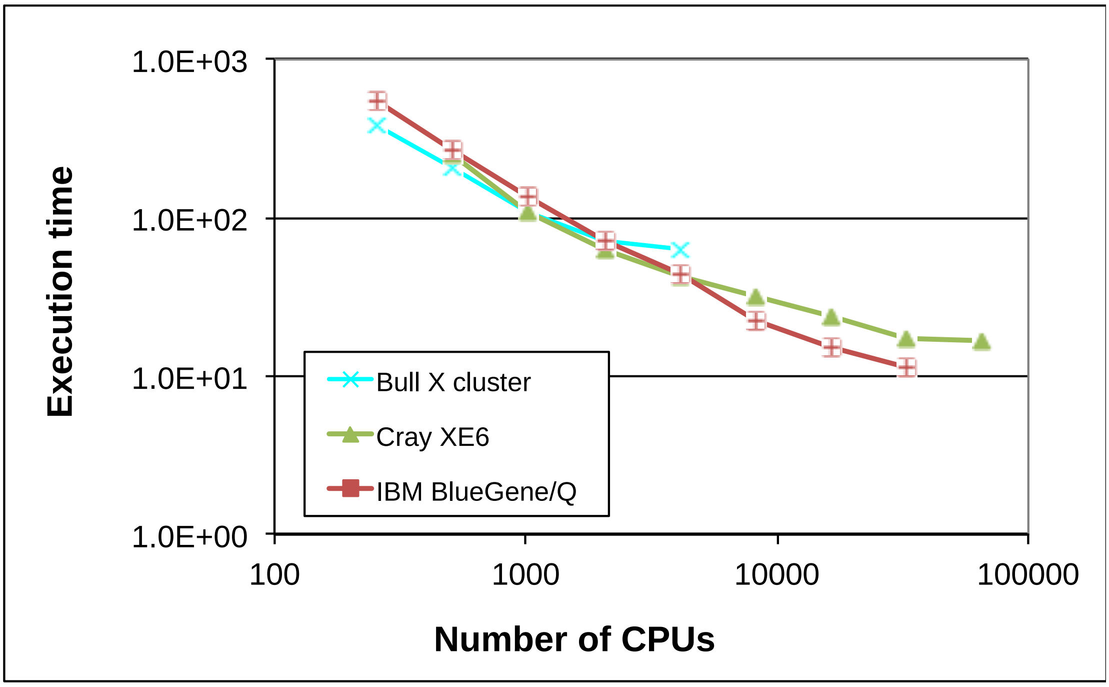

Figure 29 Execution time of QCD for Kernel A

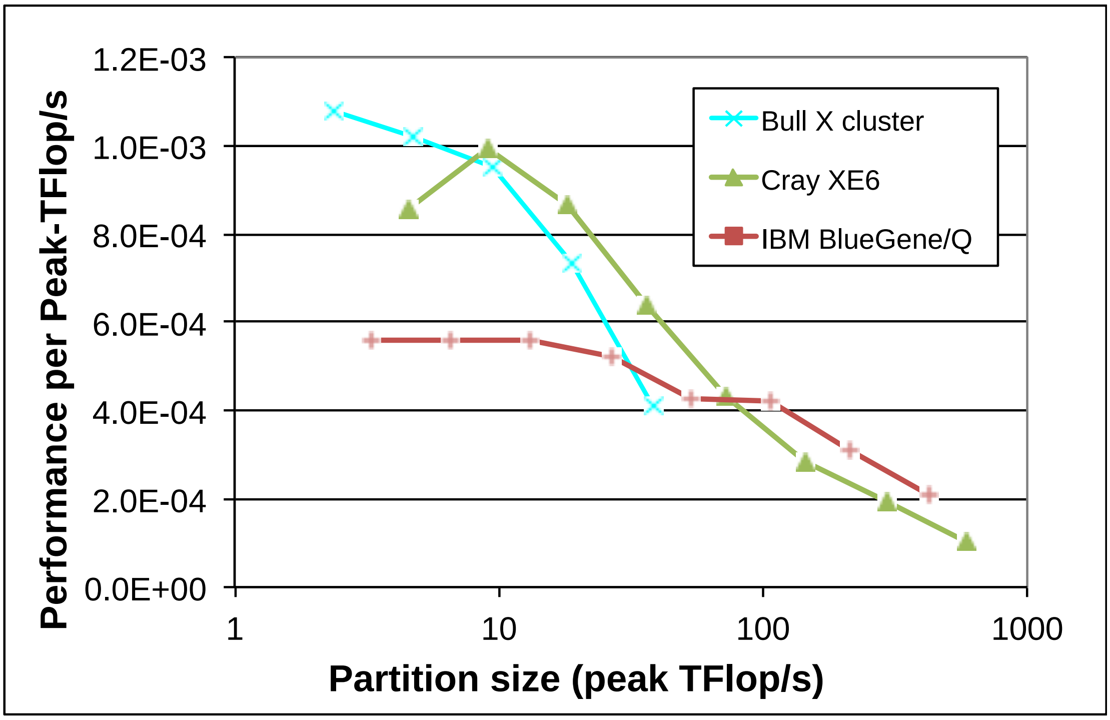

Figure 30 Performance per Peak-TFlop/s of QCD for Kernel A

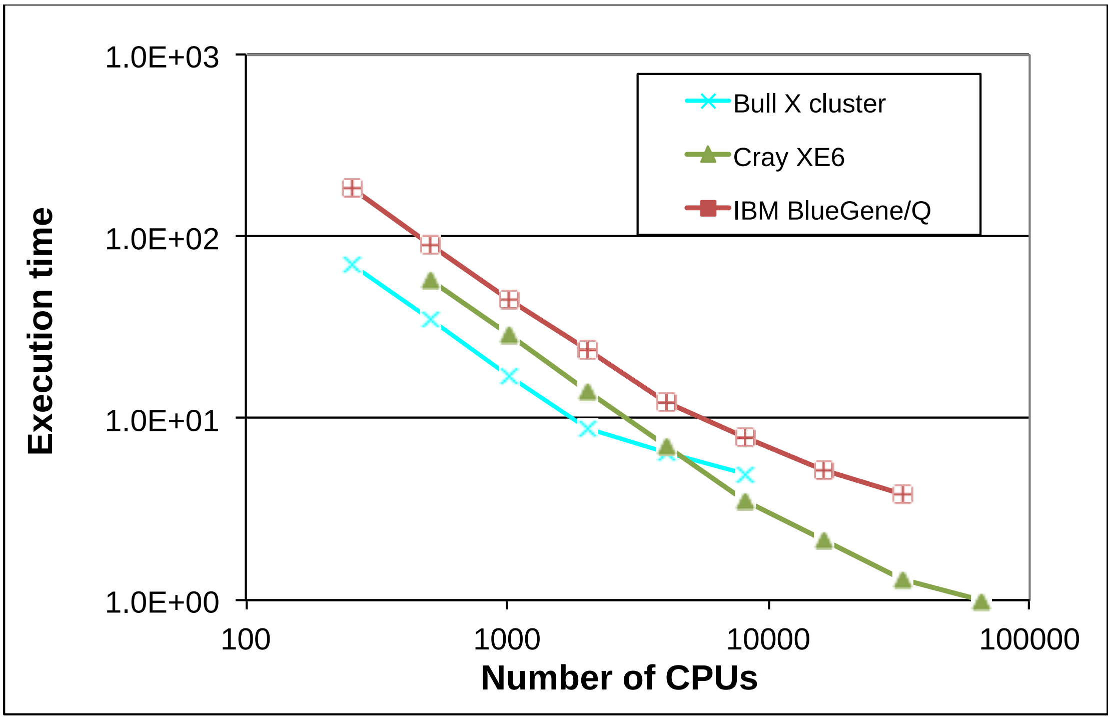

Figure 31 Execution time of QCD for Kernel B

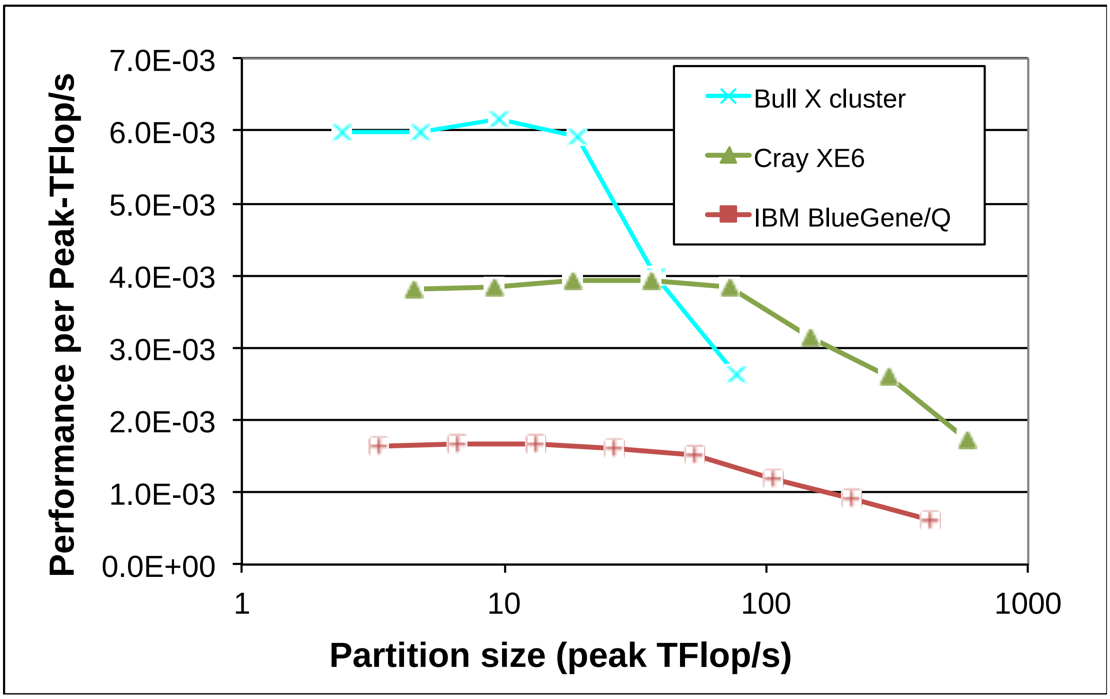

Figure 32 Performance per Peak-TFlop/s of QCD for Kernel B

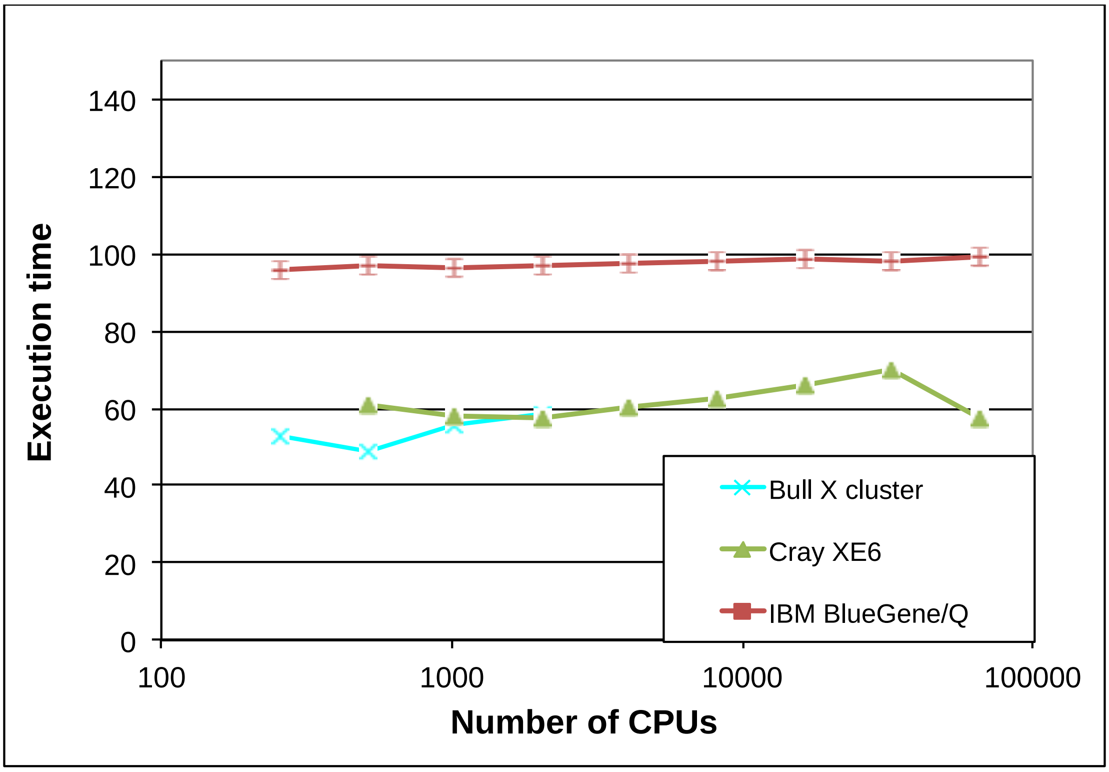

Figure 33 Execution time of QCD for Kernel C

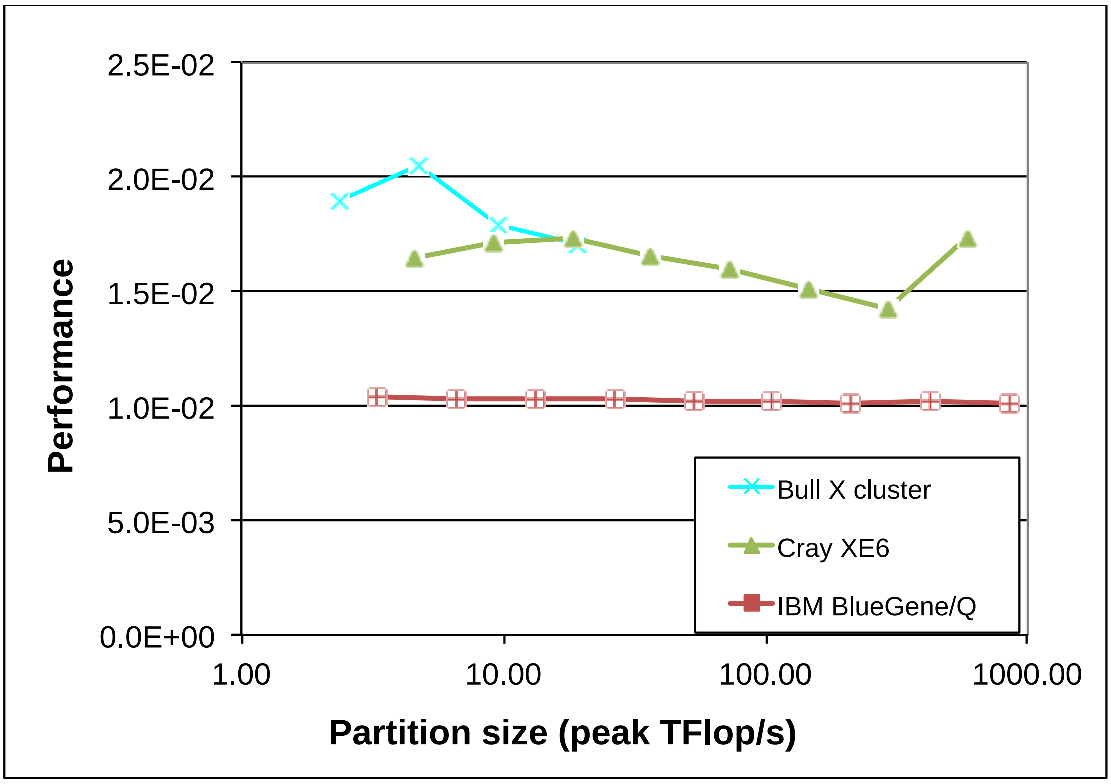

Figure 34 Performance of QCD for Kernel C

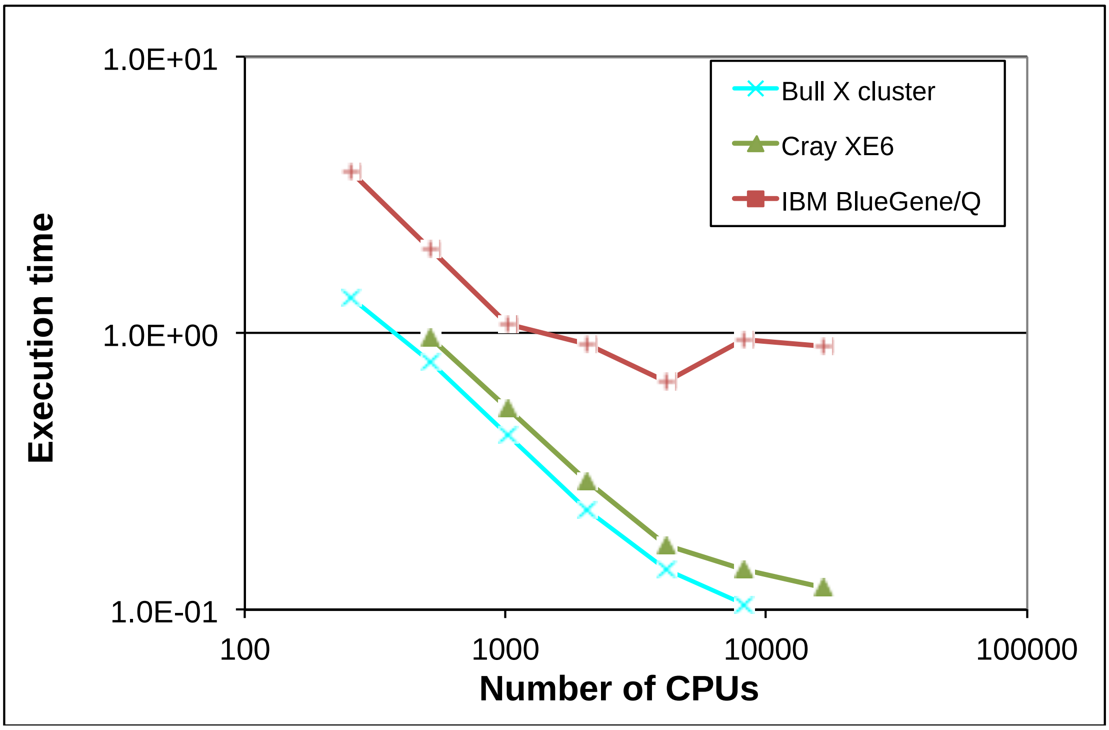

Figure 35 Execution time of QCD for Kernel D

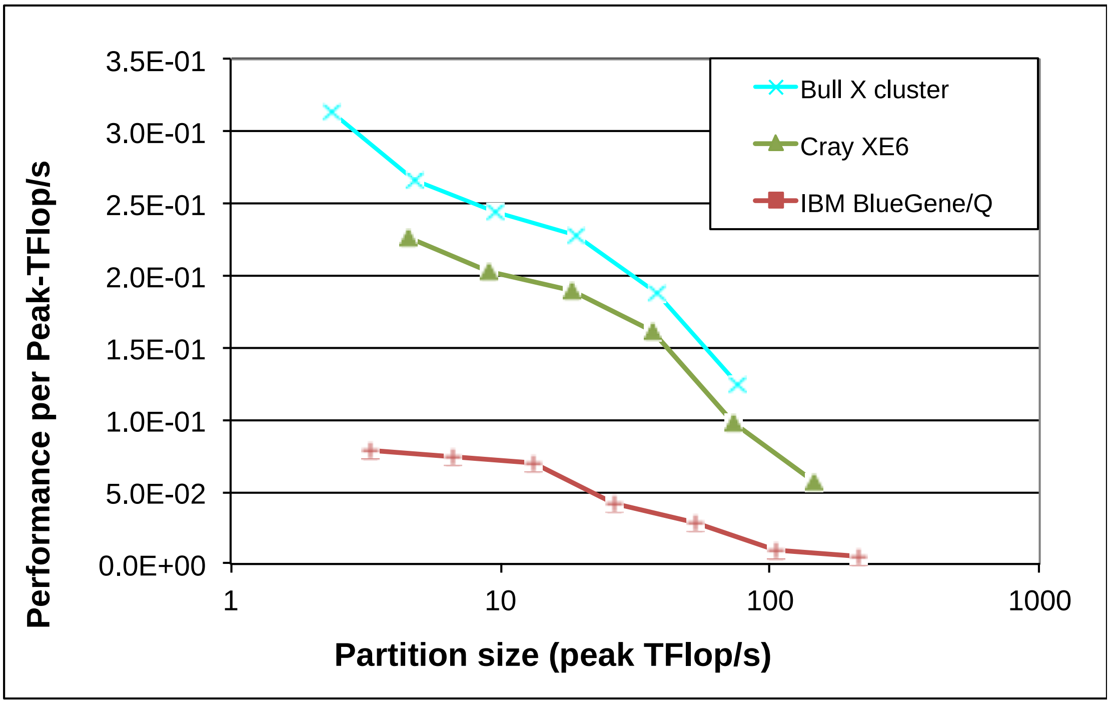

Figure 36 Performance per Peak-TFlop/s of QCD for Kernel D

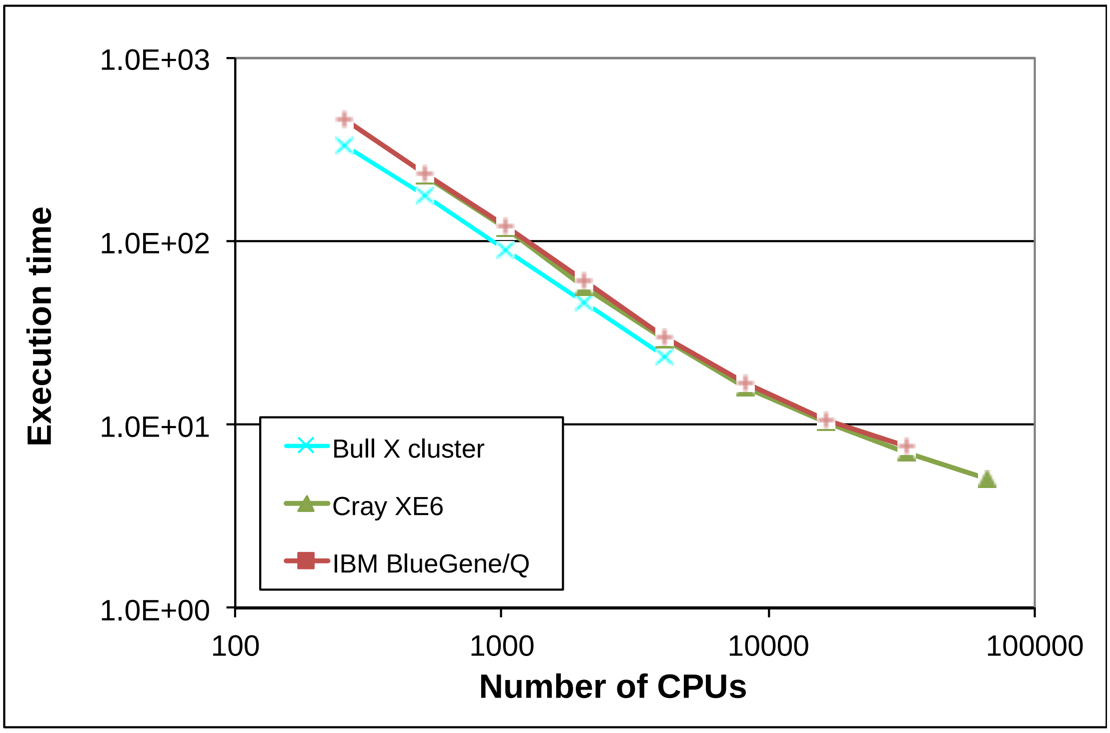

Figure 37 Execution time of QCD for Kernel E

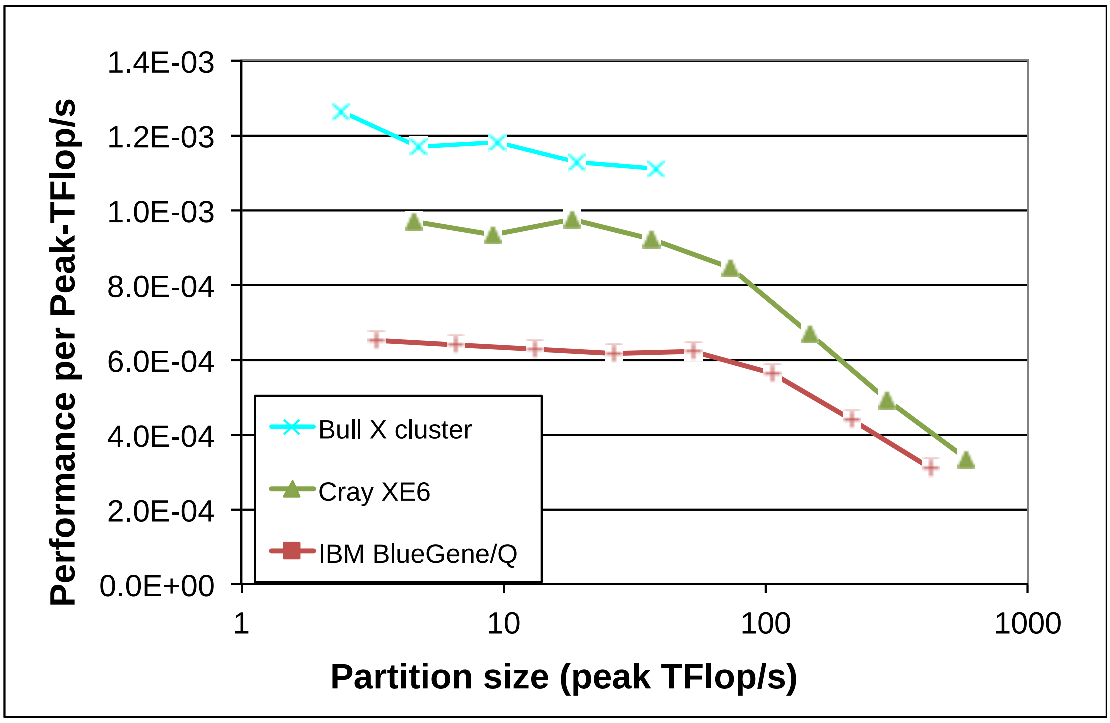

Figure 38 Performance per Peak-TFlop/s of QCD for Kernel E

3.11 Quantum Espresso

3.11.1 *Application and test cases*

QUANTUM ESPRESSO is an integrated suite of computer codes for electronic-structure calculations and materials modeling, based on density-functional theory, plane waves, and pseudopotentials (norm-conserving, ultrasoft, and projector-augmented wave). QUANTUM ESPRESSO stands for opEn Source Package for Research in Electronic Structure, Simulation, and Optimization. It is freely available to researchers around the world under the terms of the GNU General Public License. QUANTUM ESPRESSO builds upon newly restructured electronic-structure codes that have been developed and tested by some of the original authors of novel electronic-structure algorithms and applied in the last twenty years by some of the leading materials modeling groups worldwide. Innovation and efficiency are still its main focus, with special attention paid to massively parallel architectures, and a great effort being devoted to user friendliness. QUANTUM ESPRESSO is evolving towards a distribution of independent and inter-operable codes in the spirit of an open-source project, where researchers active in the field of electronic-structure calculations are encouraged to participate in the project by contributing their own codes or by implementing their own ideas into existing codes.

QUANTUM ESPRESSO is written mostly in Fortran90, and parallelised using MPI and OpenMP.

Web site: http://www.quantum-espresso.org/

Test Case A In this test case a gold surface containing 112 atoms is subject to an SCF calculation of 21 iterations (which for this example results in convergence at all core counts).

Test Case B This test case is based on two iterations of a SCF calculation on a functionalised carbon nanotube with a total of 1532 atoms.

3.11.2 *Sample results*

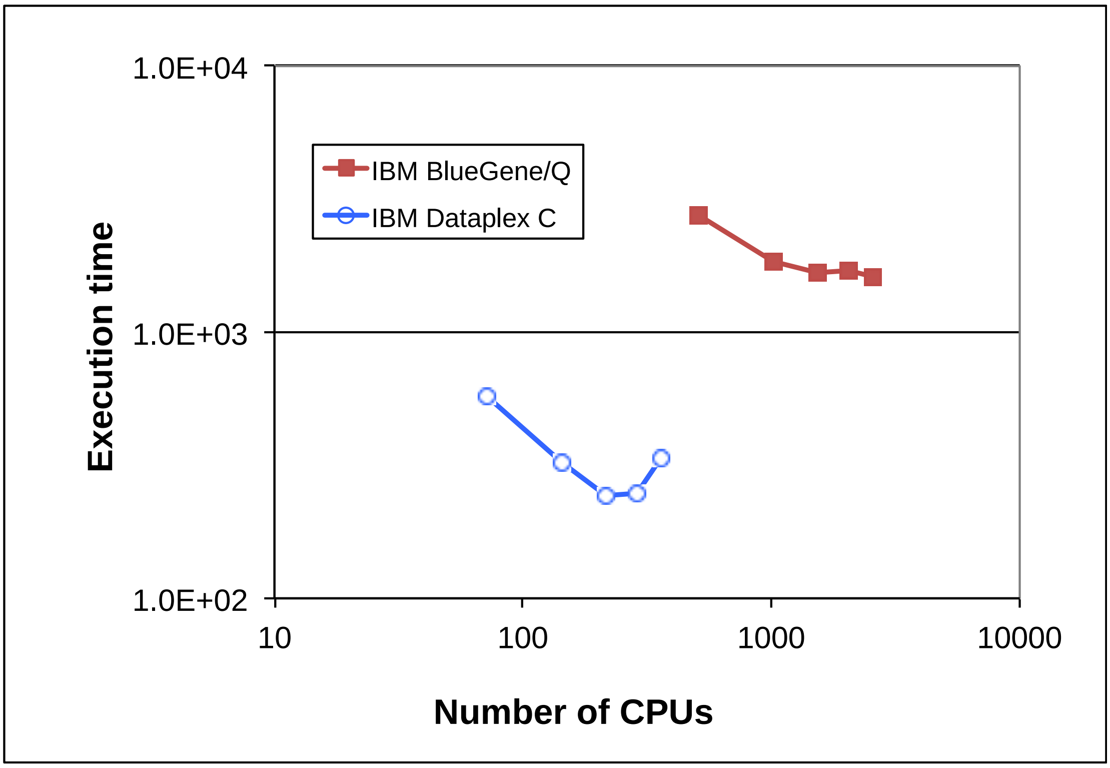

Figure 39 Execution time of Quantum Espresso for Test Case A

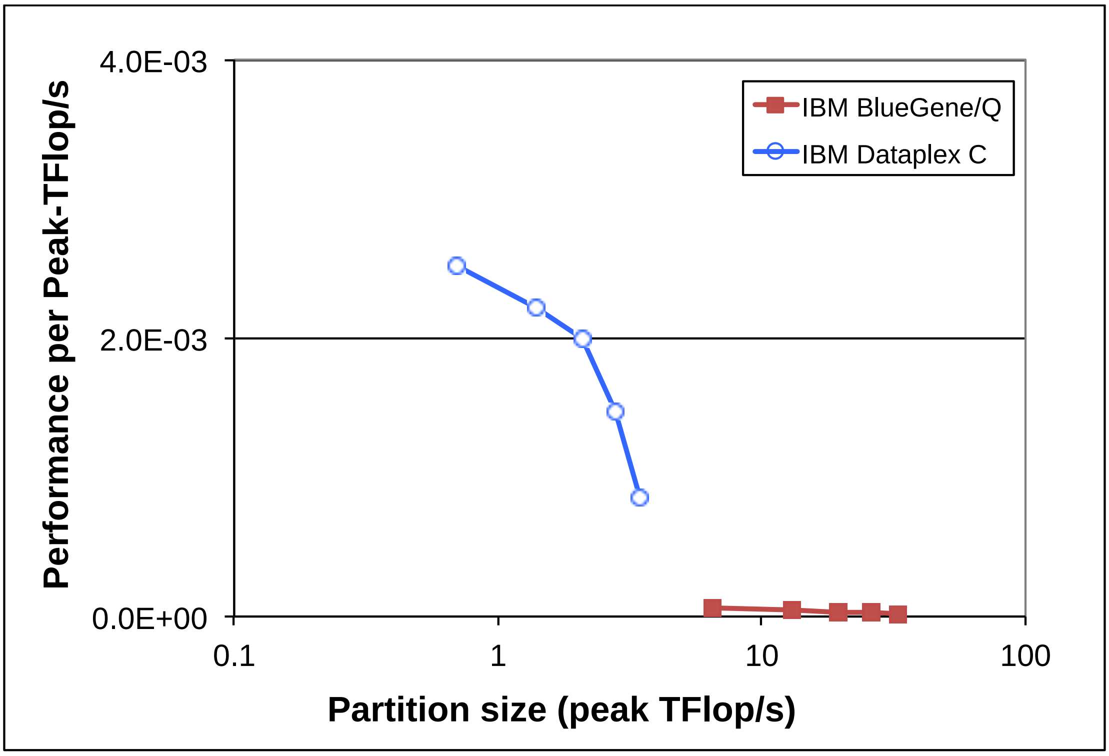

Figure 40 Performance per Peak-TFlop/s of Quantum Espresso for Test Case A

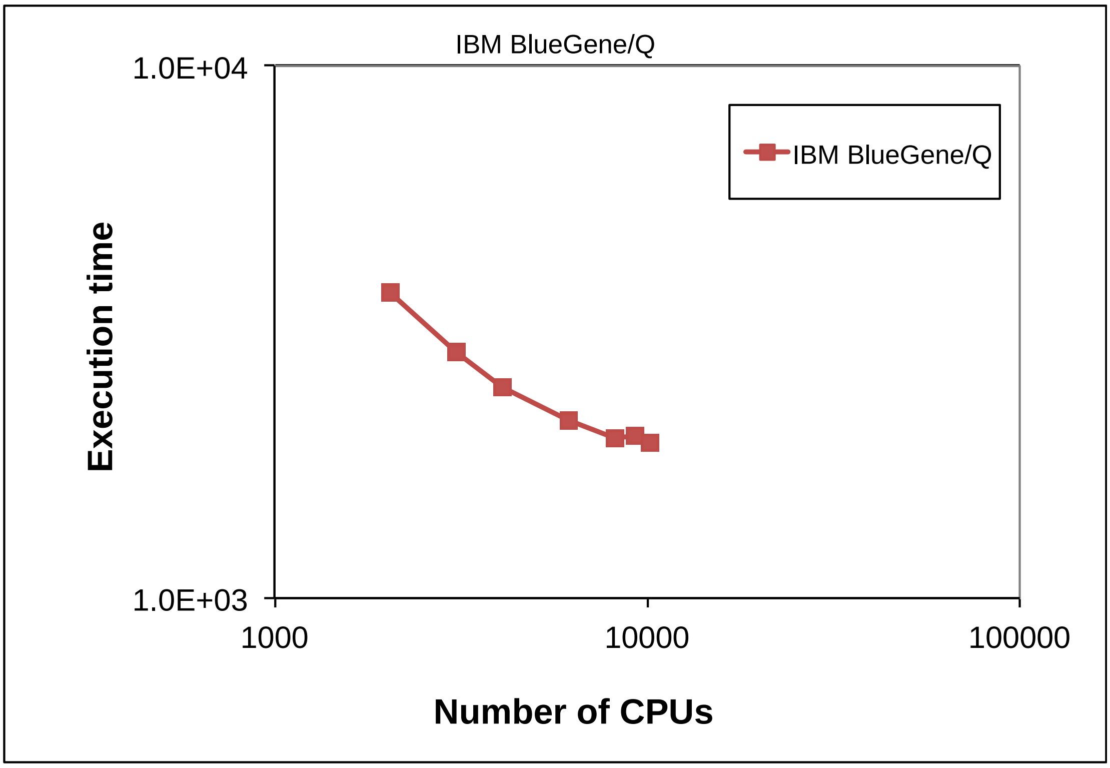

Figure 41 Execution time of Quantum Espresso for Test Case B

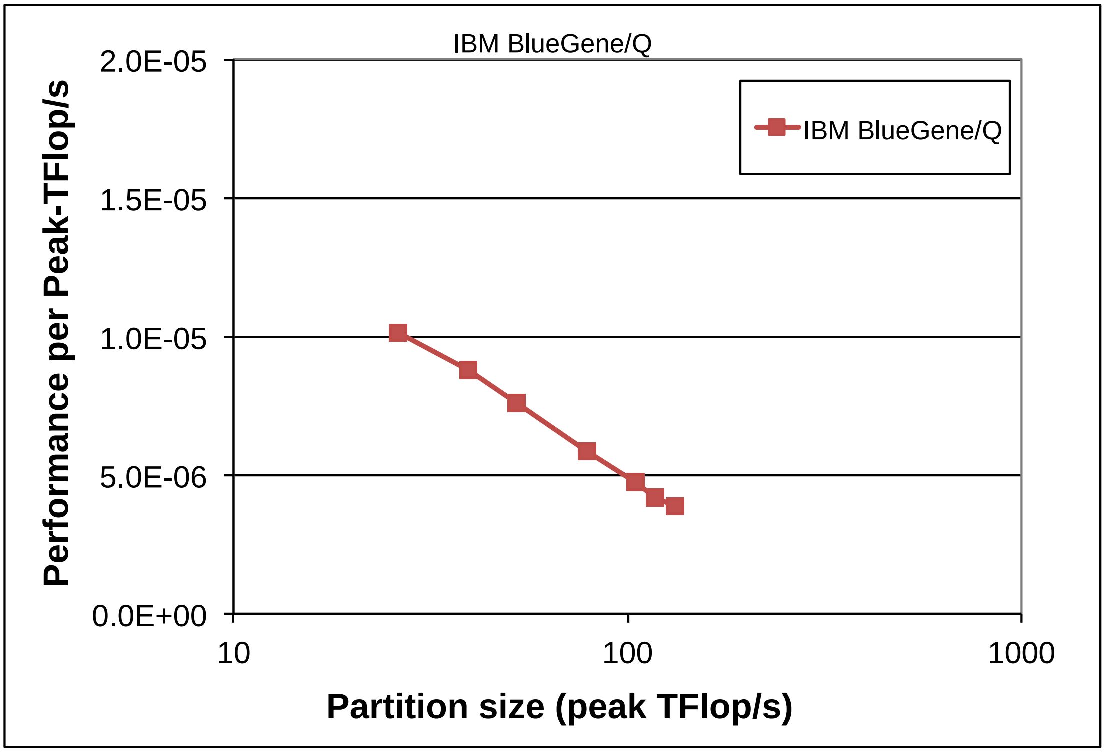

Figure 42 Performance per Peak-TFlop/s of Quantum Espresso for Test Case B

3.12 SPECFEM3D

3.12.1 *Application and test cases*

The software package SPECFEM3D simulates three-dimensional global and regional seismic wave propagation based upon the spectral-element method (SEM). All SPECFEM3D_GLOBE software is written in Fortran90 with full portability in mind, and conforms strictly to the Fortran95 standard. It uses no obsolete or obsolescent features of Fortran77. The package uses parallel programming based upon the Message Passing Interface (MPI).

The SEM was originally developed in computational fluid dynamics and has been successfully adapted to address problems in seismic wave propagation. It is a continuous Galerkin technique, which can easily be made discontinuous; it is then close to a particular case of the discontinuous Galerkin technique, with optimized efficiency because of its tensorized basis functions. In particular, it can accurately handle very distorted mesh elements. It has very good accuracy and convergence properties. The spectral element approach admits spectral rates of convergence and allows exploiting hp-convergence schemes. It is also very well suited to parallel implementation on very large supercomputers as well as on clusters of GPU accelerating graphics cards. Tensor products inside each element can be optimized to reach very high efficiency, and mesh point and element numbering can be optimized to reduce processor cache misses and improve cache reuse. The SEM can also handle triangular (in 2D) or tetrahedral (3D) elements as well as mixed meshes, although with increased cost and reduced accuracy in these elements, as in the discontinuous Galerkin method.

In many geological models in the context of seismic wave propagation studies (except for instance for fault dynamic rupture studies, in which very high frequencies of supershear rupture need to be modeled near the fault, a continuous formulation is sufficient because material property contrasts are not drastic and thus conforming mesh doubling bricks can efficiently handle mesh size variations. This is particularly true at the scale of the full Earth.

Effects due to lateral variations in compressional-wave speed, shear-wave speed, density, a 3D crustal model, ellipticity, topography and bathyletry, the oceans, rotation, and self-gravitation are included. The package can accommodate full 21-parameter anisotropy as well as lateral variations in attenuation. Adjoint capabilities and finite-frequency kernel simulations are also included.

Web site: http://www.geodynamics.org/cig/software/specfem3d-globe

The test cases simulate the earthquake of June 1994 in Northern Bolivia at a global scale with the spherically symmetric isotropic IASP91 model. The solver calculates seismograms for 129 stations, and all simulations are run for a record length of 0.2 minutes.

The different test cases correspond to different meshes of the earth. The size of the mesh is determined by a combination of following variables: NCHUNKS, which is the number of chunks in the cubed sphere (6 for global simulations), NPROC_XI, which is the number of processors or slices along one chunk of the cubed sphere and NEX_XI, which is the number of spectral elements along one side of a chunk in the cubed sphere. These three variables give us the number of degrees of freedom of the mesh and determine the amount of memory needed per core.

Test Case A uses NCHUNKS=6, NPROC_XI=12 and NEX_XI=768.

Test Case B uses NCHUNKS=6, NPROC_XI=44 and NEX_XI=1760.

3.12.2 *Sample results*

So far it has only been possible to run on one fixed core count for each test case, so scaling curves are not available. On the Bull X Cluster, Test Case A runs in 2.20 seconds on 864 cores, and Test Case B in 2.76 seconds on 11616 cores.

4 Conclusions and Future Work

We have presented the Unified European Application Benchmark Suite (UEABS), a set of 12 application codes, selected for open licensing, public availability of suitable datasets, support from developers, portability, scalability and relevance to science communities. We have described the associated test case datasets, and presented results from several PRACE systems, which show that these test cases have been run successfully and at appropriate scale.

The suite, including test cases, will be made available on the PRACE project public website. For application codes which are publically downloadable, the website will not redistribute source code, but will contain links to the primary download sites.

Task 7.3 in PRACE-3IP has the responsibility for maintaining the UEABS, and will assume responsibility for this now that the work under Task 7.4 in PRACE-2IP is complete. The contents of the suite will be reviewed to ensure that it remains relevant and continues to meet the criteria used to select the applications. The applications will be run on a more complete set of PRACE Tier-0 and Tier-1 systems, in order to help European researchers chose the systems that are appropriate for their computational requirements.