

Investigation of message-passing communication patterns (case study for the opatm-bfm application)

Vladimir Svjatnjy, Alexey Cheptsov
Department of Computer Science, National Technical University of Donetsk
lex@cs.dgtu.donetsk.ua

Rainer Keller
High-Performance Computing Center Stuttgart, University of Stuttgart
keller@hirs.de

Stefano Salon
Nazionale di Oceanografia e di Geofisica Sperimentale
ssalon@inogs.it

Abstract

Svjatnjy V., Cheptsov A., Keller R., Salon S. Investigation of message-passing communication patterns (case study for the opatm-bfm application). The communities coming from different fields of science and technology can make a strong benefit of usage Grid high-performance resources for their applications. This paper presents the first results of research activities devoted to the adaptation of the parallel OPATM-BFM application developed in OGS for an efficient usage in modern Grid based e-Infrastructures. OPATM-BFM is an off-line three-dimensional coupled eco-hydrodynamic simulation model used for biogeochemical and ecosystem-level predictions. For the application performance on standard Grid architectures providing generic clusters of workstations such results are important. We propose a message-passing analysis technique for communication-intensive parallel applications based on a preliminary application run analysis. This technique was successfully used for the OPATM-BFM application and allowed us to identify several optimization proposals for the current realization of the communication pattern. As the suggested improvements are quite generic they can be potentially useful for other parallel scientific applications.

Introduction

Environmental risks of natural or anthropogenic origin may be prevented or managed by the use of operational short-term forecasts. The Global Monitoring for Environment and Security [1] is a partnership of the European Commission and the European Space Agency, and aims to develop short-term forecasts of marine ecosystems to be used in the frame of environmental

monitoring. To tackle this challenging goal, the MERSEA European Integrated Project [2] was launched in order to develop a sustainable pan-European pre-operational forecasting system. The objective of this system was to provide products, on a regional basis, to a variety of intermediate users. The OPATM-BFM coupled model [3] that is developed in OGS is one of these products, and today represents a state-of-the-art numerical tool able to produce biogeochemical and ecosystem-level short-term predictions for different basins. The next section of this paper describes the basics of the model that can be operationally applied to a basin such as the Mediterranean Sea.

The parallel realization of the model developed in OGS allows its usage for biogeochemical and ecosystem-level predictions for a wide range of environmental and ecological systems. The efficient usage of high-performance resources (like an IBM SP5 machine of CINECA [4] currently used for the main production cycle) is a major point in reaching a high application productivity. However, while the model is expected to deliver additional products and information for different time scales as well as for climatic scenario analyses (multi-decadal period of integration), the facilities of dedicated high-performance architectures offer limited access to computation and storage resources for the application. In fact, that is the most considerable limitation of the application scalability. On the contrary, the newly deployed scientific e-Infrastructures (e.g. DORII [5]) are supposed to provide scientific applications deployed in their framework with a full-range access to the distributed computational, storage and other grid services [6,7].

The experience of porting other scientific applications from different fields of science to the grid [8] has encouraged us to examine the suitability of the grid resources for improving the OPATM-BFM application performance. For this purpose the e-Infrastructure provides the application with all necessary monitoring and development tools as well. The anticipated effect from porting to the grid is increasing of the application performance and scalability. This will allow an efficient usage of the OPATM-BFM application for more complicated use cases.

This paper describes the first research results devoted to issues of the OPATM-BFM application porting to the grid. The application testing is being performed on a cluster of workstations which is representative for a grid component today. We show that application performance can dramatically decrease because of shortcomings pertained to the current realization.

A deep understanding of the message-passing communication pattern realization was therefore the obligatory step towards porting and efficient usage of the application on grid resources. In the sections 2 and 3 of the paper we describe in detail the current implementation of the communication pattern. However, the communication pattern extraction, categorization, analysis and optimization is not a trivial task. We also discuss some problems and present a technique that allows us to proceed with analysis of communication-intensive

parallel applications. Being successfully used for the OPATM-BFM application, the technique should also be valuable for the investigation of other parallel scientific applications.

Finally, this paper describes optimization proposals for the current realization of the OPATM-BFM message-passing communication pattern and evaluation results. The defined proposals could be valuable for the application runs on dedicated resources as well as grid resources. The result might also be relevant for other scientific applications that implement a similar inter-process communication (e.g. the all-to-one communication) or for newly developed parallel applications.

Overview of the OPATM-BFM coupled model

OPATM-BFM is an off-line¹ three-dimensional MPI-parallel coupled eco-hydrodynamic model, and is the core of a forecasting system embedded in a fully automatic procedure that produces maps of biogeochemical concentrations for the whole Mediterranean basin on a weekly basis.

The model solves the transport-reaction equations (1) for the generic biogeochemical concentration c_i based on the advection-diffusion processes:

$$\frac{\partial c_i}{\partial t} + v \cdot \nabla c_i = w_i \frac{\partial c_i}{\partial z} + k_h \nabla_h c_i + \frac{\partial}{\partial z} \left[k_z \frac{\partial c_i}{\partial z} \right] + R_{bio}(c_i, c_1 \dots c_N, T, I \dots) \quad (1)$$

where v is the current velocity, w_i is the sinking velocity, k_h and k_z the eddy diffusivity constants and R_{bio} is the biogeochemical reactor that depends, in general, on the other concentrations and on temperature T , short-wave radiation I and other physical variables. The complexity of the OPATM-BFM model consists in the high number of prognostic variables (c_i) to be integrated. Bacteria, oxygen, inorganic nutrients, phytoplankton, zooplankton, organic detritus are among the 51 variables produced by the model.

The objective of developing a biogeochemical model that can be operationally applied to a basin such as the Mediterranean Sea requires an interdisciplinary effort. This work was achieved with the cooperation of different laboratories involved in the forecasting system within the framework of the Italian Group of Operational Oceanography [9]. The off-line coupling between the physics and the biogeochemistry was established between the operational forecasting system for the Mediterranean Sea managed by the National Institute of Geophysics and Volcanology (INGV) and the biogeochemical model BFM [10] embedded in the OPATM transport module. Computational resources and expertise were supplied by the Italian supercomputing center CINECA, while the Institute for Atmospheric Sciences

¹ In the off-line approach the time integration of the transport-reaction biogeochemical equations is not synchronized with the Navier-Stokes solver: the circulation field is an external forcing and it has to be known before the integration of the equations.

and Climate (ISAC-CNR) provided satellite chlorophyll data, which were used for comparison of the model results. The physical forcing fields supplied by INGV (current velocity, temperature, salinity, vertical eddy diffusivity, wind speed, short-wave radiation) are up-scaled to a lower horizontal spatial resolution, from INGV $1/16^\circ$ to OPATM-BFM $1/8^\circ$, with an interpolating interface based on the cell-merging technique, which preserves the fluxes and, consequently, the divergence of the original data. The vertical resolution (72 levels) is left unchanged in order to maintain a good reproduction of the vertical processes, which are known to be of great relevance for biogeochemical processes (mixing of the water column). This approach takes advantage of the benefits of state-of-the-art dynamic prognosis granted by INGV, that includes extensive data assimilation, and keeps the off-line dynamics files and the computational burden affordable, given the currently available resources.

The products of the simulations are the concentrations of key variables (macronutrients, chlorophyll, phytoplankton and bacterial productivity) and are routinely delivered from the OGS website [11] in order to track temporal and vertical dynamics of the basin biogeochemical properties.

This system represents a conceptual evolution when compared with previous basin-wide on-line coupled models developed for the Mediterranean Sea. Daily mean forcing fields used in the off-line coupling filter out the numerical noise introduced by integrations, consistently with the assumptions made in many experiments performed to estimate biological parameters. At the same time, every improvement in the physical model is immediately gained by the biogeochemical compartment.

OPATM-BFM has been implemented mainly in open sea regions and therefore can be safely used, when properly validated, for large-scale assessments such as:

- estimation of the carrying capacity of the Mediterranean basin (a valuable information for ecosystem-based approaches to fisheries management);
- provision of habitat suitability indicators for risk assessment of a non-Mediterranean species invasion, such as algae;
- regional ecosystem responses to climate change;
- scenario analyses;
- design of observational research cruises and activities.

The parallel realization of the model developed at OGS is based on the message-passing communication pattern implemented by means of MPI [12].

The OPATM-BFM model was originally designed to provide initial and boundary conditions for coastal biogeochemical models in the Mediterranean Ocean Observing Network [13]. However, in light of the promising results achieved so far during the pre-operational phase (10-20 days of integration) [14], we expect that additional products and information can also be delivered at

different time scales, as well as for climatic scenario analyses (multi-decadal period of integration).

Measurements and analysis results

This section gives an overview of characteristics of the application run profile. The most communication- and computation-intensive phases of the application execution are identified for both test (3 steps of numerical solution) and standard (816 and more steps) use cases. The measurements and analysis results are presented as well.

Analysis of the application run profile

Due to the long duration of the execution (several hours) for a standard OPATM-BFM use case (namely 816 steps of the main simulation - corresponds to 17 days of simulated ecosystem behavior) the size of recorded trace data increases accordingly. Hence the trace data being collected for the execution of the standard use case could not be processed by the available software for communication analysis and performance evaluation.

As for standard production run sizes with many communication and sub-function calls per iteration, the trace file may get very large (up to tens of gigabytes). This is typical for analysis of parallel scientific applications. However, the time integration of OPATM-BFM is performed in the main loop where each step has identical MPI structures. This means that the typical pattern of the MPI calls is iteration-independent. Hence, the iteratively repeated regions can be profiled for a limited number of iterations that are representative of the generic pattern communication of a longer run. The initialization and finalization of the simulation are profiled as usual. This can be done by means of event filtering in the defined regions of the execution or launching the application for special use cases with a limited number of iterations. The second approach is preferable because it allows to reduce the time needed for launching the application in the test mode.

In order to proceed with the communication analysis efficiently, the phases of the application execution are to be identified. The localization of the most computation- and communication-intensive phases without the help of profiling tools is a non-trivial and quite complicated process that requires deep understanding of the application source code as well as the model basics [19]. However, a so called application call graph from a profiling tool [20] is sufficient for a basic understanding of dependencies between the regions of the application execution as well as time characteristics of the communication events in those regions. A fragment of the application call graph for most important phases of the application execution is presented in Figure 1.

Such a run profile analysis is an excellent starting point for further investigation of the message-passing communication in the parallel application. This can be done by performing profiling of MPI operations which account for the most significant application execution regions.

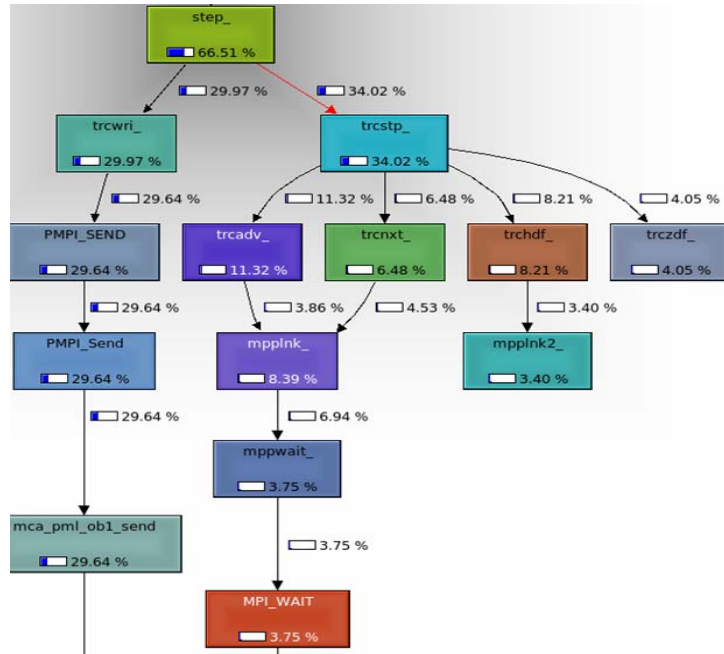


Figure 1 – A fragment of the application call graph (obtained with KCacheGrind tool).

For application run profiling we used tools from the instrumentation framework (e.g. the Valgrind tool suite [21]). Assuming that the communication mechanism implemented in the main simulation step routine does not depend on iterations, we were able to limit the number of iterations that are profiled in the main simulation routine. For this purpose a special test use case which required only 3 steps of the main simulation was specified. That corresponds to 1.5 hours real time of the ecosystem evolution.

Main results of profiling for the test use case are collected in Table 1. The timing characteristics are followed by results for the application scalability acquired by launching the application on a larger number of CPUs (Table 2). It is important to emphasize that the time distribution of the different phases compared to the total run time differs hugely for the test use case and for a real long-term simulation that requires a larger number of steps of the numerical solution [14]. This is due to the fact that only the iterative part changes but not the initialization and finalization parts. The application scalability for running on different number of CPUs is also changing accordingly to the size of the use case. Nevertheless, the internal characteristics of the iterative phase are iteration-independent and valid not only for the test use case but for all use cases.

While the most considerable part of the application execution is occupied by operations of data input from the disk storage for the test use case (80% of the application execution time), the iteratively launched main simulation routine that implements a numerical solver will become a dominating part of the execution (up to 90% of the application execution and more) for a real production cycle use case that requires a big number of steps of numerical solution. For example, for 2 weeks prediction 816 steps of numerical solution are required (see Table 1). Being launched only each 48th step of the numerical solution and at the end of the application execution, disk storage and data output operations are also an important point of the application performance optimization for both short-term and long-term forecasts.

Table 1 – Time distribution among the main phases of the execution (for the testing use case)

Phases of execution	Operations performed	Num. of iterations for the test/real use case	Computation		MPI-calls		Total	
			Time [s]	Part in phase, %	Durat. min.	Part in phase, %	Durat. min.	Part in the execution, %
1. Initialization, Input	Loading of input data	1/ 1	15:39	99	0:6	< 1	15:45	80
2. Main simulation routine	Internal MPI calls, halo cells exchange	3/ 816	0:3	38	0:5	62	0:8	2
3. Data storage, Output	Storing of output and restarting data on disc, internal MPI communication	1/ 17	0:7	3	3:34	97	3:41	18
Totally for the execution			15:49	80	3:46	20	19:35	100

Table 2 – Scalability characteristics of the application (for the testing use case)

Phases of execution	32 CPU			64 CPU			Scalability coefficient		
	Comput. time, min.	MPI calls' time, min.	Total time	Comput. time, min.	MPI calls' time, min.	Total time	Comput.	MPI calls	Total
1. Initialization, Input	15:39	0:6	15:45	37:18	0:6	37:24	0,4	1	0,4
2. Main simulation routine	0:3	0:5	0:8	0:1,7	0:5,2	0:6,9	1,7	1	1,2
3. Data storage, Output	0:7	3:34	3:41	0:2,7	2:50	2:52,7	2,6	1,25	1,4
Totally for the execution	15:49	3:46	19:35	37:22,4	3:1,2	40:23,6	0,5	1,1	0,5

The message-passing communication profile

The application is parallelized using the domain decomposition method. This required a message-passing mechanism for the data exchange between domains. This mechanism is implemented in the main step simulation routine (exchange of boundary elements of neighboring domains is performed; required on each step of simulation) and in the data storage and output routine (*trcwri*)

In the phase where the data storage and output operations are used to save restart data, the blocking point-to-point communication is used for data gathering in the root process. It is performed by means of 612 MPI_Send calls that are executed by each of the non-root processes and a corresponding number of MPI_Recv calls in the root process (all-to-one communication pattern). Cumulatively for the analyzed region 18972 data messages are transmitted by means of MPI. The size of the transmitted messages varies from 4kB (12648 messages) up to 1.125MB (204 messages).

However, the most MPI calls occur in the main step simulation routine. The frequency of message transmission in this phase of the application execution is very high (for the investigated use case totally 252960 messages are being transmitted through MPI in the main step phase). Due to these factors the main simulation step routine became the most important study point in our further investigations.

The exchange of data fields stored in different domains is realized by means of non-blocking point-to-point MPI operations (MPI_Isend, MPI_Irecv, MPI_Wait) in routines responsible for the simulation of the 3D advection scheme (*trcadv*, 816/408 MPI calls used in each non-boundary/boundary domain), horizontal diffusion (*trchdf*, 7242/3621 MPI calls accordingly) and time integration (*trcnxt*, 102/51 MPI calls accordingly).

The size of a standard message is 72 kB for the advection and time integration routines (1821MB and 227.6 MB of data are accordingly transmitted) and 1kB for the horizontal diffusion routine (224.5 MB). Hence, the total amount of transmitted data for one execution step amounts to 2273.1 MB, even for this small test use case.

Therefore, the application should be classified as communication-intensive. The optimization of the currently implemented communication pattern will be important for the further application performance and scalability improvement. In the following section we give several optimization proposals for the current implementation of the communication based on the collected quantitative properties. Proposals for the optimization of I/O operations in particular by means of using MPI I/O are also identified. The identified proposals are accompanied with some preliminary estimates of their impact on the application performance.

Optimization proposals

The message transmission mechanism that is currently used for the inter-domain communication (blocking MPI calls) is uniform - the order, structure and type of separately transmitted data do not change within the execution in each simulation step. Up to 51 variables are independently communicated each using separate MPI communication calls. Due to this property messages can be rearranged into segments transmitting more than one variable per MPI call. The current implementation foresees a transmission of only one variable per MPI operation (in the proposed implementation that would be the lower-bound segment size). Encapsulation of additional data into a segment enlarges the segment size and reduces the total number of MPI calls for data transmission. The highest data encapsulation level is reached by packing all messages into one single segment (that would be the upper-bound segment size). The effect is even larger for a network with a high latency and low bandwidth. However, a procedure of packing/unpacking messages into/from a segment can require an additional computational overhead reducing fractionally the performance effect on the communication.

Therefore, a detailed study of the optimal number of segments (varying the segment size from the lower-bound to the upper-bound size) is required for both homogeneous and heterogeneous types of message encapsulation. In case of the heterogeneous encapsulation an additional evaluation of the optimal segment size is necessary. For this purpose the application was modified in order to allow the size of a segment to be explicitly specified by a user or an automatic tuning mechanism. The first investigations for the current test configuration have proved that the data encapsulation influences positively the overall performance. The highest impact on reducing the communication time was reached using the segments with the upper-bound size (e.g. in the advection computation routine the communication time was reduced by 50 %); surprisingly, herewith a computational overhead was not detected due to direct addressing of transmitted data blocks in memory.

The data encapsulation mechanism can be also beneficial for the realization of the all-to-root communication (currently implemented by means of blocking point-to-point MPI operations). However, for this pattern we have concentrated on using collectives instead of point-to-point operations, As the practical experience for the large-scale target platforms made clear [22, 23], for the observed type of communication the usage of collective gather and reduction MPI operations is an advantage. We have investigated in detail the current implementation of the communication scheme and developed a new message transferring mechanism that is based on collective MPI operations and provides encapsulation of data arrays into segments similarly to the approach described above. For the test configuration the usage of collective MPI operations reduces the communication time by 50%. However, it is important to note that the

optimal number of segments and the segment size for both analyzed types of MPI operations can differ for other configurations, target platforms, number of launched processes, implementations of the MPI library etc.

Summary and future direction of research

The OPATM-BFM is a scientific application that can make a strong benefit of using a grid e-Infrastructure. However, the shortcomings pertained to the current realization of the application message-passing communication pattern (e.g. non-optimal size of messages transmitted by means of MPI etc.) result into the application performance degradation that can become especially dramatic after porting the application to such a e-Infrastructure. Therefore, a requirement for reaching the highest productivity of applications on a grid is the optimal realization of the internal communication patterns. The article presents a detailed analysis of the message-passing communication pattern of the OPATM-BFM application. The optimization proposals for the communication pattern given in this article have allowed us to increase the productivity of the application on a cluster of workstations. The performance expectation for the application from porting to the grid has increased accordingly. On the other hand, the described improvements will also allow us to increase the application performance on the IBM SP-5 machine currently used for running the application's main operation procedure. Besides, the acquired results for the OPATM-BFM application will be also important for other scientific parallel applications that implement similar types of inter-process communication as described in the paper (e.g. all-to-one communication), in particular for applications in development. Furthermore, the presented technique used for the analysis should also support the OPATM-BFM application providers in further understanding the communication patterns in order to improve the load balance of the application for different use cases and define bottlenecks as well as work out a solution on how to resolve the shortcomings and maximize the performance and scalability of the application.

Other future research activities include further issues of the application integration within a grid e-Infrastructure (e.g. analysis of application requirements on additional software packages installed on the grid, data management, testing and debugging on grid worker nodes with further launching through a grid job from the user interface node, providing interactivity etc.). We will also concentrate our investigations on working out optimization proposals for several types of hardware and software architectures based on obtained results (focusing especially on facilities of currently provided DORII architecture). Furthermore, interesting possibilities of improving the communication include the development of an MPI-OpenMP hybrid programming model for the application.

Last but not least, in future we plan to use the application as a use case for improving the internal profiling utilities of MPI implementations, e.g. Open MPI.

References

1. See the official web page of the GMES project – <http://www.gmes.info>
2. See the official web page of the Marine Environment and Security for the European Area (MERSEA) European Integrated project – <http://www.mersea.eu.org>
3. Crise, A., Lazzari, P., Salon, S. and Teruzzi, A. 2008. MERSEA deliverable D11.2.1.3 - Final report on the BFM OGS-OPA Transport module, 21 pp.
4. See the description of the IBM SP5 machine on the CINECA's official web page – <https://hpc.cineca.it/docs/user-guide-zwiki/SP5UserGuide>
5. See the official web page of the DORII project – <http://www.dorii.org>
6. Ian Foster. Service-Oriented Science. Science 6 May 2005: Vol. 308. no. 5723, pp. 814 – 817
7. A. Chervenak, I. Foster, C. Kesselman, C. Salisbury, S. Tuecke. The Data Grid: Towards an Architecture for the Distributed Management and Analysis of Large Scientific Datasets. Journal of Network and Computer Applications, 23:187-200, 2001 (based on conference publication from Proceedings of NetStore Conference 1999).
8. Branislav Simo, Ondrej Habala, Emil Gatial., Ladislav Hluchy. Leveraging interactivity and MPI for environmental applications. Computing and Informatics, Vol. 27, 2008, 271-284.
9. See the official web page of the Italian Group of Operational Oceanography – <http://gnoo.bo.ingv.it>
10. Vichi, M., Pinardi, N. and Masina, S. 2007. A generalized model of pelagic biogeochemistry for the global ocean. Part I: Theory. Jou. Mar. Sys. 64, 89-109
11. See the official web page of the OGS short term forecasting system of the Mediterranean Marine Ecosystem – <http://poseidon.ogs.trieste.it/cgi-bin/opaopech/mersea>
12. Jack J. Dongarra, Steve W. Otto, Marc Snir, David Walker. A message passing standard for MPP and workstations. Communications of the ACM. Volume 39 , Issue 7 (July 1996), 84 – 90.
13. See the official web page of the Mediterranean Ocean Observing Network – <http://www.moon-oceanforecasting.eu>
14. Teruzzi, A., Lazzari, P., Salon, S., Crise, A., Solidoro, C., Mosetti, V., Santoleri, R., Colella, S. and Volpe, G. 2008. Assessment of predictive skill of an operational forecast for the Mediterranean marine ecosystem: comparison with satellite chlorophyll observations. MERSEA Final Meeting, Paris, 28-30 April 2008.

15. Richard L. Graham, Galen M. Shipman, Brian W. Barrett, Ralph H. Castain, George Bosilca, Andrew Lumsdaine. Open MPI: A High-Performance, Heterogeneous MPI. Proceeding of the conference HeteroPar '06, September 2006, in Barcelona, Spain. <http://www.openmpi.org/papers/heteropar-2006/heteropar-2006-paper.pdf>
16. MPI: A Message-Passing Interface Standard Version 2.1. Message Passing Interface Forum, June 23, 2008. <http://www.mpi-forum.org/docs/mpi21-report.pdf>
17. See the description of the cluster "Cacau" on the official web page of HLRS – <http://www.hlrs.de/hw-access/platforms/cacau/>
18. Andreas Knüpfer, Holger Brunst, Jens Doleschal, Matthias Jurenz, Matthias Lieber, Holger Mickler, Matthias S. Müller, and Wolfgang E. Nagel. The Vampir Performance Analysis Tool-Set. Tools for High Performance Computing, Springer, 2008, 139-156.
19. Rolf Riesen. Communication patterns. IEEE 2006, <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=01639567>
20. Josef Weidendorfer. Sequential Performance analysis with Callgrind and KCachegrind. Tools for High Performance Computing, Springer, 2008, 93-114.
21. Julian Seward, Nicholas Nethercote, Josef Weidendorfer and the Valgrind Development Team. Valgrind 3.3 - Advanced Debugging and Profiling for GNU/Linux applications. <http://www.network-theory.co.uk/valgrind/manual/>
22. O. Hartmann, M. Kühnemann, T. Rauber, G. Rüniger. Adaptive Selection of Communication Methods to Optimize Collective MPI Operations. Parallel Computing: Current & Future Issues of High-End Computing, Proceedings of the International Conference ParCo 2005, G.R. Joubert, W.E. Nagel, F.J. Peters, O. Plata, P. Tirado, E. Zapata (Editors), John von Neumann Institute for Computing, Jülich, NIC Series, Vol. 33, ISBN 3-00-017352-8, pp. 457-464, 2006.
23. Jelena Pješčević-Grbović, Thara Angskun, George Bosilca, Graham E. Fagg, Edgar Gabriel Jack J. Dongarra. Performance Analysis of MPI Collective Operations. Cluster Computing archive Volume 10 , Issue 2 (June 2007), 127 – 143.



Prof. Dr. Vladimir Svjatnny is the head of the department of computer science of the National University of technology of Donetsk. To his scientific interests belongs simulation of complex dynamic systems, parallel programming, and automated technology.



Dr. Alexey Cheptsov is a senior researcher at the department of computer science of the National University of technology of Donetsk. Throughout his scientific career he has conducted research and published in areas of simulation support of complex dynamic systems, deployment of technologically-oriented distributed simulation environments for diverse problem areas and parallel and high-performance simulation technology.



Dr. Rainer Keller got a PhD at the High-performance Computing Center of Stuttgart (University of Stuttgart). He is heading the working group application, models and tools. He is working in several international projects, such as Open MPI and PACX-MPI.



Stefano Salon graduated in Physics (1998) and got a PhD in Applied Geophysics and Hydraulics (2004) at the University of Trieste. Post-Doc at OGS from 2004 to 2007, he has been a researcher at OGS since December 2007. His research interests focus on turbulence in tidally-driven boundary layers, operational numerical forecasts of the Mediterranean ecosystem, dynamical downscaling of the ecosystem of the lagoon of Venice based on IPCC-SRES scenarios.

Дата надходження до редакції 17.10.2008 р.