IEEE

Proceedings of the 6th IEEE International Conference on

Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications

Volume 1

IDAACS'2011

The crossing point of Intelligent Data Acquisition & Advanced Computing Systems and East & West Scientists

September 15-17, 2011
Prague, Czech Republic

ORGANIZED BY

IEEE UKRAINE I&M / CI JOINT SOCIETIES CHAPTER

RESEARCH INSTITUTE FOR INTELLIGENT COMPUTER SYSTEMS, TERNOPIL NATIONAL ECONOMIC UNIVERSITY AND GLUSHKOV INSTITUTE OF CYBERNETICS, NATIONAL ACADEMY OF SCIENCES, UKRAINE

IN COOPERATION WITH

FACULTY OF ELECTRICAL ENGINEERING, CZECH TECHNICAL UNIVERSITY IN PRAGUE, CZECH REPUBLIC

SPONSORED BY

IEEE I&M/CI Chapter

Office of Naval Research GLOBAL Science & Technology

H/test®

Agilent Technologies
Authorized Distributor

Honeywell

IEEE UKRAINE SECTION

IEEE Czechoslovakia Section

I&M IEEE instrumentation & measurement society®

IEEE Region 8

River Publishers

Integrated Tools for Molecular Dynamics Simulation Data Analysis in the MolDynGrid Virtual Laboratory

O.V. Savytskyi [1], I.A. Sliusar [2], S.O. Yesylevskyy [3], S.G. Stirenko [4], A.I. Kornelyuk [1]
[1] Institute of Molecular Biology and Genetics, National Academy of Sciences of Ukraine
Akademika Zabolotnogo Str., 150, Kyiv-03680, Ukraine, savytskyi@moldyngrid.org, http://imbg.org.ua
[2] Information and Computer Centre, National Taras Shevchenko University of Kyiv,
Glushkova prosp., 4D, Kyiv, Ukraine, cluster@cluster.kiev.ua, http://grid.org.ua
[3] Institute of Physics, National Academy of Sciences of Ukraine
Prospect Nauki, 46, Kyiv-03039, Ukraine, yesint3@yahoo.com, http://www.iop.kiev.ua
[4] High-Performance Computing Center, National Technical University of Ukraine “Kyiv Polytechnic Institute”
Prospect Peremohy, 37, 03056, Kyiv-56, Ukraine, stirenko@hpcc.org.ua, http://ntu-kpi.kiev.ua

Abstract—**Existing molecular modeling software and different techniques of molecular dynamics trajectories analysis were evaluated in terms of subsequent integration into the MolDynGrid virtual laboratory services. GROMACS 4 molecular dynamics package contains the most comprehensive set of analytical programs, thus the standard analytical tools in MolDynGrid were based on this software. G_correlation tool was ported to GPU which dramatically increased its productivity. Scripting capabilities of VMD software were used for certain non-standard analysis techniques, while complex and computationally intensive techniques were implemented using Pteros molecular modeling library. Parallel execution of analytical programs was implemented in MolDynGrid by a set of internally developed scripts.**

Keywords—CUDA; GPU; grid; GROMACS; molecular dynamics; MolDynGrid; Pteros; web-portal

I. INTRODUCTION

MolDynGrid virtual laboratory was established for interdisciplinary studies in computational structural biology and bioinformatics, especially for molecular dynamics (MD) simulations of biological macromolecules and their complexes [1], [2]. Biomolecular simulations usually require vast computational resources and huge storage space for MD trajectories. The main objective of MolDynGrid project was to provide an efficient infrastructure for automation of MD simulation and trajectories analysis in Grid computing environments.

The web-portal of MolDynGrid virtual laboratory (http://moldyngrid.org) presents a convenient interface with four functional blocks – computational, analytical and educational block and MD simulation results database [1], [2].

The users of MolDynGrid web-portal are not required to be familiar with subtle technical aspects of Grid technologies which power the back-end of the portal. It is important that the operation of all services is fully transparent to the end-users. The portal back-end is highly tunable and can be configured to use the different grid middleware and infrastructure central services similarly to file catalogs and resource directories. MolDynGrid virtual laboratory is one of the first web-oriented virtual laboratories deployed in Ukrainian National Grid infrastructure (UNG) [3], which is grown rapidly in a last few years.

Currently, the MolDynGrid is being used for performing MD simulations of different proteins, such as human immunodeficiency virus protease (HIV-1 protease), mammalian and eubacterial tyrosyl-tRNA synthetases, isolated functional N- and C-terminal domains of tyrosyl-tRNA synthetase, EMAP II cytokine, transfer RNAs and specific aminoacyl-tRNA synthetase–tRNA complexes. Common file sizes of MD trajectories and analysis results may exceed 250 GBytes.

Computational block of the web-portal was deployed in 2009 and provides a wizard-like interface for specification of MD simulation parameters and submitting of computational job to the grid environment. Since 2009 about 3500 MD trajectories were successfully computed using resources provided for MolDynGrid by the UNG infrastructure members [2], [3].

II. ANALYTICAL BLOCK

After MD trajectory was computed, it should be analyzed in order to determine the MD characteristics which may have scientific value. After a preliminary analysis, all potentially valuable trajectories were included into the MD simulations database, which allows the portal users to run deeper analysis of simulation data. Many

This work was supported by National Academy of Sciences of Ukraine (grant 15/2010 of the State scientific and technical program of implementation and application of Grid technologies).

trajectories were dropped on the first stage due to their instability or incorrectness of simulation parameters. MolDynGrid users submit a large number of MD simulation jobs to the grid, and hence a reasonable number of valuable MD trajectories can be selected for a further analysis.

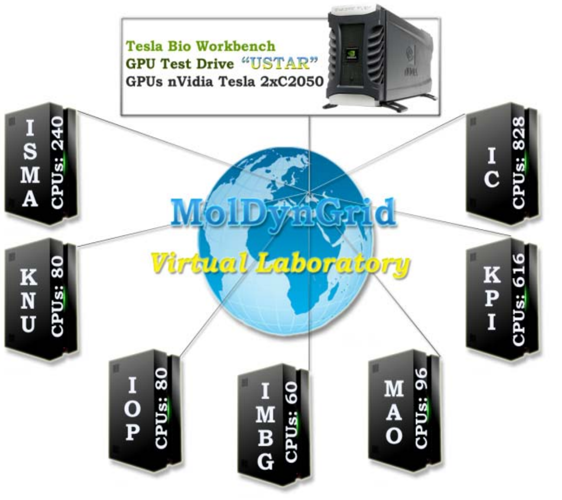

Figure 1. MolDynGrid computational resources provided by Ukrainian National Grid infrastructure. GPU Test Drive was added for testing on Tesla.

GROMACS (GROningen MAchine for Chemical Simulations) is a versatile software package for molecular dynamics simulations, i.e. solving Newtonian equations of motion for systems with hundreds to millions of particles. It was primarily designed for biological molecules like proteins, lipids, nucleic acids and their complexes, but could also be used for non-biological systems, e.g. polymers [4].

Now the GROMACS 4.5 is the primary research software in MolDynGrid virtual laboratory. GROMACS 4.5 includes about 80 tools for MD trajectories analysis. Most of these tools are single-threaded and thus do not support multicore and multinode parallel execution of a single tool for a single MD trajectory. So, invoking strictly serial analysis tools just after parallel multicore MD simulation in the context of the same job would result in a waste of computing resources. Another common case of analysis includes a bunch of pre-computed trajectories, which should be processed by the same sequence of analysis programs. Running such analysis in serial manner would be also inefficient.

The Distributed Analyzer Script (DAS) was developed to solve these problems. DAS was implemented into MolDynGrid as a job script for PBS (Portable Batch System) resource management system. It takes a list of MD trajectories and a list of analysis tools to invoke on input, builds a full list of needed analysis tools invocations and employs PBS-specific process-spawning mechanisms and spreads independent processes in parallel over the cores allocated to the job.

DAS provides the MolDynGrid users with opportunity not only to combine multiple single-threaded analysis tool invocations in a parallel PBS job, but also enable selection of several analysis profiles. The first profile includes generic analysis tools, such as g_rms, g_gyrate, etc. After observing the processing results, scientists can then choose various tools of the second analysis profile, which usually includes g_rmsf, g_cluster, g_correlation, etc. for deeper analysis.

Currently, DAS is being used on IMBG (Institute of Molecular Biology and Genetics) cluster and in future it will be adapted to the Grid environment and integrated into the MolDynGrid web-portal.

It is well understood that any truly innovative research requires unconventional methods of analysis. For those analytical problems, that cannot be solved by GROMACS standard tools and do not require huge processing power, the VMD (Visual Molecular Dynamics) molecular analysis program can be employed [5]. Besides its usual visual mode, VMD can be launched in a batch mode and execute custom analysis scripts by means of its built-in TCL interpreter. VMD provides a comprehensive API for molecular analysis, including reading and writing of trajectories and structures in various formats, geometric transformation, flexible selection of atoms, etc.

VMD was used to implement another MolDynGrid self-developed tool, the Contacts Analyzer Script (CAS). It is a script for the analysis of arbitrary interfaces between two groups of atoms in MD structure. This script reports the comprehensive statistics of contacts on the level of atoms and residues, which are determined by a distance between the centers of atoms [6]. CAS supports both interactive and the command-line modes and will be included into the analytical block of MolDynGrid web-portal. VMD-powered analysis, like CAS, can also be launched via the DAS script.

Despite great flexibility the TCL scripts provide poor performance in comparison to compiled programs and thus cannot be used for computation-intensive analysis.

The module of the non-standard analysis, which requires intensive computations, will be implemented using the open source molecular modeling library Pteros, being developed earlier [6]. The latest source code of Pteros is available from the SVN repository at http://sf.net/projects/pteros/. Pteros was implemented on high-level C++ and uses highly optimized Eigen library (http://eigen.tuxfamily.org) for vector and matrix operations. This allows the combining of high performance with extremely simple high-level API. Pteros has shallow learning curve and may be targeted to researchers and students. Currently, the library consists of the core, the modules for various analysis techniques and the bindings for the Python programming language. The built-in GNM computations and the HCCP (Hierarchical Clustering of the Correlation Patterns) technique for protein domain identification are unique features of Pteros [7]. The library is already used in practical applications with great success [8]–[11] and will be integrated into the MolDynGrid virtual laboratory.

Besides the standard analysis tools provided by GROMACS and MolDynGrid self-developed tools, there are several analysis tools, which are commonly used inside MolDynGrid and employ intensive computations. One of the most valuable tools is g_correlation instrument [12]. It computes all correlated motions in biological macromolecules using innovative algorithm, which handles non-linear correlations. Usually g_correlation can run for several months on a single CPU for 100 nanoseconds MD trajectory with approximately 10 000 atoms. As a rule g_correlation supports parallelization using MPI, but unfortunately it does not compile with current version of GROMACS. This problem was recently solved by implementing a patch, which enabled support for GROMACS 4.0 and 4.5. Adaptation of g_correlation algorithms for GPUs has also been investigated.

III. GPU-Test Drive

Recently the GPU Test Drive system was installed in Ukraine within the scope of NVIDIA's Tesla Bio Workbench Project [13]. This project aims at evaluation of performance of molecular dynamics simulations and trajectory analysis computations employing GPU accelerator support in specialized software including GROMACS 4.5, AMBER 11 and NAMD 2.7.

The test system was equipped with two Tesla C2050 GPGPUs (General-purpose computing on graphics processing units) with the latest drivers and CUDA (Compute Unified Device Architecture) toolkit [14], 4 GB of RAM and high-performance Intel Core i7 CPU. OpenMPI was also installed to have ability to compare GPGPU and multicore CPU performance.

TABLE I. Performance of Molecular Dynamics Simulations

Size of matrix Processor Unit	*128*	*256*	*512*	*1024*	*2048*	*4096*
	Execution time, ms					
Core™ i7-965	5,45	18,11	58,93	212,53	850,61	3857,10
GF8600GT	3,02	5,26	12,50	20,47	59,68	199,34
Tesla C2050	0,95	1,23	2,38	3,50	5,91	13,47
	Acceleration rate					
GF8600GT	1,80	3,44	4,71	10,38	14,25	19,35
Tesla C2050	5,72	14,76	24,81	60,71	143,90	286,31

From the MolDynGrid perspective, further research using this platform aims at evaluating feasibility of porting of trajectory analysis tools to CUDA architecture. First efforts in this direction were made for the g_correlation tool. It was partly rewritten to implement CUDA support and made available for public use within the scope of GPU Test Drive project.

IV. G_correlation for CUDA

The methods and algorithms involved in the g_correlation tool were studied and CUDA-based version was developed earlier [Stirenko et al., unpublished results].

Considering that GPU has significantly different architecture than conventional CPUs, the computational algorithms have to be completely re-implemented to leverage the power of the architecture.

G_correlation searches for correlated motions using the Kraskov's algorithm [15]. This implementation algorithm is written in CUDA language and consists of two parts:

1. A search of mutual information between two points.
2. Reduction of the found values of mutual information for all pairs of points.

The first step could not be well parallelized. The principle of coarse-grained parallelism was used: every stream of GPU chooses the pair of points from memory and executes operations on them in accordance with the Kraskov's algorithm. The result is stored in the shared memory.

The second step, which is the most computation-intensive, is a reduction of array with natural parallelism. All available CUDA optimizations were used at this stage (utilization of shared memory, branch minimization, conflicts elimination using access to the banks of memory, coalescing queries to memory), which reduces the execution time substantially (Table I).

The algorithm was tested using Tesla C2050 GPU (448 cores). The results of testing are presented in Fig. 2 and 3.

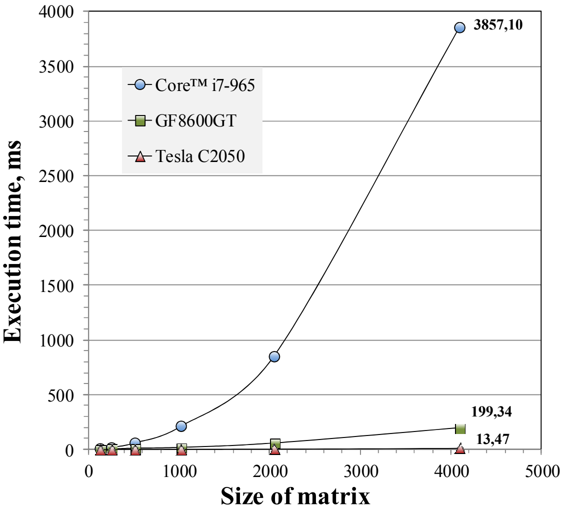

Figure 2. Dependence of execution time on the size of correlation matrix (lower values are better). Tesla C2050 shows the best result with 13.47 ms.

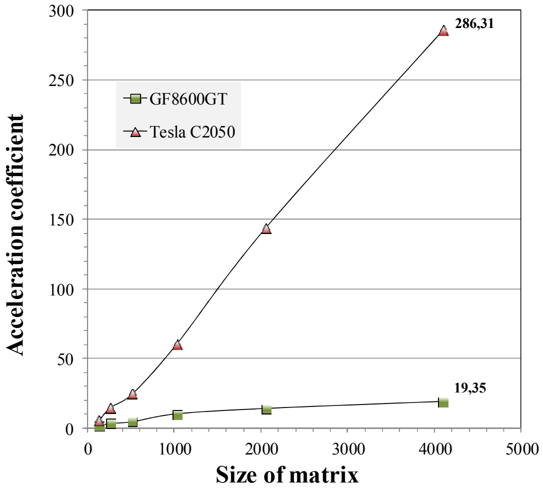

Figure 3. Dependence of the acceleration rate on the size of correlation matrix for GF8600GT and Tesla C2050.

V. Conclusions

MolDynGrid virtual laboratory has grown into the unique service complex for performing MD simulations in Grid environment. It provides powerful facilities for running the resource-intensive biomolecular simulations and performing state-of-the-art analysis of MD trajectories. Analytical capabilities currently include the full set of standard GROMACS tools as well as self-developed scripts, i.e. contacts analysis script (CAS) executed in VMD, the framework for recourse-intensive non-standard analysis based on the Pteros library and distributed parallel execution of analytical jobs by means of DAS. Currently MolDynGrid participates in the GPU Test Drive project providing the porting of g_correlation tool into CUDA.

Acknowledgment

The authors wish to acknowledge Andrii Salnikov (KNU), Volodymyr Saviak and Volodymyr Osnach (USTAR), Vladimir Kazakov (Axinom, Estonia), Oleg Zhurakivskyi (IEC, NAU), Andrew Adinetz (RCC, MSU) and NVIDIA Russian office for invaluable help and support. Ukrainian National Grid infrastructure members are acknowledged for providing computational resources.

References

[1] A.O. Salnikov, I.A. Sliusar, O.O. Sudakov, O.V. Savytskyi, A.I. Kornelyuk, “MolDynGrid Virtual Laboratory as a Part of Ukrainian Academic Grid Infrastructure,” *Proceedings of the 5-th IEEE Workshop IDAACS 2009*, Rende (Cosenza), Italy, pp. 237-240, 21-23 September 2009.

[2] A. Salnikov, I. Sliusar, O. Sudakov, O. Savytskyi, A. Kornelyuk, “Virtual Laboratory MolDynGrid as a Part of Scientific Infrastructure for Biomolecular Simulations,” *International Journal of Computing*, Vol. 9, Issue 4, 2010, pp. 294-300.

[3] Ukrainian National Grid community, 2010. http://grid.nas.gov.ua, http://grid.bitp.kiev.ua.

[4] B. Hess and C. Kutzner and D. van der Spoel and E. Lindahl, “GROMACS 4: Algorithms for Highly Efficient, Load-balanced, and Scalable Molecular Simulation,’ *J. Chem. Theory Comput. 4*, 2008, pp. 435-447.

[5] W. Humphrey, A. Dalke and K. Schulten, "VMD - Visual Molecular Dynamics," *J. Molec. Graphics*, vol. 14, 1996, pp. 33-38.

[6] O. Savytskyi, R. Nikolaenko and A. Kornelyuk, “Molecular Dynamics Simulation of Mutant Tyrosyl-tRNA Synthetase Accociated with Charcot-Marie-Tooth Neuropathy Reveals the Stabilization of Enzyme Dimer Interface,” *Proceedings of Workshop Physical Chemistry of Biointerfaces*, CIC biomaGUNE, Donostia - San Sebastian, Spain, p. 21, 19-24 July 2010.

[7] S.O. Yesylevskyy, V.N. Kharkyanen, A.P. Demchenko, “Dynamic Protein Domains: Identification, Interdependence and Stability,” *Biophysical Journal*, Volume 91, Issue 2, pp. 670-685.

[8] S.O. Yesylevskyy, O.V. Savytskyi, K.A. Odynets and A.I. Kornelyuk, “Interdomain Compactization in Human Tyrosyl-Trna Synthetase Studied by the Hierarchical Rotations Technique,” *Biophysical Chemistry*, Vol. 154, Issues 2-3, pp. 90-98, March 2011.

[9] S.O. Yesylevskyy, “New Technique of Identifying the Hierarchy of Dynamic Domains in Proteins using the Data of Molecular Dynamics Simulations,” *Biopolymers and Cell*, 26(2), 2010, pp. 146-152.

[10] S.O. Yesylevskyy, “Identifying the Hierarchy of Dynamic Domains in Proteins: the Comparative Study of HDWA and HCCP Techniques,” *Biopolymers and Cell*, 26(4), 2010, pp. 311-346.

[11] S.O. Yesylevskyy, “Identifying the Hierarchy of Dynamic Domains in Proteins using the Data of Molecular Dynamics Simulations,” *Protein & Peptide Letters*, 17(4), 2010, pp. 507-516.

[12] O.F. Lange, H. Grubmüller, “Generalized Correlation for Biomolecular Dynamics,” *PROTEINS: Structure, Function, and Bioinformatics*, 62, 2006, pp. 1053-1061.

[13] Tesla Bio Workbench, http://www.nvidia.com/object/tesla_bio_workbench.html.

[14] nVidia CUDA Toolkit 3.2 (developer zone). http://developer.nvidia.com/object/cuda_3_2_downloads.html.

[15] A. Kraskov, H. Stogbauer, P. Grassberger, “Estimating Mutual Information,” *Phys Rev E*, 69, 066138, 2004.