Volume 7, Issue 4, April 2017 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

A Study on Utility of Cloud Computing Resources for Accelarating Research in Biomolecular Simulations

Rahul Wargad
Research Scholar AIMS Pune,
Maharashtra, India

Dr. Manimala Puri
Director JSPM Pune,
Maharashtra, India

Abstract: The molecular biology is highly dynamic in nature, innumerable conformational states are accessible at physiological temperatures. Since molecular biology is highly dynamical in nature, the static illustrations of proteins, nucleic acids, and other bio molecular structures are normally printed in books. A given bio molecule also samples a rapidly fluctuating local environment comprised of other biopolymers, small molecules, water, ions, etc. that diffuse to within a few nano-meters, leading to inter-molecular interactions and the formation of supra molecular assemblies. These intra- and inter-molecular contacts are governed by the same physical principles (forces, energetic) that characterize individual molecules and inter-atomic interactions, thereby enabling a unified picture of the physical basis of molecular interactions from a small set of fundamental principles. Computational approaches are well-suited to studies of molecular interactions, from the intra-molecular conformational sampling of individual proteins (such as membrane receptors or ion channels) to the diffusional dynamics and inter-molecular collisions that occur in the early stages of formation of cellular-scale assemblies. To study such phenomena, two major lineages of computational approaches have developed in molecular biology: physics–based methods (often referred to as simulations) and informatics–based approaches (often termed the data-mining or machine learning approach to knowledge extraction via statistical inference). An advantage of the former approach is its physical realism, while an advantage of the latter approach is its potential to illuminate the evolution of a genetically related to group of organisms as distinguished from the development of the individual organism relationships (evolutionary features). This paper highlights the utility of cloud computing for bio-molecular simulation (i.e. the physics based method) and suggests rules for setting up of cloud computing facilities to facilitate the research using bio-molecular simulation.

Keywords : Cloud Computing, Biomolecular Simulation, Molecular Modelling

I. INTRODUCTION

Molecular simulation offers a uniquely powerful approach to analyze biomolecular structure, mechanism, and dynamics; this is possible because the molecular contacts that define a complicated biomolecular system are governed by the same physical principles (forces, energetic) that characterize individual small molecules, and these simpler systems are relatively well-understood.

Simulations using powerful computers can show how biological molecules 'work' in atomic detail. For example, molecular simulations can show drugs bind to their biological targets, how enzymes catalyse reactions, and how proteins fold into their functional forms. Biomolecular simulation is a vibrant and growing area, making increasingly significant contributions to biology. It is an area of growing international importance. Simulations of biological molecules complement experiments in building a molecular-level understanding of biology: they can test hypotheses and interpret and analyse experimental data in terms of interactions at the atomic level. A wide variety of simulation techniques have been developed, applicable to a range of different problems in biomolecular science. Biomolecular simulations have already shown their worth in helping to analyse how enzymes catalyse biochemical reactions, and how proteins adopt their functional structures e.g. within cell membranes. They contribute to the design of drugs and catalysts, and in understanding the molecular basis of disease. Simulations have played a key role in developing the conceptual framework now at the heart of biomolecular science, that is, the understanding that the way that biological molecules move and flex - their dynamics - is central to their function. Developing methods from chemical physics and computational science will open exciting new opportunities in biomolecular science, including in drug development and biotechnology. Much biomolecular simulation demands high end computing (HEC) resources: e.g. large-scale simulations of biological machines such as the ribosome, proton pumps and motors, membrane receptor complexes and even whole viruses. A particular challenge is the integration of simulations across length and timescales: different types of simulation method are required for different types of problems).

Biomolecular Simulations are contributing increasingly to areas such as biotechnology, drug design, biocatalysis and biomedicine. A better knowledge of biomolecules is the key to understanding mechanistic details of the various biochemical processes that occur in all living cells. The biomolecular structure, dynamics and function span multiple scales of time and length. In the past, experimental techniques have provided a wealth of information into the working of biomolecules; more recently theoretical and computational multi-scale modelling techniques based upon biomolecular simulations continue to provide novel insights. Till recently, the computing power required for simulating the length and time scales relevant to biomolecules were beyond the reach of even the fastest supercomputers. In particular, the dynamics and functions of biomolecules span more than 15 orders of magnitude in time; the computing power falls short by 4-6 orders of magnitude in its ability to simulate the desired time-scales.

The fundamental question for biomolecular simulation frameworks is whether multiple cores per processor can provide performance commensurate with initial expectations. The shared memory and I/O (network) bandwidth of multiple cores in a socket draws into question both how efficiently an application can use multiple cores and what methods provide the highest efficiency. In this preliminary study, we characterize computation, communication and memory efficiencies of a scalable bio-molecular simulation framework based on cloud computing infrastructure.

II. OBJECTIVES

This paper explains the concept and challenges in bio molecular simulation It identifies different advanced Molecular Modelling techniques and benefits and utility of cloud computing to address current and future Bioinformatics challenges.

III. DESIGN

This paper is conceptual in nature wherein qualitative method has been used to substantiate the significant issues in biomolecular simulation. An attempt is made to explore the possible solutions to handle this challenge in order to make certain vital observations to lay down conclusion.

IV. BIOMOLECULAR SIMULATION CHALLENGES

Numerous applications use molecular dynamics (MD) for biomolecular simulations. MD and related techniques can be defined as computer simulation methodology where the time evolution of a set of interacting particles is modelled by integrating the equation of motion. The underlying MD technique is based on the law of classical mechanics—most notably Newton’s law, F = ma. The MD steps performed in MD engines consist of three calculations: determining energy of a system and forces on atoms centers, moving the atoms according to forces, and adjusting temperature and pressure. A typical bimolecular simulation contains atoms for solute, ions, and solvent molecules. The force on each atom is represented as the combination of the contribution from forces due to atoms that are chemically bonded to it and non-bond forces due to all other atoms.

MD simulations enable the study of complex, dynamic processes that occur in biological systems. MD methods are now routinely used to investigate the structure, dynamics, functions, and thermodynamics of biological molecules and their complexes. The types of biological activity that have been investigated using MD simulations include protein folding, enzyme catalysis, conformational changes associated with bimolecular function, and molecular recognition of proteins, DNA, and biological membrane complexes. Biological molecules exhibit a wide range of time and length scales over which specific processes occur, hence the computational complexity of an MD simulation depends greatly on the time and length scales considered. With an explicit solvation model, typical system sizes of interest range from 20,000 atoms to more than 1 million atoms; if the solvation is implicit, sizes range from a few thousand atoms to about 100,000. The simulation time period can range from pico-seconds (10-12 seconds) to a few microseconds or longer (>10-6 seconds) on contemporary platforms.

Although protein folding simulations are a tremendously intensive use of computer power, it should be noted that simulation expense is not limited to protein folding. An atomistic understanding of PagP function will be useful for the rational design of small molecule inhibitors. To determine the substrate binding pathway and the catalytically relevant bound state, Neale embedded PagP in an explicit phospholipid bilayer solvated by explicit water and simulated under equilibrium conditions using molecular dynamics simulations. The pathway by which the SN-1 acyl chain of a donor phospholipid binds the internal hydrophobic furrow of PagP and adopts a confirmation that may be permissive to the acyltransferase/phospholipase activity of PagP. Experimental studies have verified the proposed acyl chain entry route, and theoretical predictions of putative inhibitors of the proposed transition state conformation are planned.

Scenarios such as these are increasingly important in a research context. The high resolution of a MD simulation can provide an incredibly detailed hypothesis focusing the efforts of experimental confirmation. The computational demands of MD renders this approach intractable for all but a few relatively small molecular systems or large supercomputing centres.

V. ADVANCED MOLECULAR MODELLING TECHNIQUES

Most molecular modelling studies involve three stages:

- In the first stage a model is selected to describe the intra- and inter- molecular interactions in the system. The two most common models that are used in molecular modelling are *quantum mechanics* and *molecular mechanics. These models enable the energy of any arrangement of the atoms and molecules in the system to be calculated, and allow the modeller to determine how the energy of the system varies as the positions of the atoms and molecules change.*
- The second stage of a molecular modelling study is the calculation itself, such as an energy minimisation, a molecular dynamics or Monte Carlo simulation, or a conformational search.
- Finally, the calculation must be analysed, not only to calculate properties but also to check that it has been performed properly.

VI. CLOUD COMPUTING INFRASTRUCTURE FOR BIO-MOLECULAR RESEARCH

The truth when shared computing resources were not available is that the molecular modelling used to be restricted to a small number of scientists who had access to the necessary computer hardware and software. Its practitioners wrote their own programs, managed their own computer systems and mended them when they broke down.

The current technological improvement of molecular biology techniques results in a huge expansion of biological data, whose satisfactory management and analysis are a challenging task. In particular, the adoption of an adequate computational infrastructure is becoming too expensive, in terms of costs and efforts of establishment and maintenance, for small-medium biotechnological laboratories.

VII. CASE STUDIES OF COMPUTING DEPLOYMENTS FOR SCIENTIFIC RESEARCH

Case I. Campus SGE.

The Sun Grid Engine (SGE) at the University of Notre Dame is maintained as a dedicated platform for running high performance scientific applications. The compute nodes run Red Hat Enterprise Linux (RHEL) as their operating environment. The compute nodes here are typically composed of high-end hardware.

- The jobs are submitted to the compute nodes via the SGE batch submission system.
- The workers are queued and submitted as jobs to this grid.
- Upon being scheduled, the workers connect to the master and execute the assigned workloads.

Case II. Amazon EC2.

- The Elastic Compute Cloud or EC2, built by Amazon.com, is a platform that allows virtual machine instances to be requested, allocated, and deployed on demand by users.
- Different instance sizes are provided with varying hardware configurations to satisfy different requirements and workloads of the users.
- The instances allocated can be installed and run with different Linux operating system flavors and kernels and their operating environments can also be customized.
- Since the instances can be installed and customized to run a Linux environment, the implementation to EC2 is similar to SGE with advantage of offering required computing configuration.

Case III. Microsoft Azure.

- The Windows Azure platform, from Microsoft, offers virtual instances running an image of the Azure operating system. The virtualized instance is offered through the Azure hypervisor and provides an operating environment based off the Windows Server 2008 R2 VM system.
- There are two computational roles offered in the platform - the web role that serves as the front end interface to the allocated compute instances, and the worker role that serves as the core computing unit that runs tasks and applications.
- As a result of this two tiered architecture, wrapper scripts can be built that communicate to the web role and invoke workers on the worker roles.

VIII. CLOUD COMPUTING REQUIREMENTS FOR BIOMOLECULAR SIMULATION

To build, design, cloud computing framework for creating or modifying high performance applications for the cloud, we establish a set of rules to follow, which are presented below:

Rule 1: Scalability. The cloud computing framework must allow applications to scale in size.

Rule 2: Resource Adaptability. The framework must dynamically harness resources as they become available and allow applications to utilize these resources to progress in their execution. (***Example hypervisor in case of Microsoft Azure)***

Rule 3: Fault tolerance. The framework must provide robust fault-tolerance and allow the application to continue execution even in the presence of hardware failures, communication failures, site failures, and execution errors and failures. The framework must dynamically rerun failed tasks or seamlessly migrate them to other sites in the event of such failures.

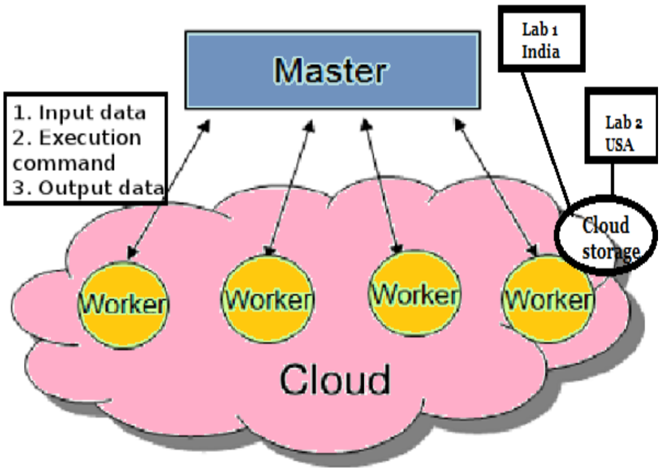
Rule 4: Portability. The framework must be able to deploy and run the application on different cloud computing platforms with minimal effort and intervention from the user.

Rule 5: Platform independence. It must be independent of any platform, operating system, and hardware characteristics from the application.

Rule 6: Ease of effort. The framework must allow users to migrate their applications with minimal effort to the cloud.

Cloud Computing framework suggested for Bio molecular Simulation

The Work Queue framework is based on the master-worker paradigm, where multiple worker processes can receive and execute workloads sent by the master. The master coordinates the execution of a given application by assigning and scheduling work units to each of the workers. Figure illustrates the master-worker architecture of Work Queue. The arrows describe the communications between the master and worker. The communications occur at the following times: (a) transfer of input including application executables, binaries, input files etc., from master to workers, (b) communication of the task execution commands and their arguments by master to its workers, and (c) transfer of output including output files and logs from workers to master.



The workers are deployed as executables on the cloud platform and they are invoked and run as jobs on these platforms through their respective job submission interfaces. The workers can be compiled, installed, and run on any Portable Operating System Interface (POSIX) compliant environment. This implies that the worker can virtually be deployed and run on any operating environment including Microsoft Windows based environments.

The master script implemented by the user is often relatively simple. This is because the master script only contains the input file specifications, the executables required for task execution, the output file specifications, and the task execution command and arguments.

IX. CONCLUSION

Advanced computational methods like cloud computing are finding increasing use in the area of the bio-molecular simulation . In this paper we have extended the basic masker –worker model where multiple worker processes can receive and execute workloads sent by the master. The master coordinates the execution of a given application by assigning and scheduling work units to each of the workers. This approach can yield an improvement in the overall simulation efficiency. The data in central cloud repository can not only allow for extensive and valuable comparisons to be made between related simulations, thereby yielding more and more reliable biochemical interpretation, but will also allow data to be readily shared between laboratories.

REFERENCES

[1] On the Path to Enable Multi-scale Biomolecular Simulations on PetaFLOPS Supercomputer with Multi-core Processors Sadaf R. Alam and Pratul K. Agarwal Computer Science and Mathematics Division Oak Ridge National Laboratory Oak Ridge, TN, USA 37831

[2] An Introduction to Biomolecular Simulations and Docking Cameron Mura, Charles E. McAnany

[3] The UK High-End Computing Consortium for Biomolecular Simulation , Mulholland, Professor AJ, University of Bristol

[4] Cloud Infrastructures for In Silico Drug Discovery: Economic and Practical Aspects Daniele D'Agostino,Andrea Clematis, Alfonso Quarati

[5] Data model, dictionaries, and desiderata for biomolecular simulation data indexing and sharing , Julien C Thibault, Daniel R Roe, Julio C Facelli and Thomas E Cheatham

[6] Molecular modeling : principles and applications / Andrew R. Leach ISBN 0-582-38210-6 (pbk) 2. ed. Harlow: Prentice Hall, 2001

[7] Converting A High Performance Application to an Elastic Cloud Application Dinesh Rajan, Anthony Canino, Jesus A Izaguirre, and Douglas Thain Department of Computer Science and Engineering University of Notre Dame Notre Dame, Indiana 46556

[8] K. Keahey, R. Figueiredo, J. Fortes, T. Freeman, and M. Tsugawa, "Science clouds: Early experiences in cloud computing for scientific applications," 2008.

[9] G. Juve, E. Deelman, K. Vahi, G. Mehta, B. Berriman, B. Berman, and P. Maechling, "Scientific workflow applications on amazon ec2," in E-Science Workshops, 2009 5th IEEE International Conference on, December 2009, pp. 59–66.

[10] C. Hoffa, G. Mehta, T. Freeman, E. Deelman, K. Keahey, B. Berriman, and J. Good, "On the use of cloud computing for scientific workflows," in eScience, 2008. eScience '08. IEEE Fourth International Conference on, 2008, pp. 640 –645.

[11] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," in Grid Computing Environments Workshop, 2008. GCE '08, 2008, pp. 1 –10.

[12] L. Yu and et al., "Harnessing parallelism in multicore clusters with the all-pairs, wavefront, and makeflow abstractions," Journal of Cluster Computing, vol. 13, no. 3, pp. 243–256, 2010.

[13] Y. Sugita and Y. Okamoto, "Replica-exchange molecular dynamics method for protein folding," Chemical Physics Letters, vol. 314, no. 1-2, pp. 141 – 151, 1999.

[14] Y. Sugita and Y. Okamoto, “Replica-exchange molecular dynamics method for protein folding,” *Chemical Physics Letters,* vol. 314, no. 1-2, pp. 141 – 151, 1999.

[15] S. Ferretti, V. Ghini, F. Panzieri, M. Pellegrini, and E. Turrini, “QoS-aware clouds,” in Proceedings of the 3rd IEEE International Conference on Cloud Computing (CLOUD ’10), pp. 321–328, July 2010.

[16] S. P. Ahuja and S. Mani, “The state of high performance computing in the cloud,” Journal of Emerging Trends in Computing and Information Sciences, vol. 3, no. 2, pp. 262–266, 2012.

[17] D. Salomoni, A. Italiano, and E. Ronchieri, “WNoDeS, a tool for integrated Grid and Cloud access and computing farm virtualization,” Journal of Physics, vol. 331, no. 5,Article ID52017, 2011.

[18] Q. Buyya, J. Broberg, and A.M. Goscinski, Cloud Computing: Principles and Paradigms, JohnWiley & Sons, 2011.