

Available online at www.sciencedirect.com**SciVerse ScienceDirect**

Procedia Computer Science 9 (2012) 136 – 145

Procedia
Computer Science

International Conference on Computational Science, ICCS 2012

A VMD Plugin for NAMD Simulations on Amazon EC2

Adam K.L. Wong^{a*}, Andrzej M. Goscinski^a^a*School of Information Technology, Deakin University, Waurn Ponds, VIC3216, Australia*

Abstract

VMD and NAMD are two major molecular dynamics simulation software packages, which can work together for mining structural information of bio-molecules. Carrying out such molecular dynamics simulations can help researchers to understand the roles and functions of various bio-molecules in life science research. Recently, clouds have provided HPC clusters on demand that allow users to benefit from their flexibility, elasticity, and lower costs. Although cloud computing promises to provide seamless access to HPC clusters through the abstraction of services, which hide the details of the underlying software and hardware infrastructure, users without in depth computing knowledge are still forced to cope with many low level system and programming details. Therefore, we have designed and developed a software plugin of VMD, which can provide an integrated framework for NAMD to be executed on Amazon EC2. The proposed Amazon EC2 Plugin for VMD frees users from performing many tedious computing tasks such as launching, connecting and terminating Amazon EC2 compute instances; configuring a HPC cluster; and installing middleware and software applications before the system is readily available for any scientific investigation. This allows VMD/NAMD users to spend less time getting applications to work on HPC clusters but more time for bio-research.

Keywords: Molecular Dynamics Simulation; High Performance Computing; Cloud; VMD; NAMD; Amazon EC2

1. Introduction

Many research problems require running computationally demanding software simulations on high performance computing (HPC) systems. One of such areas is the molecular dynamics simulation of bio-molecules in life science research. Molecular dynamics simulation is an important tool for biological scientists to study the physical basis of the structure and function of proteins (bio-molecules) since the internal motions of individual atoms play an essential role in the functional mechanism of proteins in living organisms. NAMD [11] is a parallel molecular dynamics program designed for high-performance simulation of large bio-molecular systems. This parallel software program can be scaled well to run on a large number of processors of computer clusters or a large number of cores in graphics processing unit (GPU) hardware. NAMD usually works together with its sister molecular graphics program VMD [20], which provides easy-to-use tools that explore structural information of bio-molecules. In a nutshell, VMD is a molecular visualization program for simulation preparation, displaying, animating, and analyzing large bio-molecular systems using 3D graphics and built-in scripting. NAMD and VMD are distributed

* Corresponding author. Tel.: +61-3-52271483; fax: +61-3-52272028; *E-mail address:* aklwong@deakin.edu.au.

free of charge with source codes and have supports on various OS platforms such as Linux, Windows and Mac OS.

An important component included in the VMD package is its plugin interface, which provides a means for extending VMD at run-time without the necessity to recompile the program. By using plugins, new functionalities or updated codes can be dynamically loaded from Tcl/Python scripts and shared object files for Linux/Unix systems, or dynamic link libraries for Windows systems [21]. This also allows third party developers to distribute extensions to the VMD user community. Common scripting extensions, which implement new commands and user interfaces for performing tasks such as structure building, data importing and exporting, simulation setup, molecule visualization as well as result analyzing, can be found in [21].

Traditionally, organizations with high performance computing needs have been required to fund expensive and in-house compute clusters by either purchasing and maintaining them or paying an inflexible amount of subscription fee to HPC service providers. The demand for high-performance cluster computing often exceeds the financial capacity of many organizations because of these upfront investments. Thus, many projects are cut altogether or application jobs wait in long queues to access shared resources. Recently, some public cloud vendors including the Amazon Elastic Compute Cloud (Amazon EC2) [1] have provided computer instances, which are specifically designed for running HPC applications. Thus, business and researchers now have access on demand to high performance computing facilities that they need with a very flexible pay-as-you-go pricing method. HPC Cloud gives users the opportunity to test and run their parallel applications in the cloud at a price and performance level that can leverage their budgets and computation requirements. Furthermore, it also provides the ability to scale on-demand as the users' requirements change.

Although cloud technologies offer seamless access to HPC clusters through abstractions of services that hide the details of the underlying software and hardware infrastructure, users without in depth computing knowledge are forced to cope with many details to set up and configure a HPC cluster; and install middleware and software applications before the system is available for any scientific investigation. A typical VMD/NAMD procedure requires users to (i) prepare simulation input data in VMD on a host computer; (ii) prepare a HPC cluster which must be NAMD ready; (iii) transfer the prepared simulation input data from the host computer to the HPC cluster and start the simulation; and (iv) transfer the simulation result back to the host computer for analyzing in VMD.

To relieve researchers of molecular dynamics from performing these tedious computing tasks, we propose a generic solution by integrating VMD, NAMD and Amazon EC2 together into a software plugin of VMD. This so called Amazon EC2 Plugin for VMD supports NAMD simulations on the Cluster Compute Instances of EC2. It allows users to 1) create instantly a HPC cluster on Amazon EC2; 2) submit a NAMD parallel simulation from VMD to Amazon EC2; and 3) transfer results from Amazon EC2 to the host computer running VMD for post-processing. Additionally, our Amazon EC2 Plugin has also integrated with the Interactive Molecular Dynamics (IMD) plugin [9], which can make a standard simulation interactive and display the simulation in real-time. Most importantly, it hides completely the details of any NAMD simulation deployment in the underlying HPC cloud. All of those tasks can be carried out by just a few button clicks.

This paper presents the key features of the Amazon EC2 Plugin that we have developed for VMD. The rest of this paper is organized as follows. Section 2 provides the background of molecular dynamics simulations using VMD and NAMD, and a brief description of Amazon EC2. Section 3 describes the design and implementation of the Amazon EC2 Plugin. A case study of molecular dynamics simulations of Satellite Tobacco Mosaic Virus on the Amazon EC2 via the support of our plugin is detailed and its benefits are discussed in section 4. Finally, conclusion and future work are presented in section 5.

2. Integration Components: VMD, NAMAD and Amazon EC2

VMD and NAMD are two major molecular dynamics software packages, which can work together for mining structural information of bio-molecules.

2.1. VMD

VMD is designed for modeling, visualization, and analysis of biological systems such as proteins, nucleic acids, etc [17]. In particular, VMD can act as a graphical front end for an external molecular dynamics software application such as NAMD by displaying and animating a molecule undergoing simulation on a remote computer.

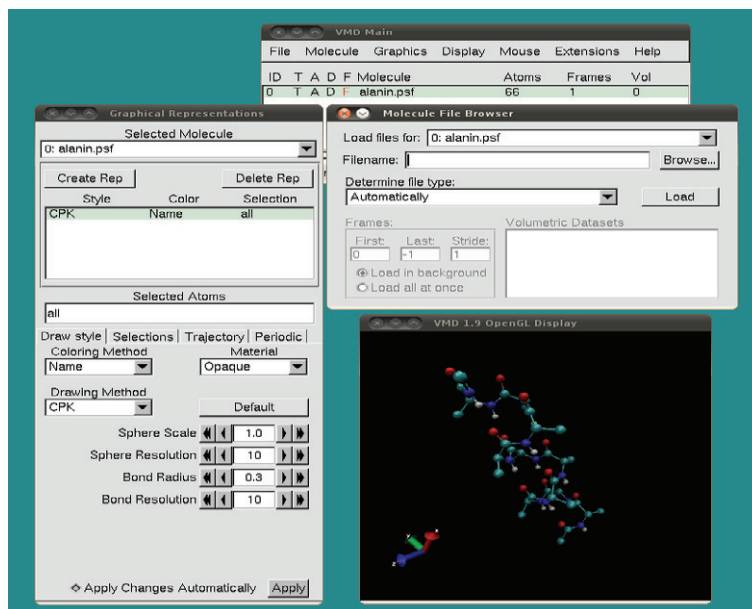


Fig. 1. A snapshot of VMD displaying a deca-alanine molecule

Fig. 1 shows VMD displaying the 3D structure of deca-alanine, a protein with 66 atoms. When VMD is started, two windows (the VMD Main and VMD OpenGL Display) are first popped up. The VMD Main window provides various functions, extensions and other interfaces to manipulate bio-molecular structures whereas the VMD OpenGL Display window is used to display the bio-molecule being manipulated. The first file needed to be loaded into VMD is the protein structure file, which contains important information about the bio-molecule system such as which atoms are bonded together, what charge they are, and the mass of each atom. The second file needed is the Protein Data Bank file, which contains the coordinates of the atoms. These files are loaded via the Molecule File Browser window, which can be popped up by selecting File -> Load Data Into Molecule function from the VMD Main window. To adjust the methods for rendering and coloring a bio-molecule, the Graphical Representations window is used, which can be popped up by selecting File -> Graphical Representations function from the VMD Main window.

2.2. NAMD

NAMD is a parallel, object-oriented molecular dynamics program designed for high performance simulation of large bio-molecular systems. NAMD employs the Charm++/Converse parallel runtime system [4] to handle inter-process communication among processes by default. It allows excellent parallel scaling on both massively parallel supercomputers and commodity workstation clusters. NAMD also supports the use of MPI [18] libraries in place of Charm++ for portability.

In order to run any NAMD simulation, a user needs to prepare the following:

1. Protein Data Bank (pdb) file - stores atomic coordinates and/or velocities for the system. pdb files may be generated by the user, but they are also available in the Protein Data Bank repository at [12] for many proteins;
2. Protein Structure File (psf) file - stores structural information of the protein such as various types of bonding interactions. This file must be generated by the user [3];
3. Force field parameter file - defines the force fields such as bond strengths, equilibrium lengths, etc. for atoms in the system. Parameter files for a given class of molecule could be obtained from CADD [5]; and
4. NAMD configuration (conf) file - specifies all the options that NAMD should adopt in running a simulation. This file must also be prepared by the user.

Among these four files, the NAMD configuration file is used as the input for a NAMD simulation. It specifies the search paths of pdb file, psf file, parameter file and other options, which define properties of the simulation. The

output of a NAMD simulation consists of two major files, one containing the final coordinates (coor) and another containing the final velocities (vel) of all atoms in the simulation. Additional output file such as the dcd file, which stores the trajectory of all atom position coordinates of an animation, can also be produced. Fig. 2 shows the execution flow of a NAMD simulation. Once a NAMD configuration file is prepared, the NAMD stimulation can be run on a HPC cluster by invoking the following command:

```
>charmrun ++remote-shell ssh ++odelist nodelistfile namd2 filename.conf ++p Num
```

where multiple namd2 instances are launched by the charmrun run-time. The option ++remote-shell specifies the remote shell to use in the cluster, option ++odelist specifies a list of compute nodes to use in the cluster, and option ++p specifies the number of namd processes to be launched.

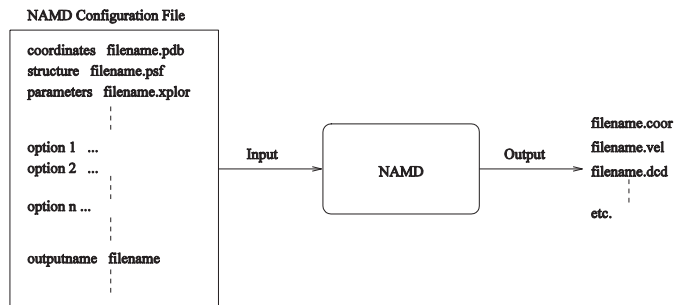


Fig. 2. A typical NAMD simulation

2.3. Amazon EC2

The simplest way to use the Amazon EC2 services is by accessing the AWS Management Console [3] with a web-based interface as shown in Fig. 3.

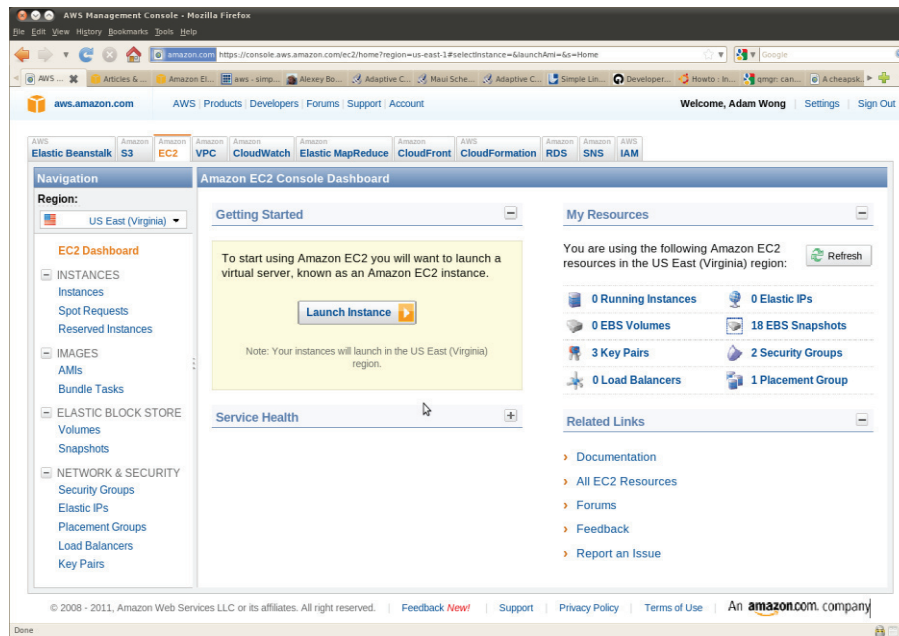


Fig. 3. A snapshot of the AWS Management Console

After logging on to the AWS Management Console, a user can get started with Amazon EC2 by performing the following tasks:

1. Launch instance (assuming Linux/Unix instance)
 - a. Choose an Amazon Machine Image (AMI), which contains all the information needed to create a new computer instance.
 - b. Create or select a key pair (a security credential), which will be used by the user to securely connect to the instance after it is running.
 - c. Create or select a security group, which defines firewall rules for the instance.
 - d. Launch the specified number of instances using the selected AMI.
2. Connect to instance
 - a. From a Linux/Unix host, use the ssh (secure shell) command to connect to the Linux/Unix instance.
 - b. Install software applications, if they are not already available, in the instance.
 - c. Transfer any files, which are required for the applications to the instance.
 - d. Run the applications.
 - e. Transfer any files back to the host if necessary.
3. Terminate instance
 - a. In the Management Console, locate the instance(s) in the list of instances on the Instances page.
 - b. Confirm to terminate the instance(s).

3. Amazon EC2 Plugin

As presented in Section 2, the current model of adopting a public HPC cloud service such as Amazon EC2 to carry out NAMD simulations is quite ad-hoc and could be tedious for researchers of molecular dynamics who have little background in HPC. On top of the work in launching, connecting and terminating AWS computer instances, they are also forced to deal with many details to set up and configure a HPC cluster; and install middleware and software applications before the system is available for any actual scientific investigation. For this reason, we have developed a software plugin for VMD, which can provide an integrated framework for VMD, NAMD and Amazon EC2 to work together for exploring structural information of bio-molecules.

3.1. Design

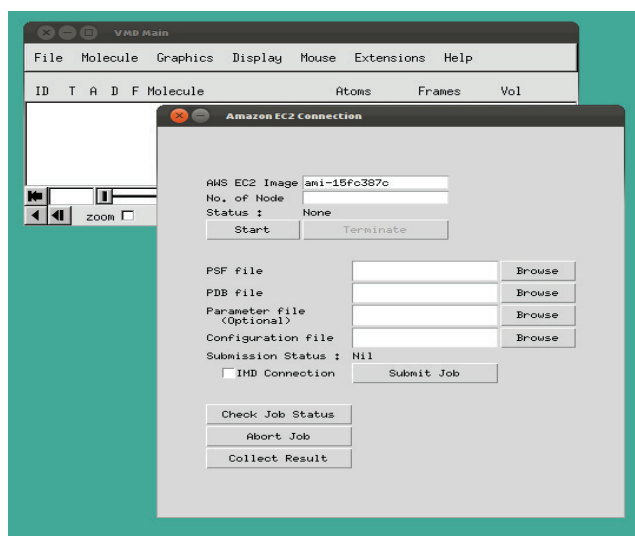


Fig. 4. A snapshot of the VMD Amazon Plugin

In VMD, users are free to dynamically extend its functionalities by providing new software plugins without the

need to recompile the entire program from scratch. The simplest type of plugins to be created for VMD is Tcl/Tk plugin [22]. Since Tcl is a scripting language, Tcl packages are relatively easy to develop, test and deploy. On the other hand, the Tk toolkit, a graphical user interface library, makes graphical application development easy. Our Amazon EC2 Plugin for VMD is implemented in Tcl/Tk as a graphical user interface (GUI) application, which provides various functions to automate the following actions:

1. Create instantly a HPC cluster on Amazon EC2.
2. Shutdown and terminate a HPC cluster on Amazon EC2.
3. Submit a NAMD parallel simulation from VMD at a host computer to Amazon EC2.
4. Integrate with the Interactive Molecular Dynamics (IMD) plugin, which can make a standard simulation running on Amazon EC2 become interactive and the simulation is displayed in real-time.
5. Collect result from Amazon EC2 to the host computer running VMD.

Fig. 4 shows the VMD Amazon Plugin window, which can be popped up by selecting Extensions ->Amazon EC2 Connection function from the VMD Main window. The first set of buttons handles HPC cluster creation and termination in Amazon EC2. The second set of buttons handles NAMD job submission and IMD connection to Amazon EC2. Finally, the third set of buttons handles NAMD results retrieval from Amazon EC2. Users do not need to access the AWS Management Console at all.

3.2. Implementation

As described in section 2.3, the AWS Management Console has provided users a point-and-click web interface for Amazon Web Services. Although the AWS Management Console can provide an interactive and convenient management of compute, storage, and other cloud resources to general users, it fails to support software developers to create applications which are built on AWS. Alternatively, there are other methods to access Amazon EC2 such as 1) coding directly to the web service API; 2) using a programming library for languages such as Java, PHP, Python, Ruby and Windows .Net; and 3) using the command line tools provided either by Amazon or other developers from AWS developer community. From among these three methods, we have chosen to use the last one to implement our Amazon EC2 Plugin as that is the simplest way to access Amazon EC2 services effectively from within any software program. Currently, we have provided its implementation in the Linux platform.

3.2.1. Amazon EC2 Accessing Scripts

To implement our Amazon EC2 Plugin for VMD, we have selected to use the Amazon EC2 command line tools developed by Timothy Kay [15] because it is simple to install and use. The only dependent tools used by the command line tools are Perl [19] and Curl [6] and they are normally included in many of the common Linux distributions. By using this command line tools, we have implemented a set of bash shell scripts, which are programmed to handle HPC cluster creation and termination in Amazon EC2; NAMD job submission and IMD connection to Amazon EC2; and NAMD results retrieval from Amazon EC2 as described in Section 3.1. Finally, a Tcl/Tk package was developed that implements the GUI-based plugin for VMD as well as the programming logic to invoke any one of the bash shell scripts in order to perform the Amazon EC2 handling tasks as specified in Section 3.1.

3.2.2. Providing public AMI that supports NAMD simulation

When a HPC cluster is to be created and started on Amazon EC2, users of our VMD Amazon Plugin must provide: 1) an Amazon Machine Image (AMI) for creating instance(s) and 2) a total number of instances for the HPC cluster. The former contains all information necessary to boot instances of software such as operating system, middleware and applications. The latter quantifies the size of the HPC cluster, which in turn defines the maximum computing power to be provided.

There is a large selection of public AMIs available from Amazon and the Amazon EC2 community. However in many cases, users may need to customize a public AMI and then save that customized AMI or even create new AMI from scratch for their own use. To help VMD/NAMD users with little computing and HPC knowledge, an AMI has been provided, which was constructed specifically for running NAMD simulations. The AMI runs Ubuntu Linux

and NAMD version 2.7. This AMI has been set publicly accessible from Amazon EC2. Users of our VMD Amazon Plugin can access this AMI by referencing its identifier, the AMI ID. This way, the construction of a HPC cluster and the installation and configuration of applications (NAMD in this case) has been abstracted into selecting an AMI from Amazon to use. In the current version of our plugin, this AMI is pre-coded in the bash shell script, `awsConnect`, of the Tcl/Tk package. We are planning to supply more AMIs in the future, which will define different machine images to support various NAMD versions including those with GPU acceleration.

3.2.3. Embedding the Amazon EC2 Plugin into VMD

To embed a plugin into VMD, a plugin directory is needed to store the Tcl/Tk package in a host computer, which runs VMD. Then, the plugin directory and the plugin name must be known by VMD through their settings in the VMD start-up `.vmdrc` file. The variable `auto_path` to be specified in `.vmdrc` sets a searchable path for the locations of plugin directories and variable `vmd_install_extension` to be specified in `.vmdrc` defines third party plugins in VMD. Finally, credentials of Amazon Web Services of a user account must be made accessible to the VMD Amazon Plugin in order to authenticate one's requests to any AWS service. Two types of access credentials must be provided in this case. The first one is the Access Keys pair, which is used to make secure REST or Query protocol requests to any AWS service API. The second one is the Amazon EC2 Key Pair, which is used to launch and then securely access Amazon EC2 instances. Details in setting up these credentials of AWS can be found in AWS webpage.

4. A Case Study of Molecular Dynamics Simulations

In this section, a case study of molecular dynamics simulations is used to demonstrate how accessibility and user-friendliness are improved by using the plugin. This case study also provides a performance evaluation of the molecular dynamics simulations in the HPC cluster created on Amazon EC2 and the total cost involved in carrying out such simulations.

4.1. Satellite Tobacco Mosaic Viruse

Satellite Tobacco Mosaic Virus (STMV) [14] is a small icosahedral plant virus, which lives on both a host cell and a host virus, the tobacco mosaic virus in this case, for it to reproduce. The entire STMV particle consists of 60 identical copies of a single protein that make up the viral capsid (coating), and a 1063 nucleotide single stranded RNA genome, which codes for the capsid and one other protein of unknown function. Although the full crystal structure of the STMV with both its capsid and RNA has been solved by Alexander McPherson [10] using X-ray crystallography and Atomic Force Microscopy (AFM) techniques and that has revealed a great deal of information about STMV, some unresolved questions remain regarding this virus. One of such is how does the crystal structure of STMA assemble and disassemble inside a host cell, because it has been impossible to obtain a high enough resolution electron map for that part of the structure. Therefore, STMV makes a good candidate for study using molecular dynamics simulations that can provide both a visual representation and quantitative data regarding the thermal motion of a system's structure over time, or its response to applied forces [7].

STMV was chosen for our NAMD simulations experiments because the fully solvated STMV system consists of approximately 1 million atoms, which is on the high end of what is feasible to simulate using molecular dynamics. This can demonstrate well for the needs of researchers to run their simulations on HPC hardware. The purpose of our experiment carried out however is not meant to provide any study on the virus itself.

4.2. Experiment

The experimental testbed used in this case study of molecular dynamics simulations is a computer cluster of 8 Cluster Compute Instances created in Amazon EC2. The computer cluster offers high computing power and low network latency features for optimal performance with HPC applications. Each Cluster Compute Instance has 23 GB

of memory, 33.5 EC2 Compute Units^b (2 x Intel Xeon X5570, quad-core “Nehalem” architecture), 1690GB of instance storage and a 10 Gigabit Ethernet network interface. Thus, this is a HPC cluster with 64 CPUs. Besides, a standalone workstation computer (Intel Xeon dual-core, 1.6GHz) with 4GB of memory is used to connect remotely to the cluster in Amazon so that a terminal can be created in the cluster to control the experiment.

The simulations source of STMV was obtained from [16]. It contains the structure file (stmv.psf), coordinates file (stmv.pdb), parameters file (parameter.inp) and configuration file (stmv.conf). A simulation workload of STMV with a simulation period of 30 picoseconds was created where all other simulation parameters were unchanged as set up in the original source. This workload was separately run on a standalone workstation computer and the HPC cluster with 64 CPUs. The job execution time was measured for the simulations in both cases.

4.3. Using the Amazon EC2 Plugin

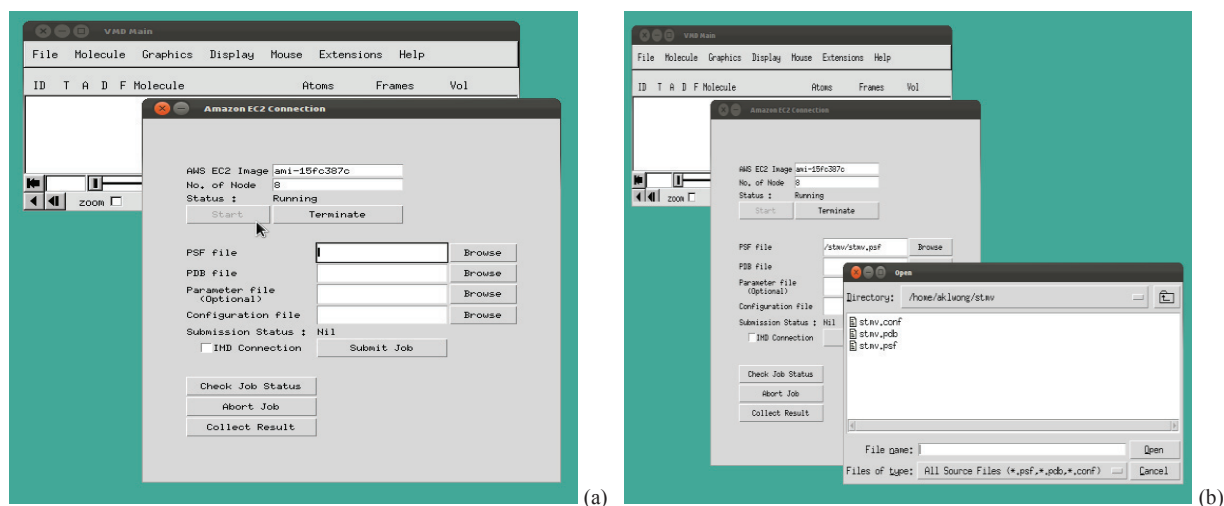


Fig. 5. Two snapshots of the VMD Amazon Plugin (a) showing 8 compute nodes have been started and (b) showing input files for a NAMD simulation are being submitted

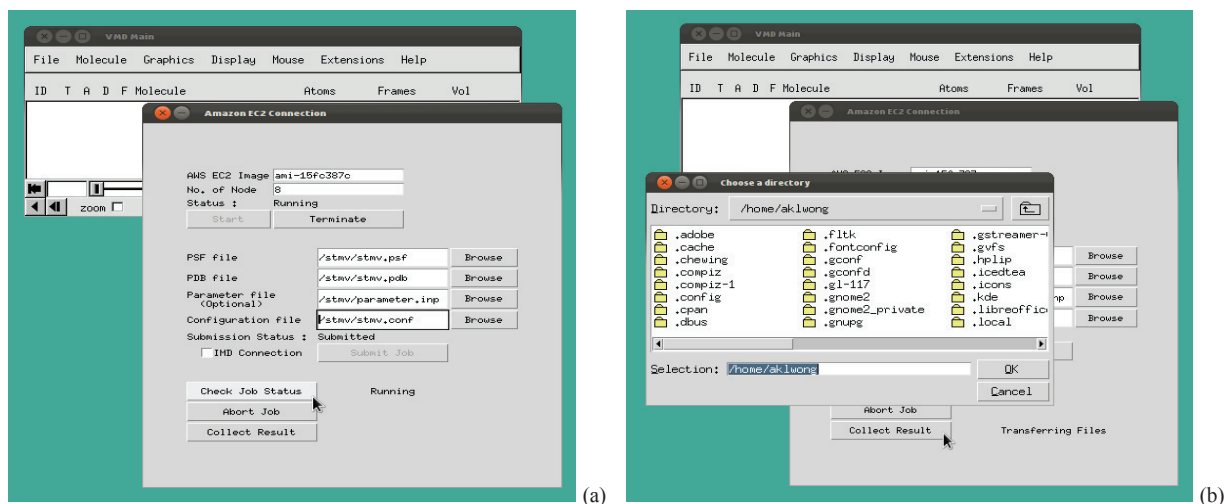


Fig. 6. Two snapshots of the VMD Amazon Plugin (a) showing a NAMD simulation has been submitted for execution and (b) showing a NAMD simulation has been finished and the result files are being collected

^b The Elastic Compute Unit (ECU) was introduced by Amazon EC2 as an abstraction of compute resources. One EC2 Compute Unit provides the equivalent CPU capacity of a 1.0-1.2 GHz Opteron or Xeon processor.

The Amazon EC2 Plugin for VMD relieves users from many tedious computing tasks such as launching, connecting and terminating Amazon EC2 compute instances; configuring a HPC cluster; and installing the NAMD software application before the system is readily available for running a NAMD simulation in the Amazon EC2. The simulation result can be transferred back to the user easily as well. All these tasks have been encapsulated into a few button clicks as illustrated in Fig. 5 and Fig. 6 rather than requiring users to perform those tasks manually as described in Section 2.3.

4.4. Result

Fig. 7 presents the result obtained from the experiment. The workload of NAMD simulations of STMV was 30 picoseconds. The execution time of the NAMD simulations obtained from running on a standalone workstation and the Amazon EC2 HPC cluster (all 64 CPUs) are 129.6 hours (5.4 days) and 4.6 hours correspondingly. The first scenario of using a standalone workstation for the NAMD simulations represented the case that no upfront investment in HPC hardware was affordable by the user. On the other hand, the second scenario of employing Amazon EC2 for the same NAMD simulations represented a contrasting case that a very flexible pay-as-you-go method of hiring HPC hardware could be afforded by the user. According to the latest pricing information that can be found in the Amazon EC2 website, the standard cost of a Cluster Compute Instance is \$1.60 per hour^c. The 30 picoseconds NAMD simulation has run on 8 Cluster Compute Instances (64-CPU) for 4.6 hours and thus has incurred a total cost of \$64.00.

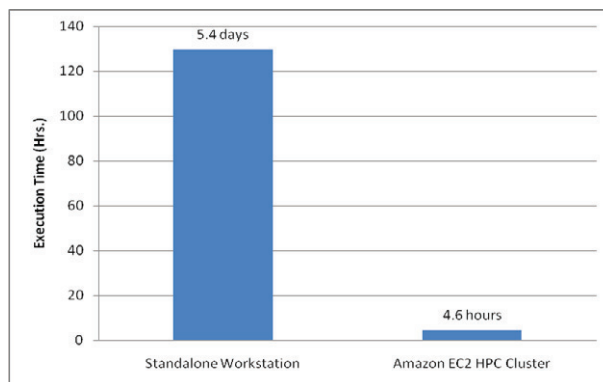


Fig. 7. STMV simulations (30ps) on a workstation and an Amazon EC2 HPC Cluster

5. Conclusion

We have designed and implemented a software plugin of VMD for NAMD simulations on the Amazon EC2. The plugin has provided an integrated platform for the molecular dynamics users of VMD/NAMD to harness a flexible pay-as-you-go method of purchasing HPC resource from the Amazon EC2. More importantly, it has released the users from performing many tedious computing tasks such as setting up and configuring a HPC cluster; and installing middleware and software applications before the system is readily available for any actual scientific investigation. As presented in Section 3.1, the graphical user interface of the plugin has abstracted many low level details in HPC cluster creation, NAMD job submission as well as result retrieval into just a few button clicks in the plugin. A proof of concept was a NAMD simulation case study, where the plugin allows VMD/NAMD users to spend less time getting applications to work on HPC clusters but more time on researching.

^c Pricing is per instance-hour consumed for each instance, from the time an instance is launched until it is terminated. Each partial instance-hour consumed will be billed as a full hour.

The NAMD simulations case study has also served to demonstrate a feasibility of running HPC applications on clouds. Our performance result obtained from running NAMD simulations on Amazon EC2 Cluster Compute Instances is in-line with other studies [8] [13], which has confirmed that a communication-bounded parallel application, like NAMD, can benefit from using Amazon EC2 HPC cluster. The flexible cost advantage of Amazon EC2 gives users the opportunity to test and run their high performance computing applications in the cloud at a price and performance level that can leverage their budgets and computation requirements.

Currently, we are improving the GUI of our Amazon EC2 Plugin by including a monitoring unit that can report on Amazon EC2 resources being used by users. The monitoring unit will provide information about a job's total execution time since it is started and an estimated total cost incurred so far for the job being executed.

Acknowledgements

This work was supported by the Amazon Web Services (AWS) in Education Research Grant.

References

1. Amazon Elastic Compute Cloud (Amazon EC2). <http://aws.amazon.com/ec2>. Last access: August 2011.
2. Amazon Web Service (AWS) Management Console. <http://aws.amazon.com/console>. Last access: August 2011.
3. M. Bhandarkar et al. "NAMD User Guide Version 2.8", Theoretical Biophysics Group University of Illinois and Beckman Institute. 2011.
4. Charm++. <http://charm.cs.uiuc.edu>. Last access: August 2011.
5. Computer-Aided Drug Design Center, School of Pharmacy, University of Maryland. http://mackerell.umaryland.edu/CADD/CADD_center.html. Last access: August 2011.
6. Curl. <http://curl.haxx.se>. Last access: August 2011.
7. P. L. Freddolino, A. S. Arkhipov, S. B. Larson, A. McPherson, and K. Schulten. "Molecular dynamics simulations of the complete satellite tobacco mosaic virus", *Structure*, 14:437-449, 2006.
8. A. Goscinski, M. Brock and P. Church. "High Performance Computing Clouds. Cloud computing: methodology, system, and applications", CRC, Taylor & Francis group. 2011.
9. Interactive Molecular Dynamics Simulation. <http://www.ks.uiuc.edu/Research/vmd/imd>. Last access: August 2011.
10. A. McPherson. Department of Biological Sciences, Molecular Biology and Biochemistry, University of California, Irvine. <http://darwin.bio.uci.edu/~faculty/mcpherson>. Last access: August 2011.
11. NAMD Molecular Dynamics Simulator. <http://www.ks.uiuc.edu/Research/namd>. Last access: August 2011.
12. Protein Data Bank. <http://www.pdb.org>. Last access: August 2011.
13. J. K.R. Ramakrishnan, L. M. K. Canon, S. Cholia, S. Shalf, H.J. Wasserman and N.J. Wright. "Performance Analysis of High Performance Computing Applications on the Amazon Web Services Cloud", In Proceedings of IEEE Second International Conference in Cloud Computing Technology and Science (CloudCom), pp. 159-68, 2010.
14. Satellite Tobacco Mosaic Virus. <http://www.ks.uiuc.edu/Research/STMV>. Last access: August 2011.
15. Simple Command-Line Access to Amazon EC2. <http://aws.amazon.com/developertools/739>. Last access: August 2011.
16. STMV simulation source. <http://www.ks.uiuc.edu/Research/namd/utilities>. Last access: August 2011.
17. Theoretical and Computational Biophysics Group at the Beckman Institute of the University of Illinois at Urbana-Champaign. <http://www.ks.uiuc.edu/Overview/People>. Last access: August 2011.
18. The MPI Forum. <http://www.mpi-forum.org>. Last access: August 2011.
19. The Perl Programming Language. <http://www.perl.org>. Last access: August 2011.
20. VMD Molecular Graphics Viewer. <http://www.ks.uiuc.edu/Research/vmd>. Last access: August 2011.
21. VMD Plugin Library. <http://www.ks.uiuc.edu/Research/vmd/plugins>. Last access: August 2011.
22. B. Welch, K. Jones and J. Hobbs. "Practical Programming in Tcl and Tk", Prentice Hall, Upper Saddle River, NJ, USA, 2003.