IaaS Clouds vs. Clusters for HPC: A Performance Study

Philip C. Church and Andrzej Goscinski

Deakin University, School of IT
75 Pigdons Road, Waurn Ponds, Victoria, 3216, Australia
Email: pcc@deakin.edu.au, andrzej.goscinski@deakin.edu.au

***Abstract*—The increasing amount of data collected in the fields of physics and bio-informatics allows researchers to build realistic, and therefore accurate, models/simulations and gain a deeper understanding of complex systems. This analysis is often at the cost of greatly increased processing requirements. Cloud computing, which provides on demand resources, can offset increased analysis requirements. While beneficial to researchers, adaption of clouds has been slow due to network and performance uncertainties. We compare the performance of cloud computers to clusters to make clear the advantages and limitations of clouds. Focus has been put on understanding how virtualization and the underlying network effects performance of High Performance Computing (HPC) applications. Collected results indicate that performance comparable to high performance clusters is achievable on cloud computers depending on the type of application run.**

Keywords – Cloud Computing, Benchmarking, Performance, System Biology, N-body simulation

I. INTRODUCTION

Cloud computing provides on demand computational resources of the Internet through use of virtualization, services and a pay-per-use paradigm. There has been interest in applying this computing technology to solve large scientific and industrial problems. By drawing resources from the cloud, even small research groups can solve these problems without investing in large amounts of computer infrastructure. However, cloud computing is still a developing technology and there have been many concerns about the overhead of virtualization and communication latency.

Virtual machines are used in Infrastructure as a Service (IaaS) clouds to provide users with dedicated systems which share underlying physical hardware. However there is a cost to create and maintain these isolated systems, a virtualization overhead, which is constantly subtracted from a user's allocated virtual resources [1]. Inconsistent network traffic flow also exists in clouds, which is problematic when running communication heavy applications [2]. It is in response to these issues, that some cloud providers have provided compute nodes which utilize hardware found in high performance computer clusters [3]. It is claimed that these High Performance Computing (HPC) enabled cloud nodes are optimized for running HPC applications yet it has not been proven in a practical manner.

This paper shows results of both the investigation of the feasibility of running HPC applications on clouds through benchmarking and the comparison of these results to cluster results. Two practical applications, an embarrassingly parallel bio-informatics visualization and communication bound N-body physics simulation, were chosen to represent classes of parallelization, data and functional parallelization. Using these applications HPC enabled clouds, standard IaaS clouds and a HPC cluster have been tested and compared. Of interest are the effects of virtualization and network latency, which have been documented to be the main performance issues [1][2].

The rest of this paper is as follows; Section II describes previous cloud benchmarks, their results and short fallings. Section III introduces the applications used during the benchmark; this is followed by a section introducing each computing platform and their specifications. Section V describes the methodology taken to setup each machine. Section VI presents performance results from the benchmarking, which is followed by a section investigating execution cost of the Amazon Elastic Compute Cloud (EC2) [13]. Finally, a conclusion and future work section is presented.

II. STATE OF THE ART

There are many advantages in using cloud computing for scientific research. For bio-informatics, running sequence alignment on the cloud (on a once per experiment basis) represents significant savings. Despite the increased range of cloud compatible bio-informatics software [4], adoption of on demand computing has been slow. Reasons for this slow adoption include usability and performance uncertainties [5].

A number of recent studies have investigated the performance of cloud computers. A solution by Napper and Bientinesi [6] runs LINPACK on Amazon Extra-Large instances (in both the Standard and High-CPU categories). Results indicate that these Amazon instances are not yet mature enough for HPC computations. Suggestions are made to offer better interconnects or nodes provisioned with more physical memory.

A study done by Indiana University measures the virtualization overhead of Xen and Eucalyptus through three practical applications (matrix multiplication, k-means clustering and the concurrent wave equation solver) Results showed a moderate-to-high virtualization overhead when running Message Passing Interface (MPI) applications [1].

 ISBN: 978-1-61208-153-3

A recent study by W. Guohui and T. S. E. Ng [2] investigated the network interconnect of EC2. An application called CPUtest was used to measure processor sharing, Round-trip Delay Time, Transmission Control Protocol (TCP)/User Datagram Protocol (UDP) throughput and packet loss. Observed results show abnormally large packet delay variations between cloud instances. Unstable TCP/UDP throughput was also seen, caused by end host virtualization.

The main criticisms of these studies were addressed by Amazon's recent addition of HPC cluster instances. These instances have 10 Gb Ethernet interconnect and more physical memory. Limited performance results exist for this machine, the most relevant is a LINPACK study run on a cluster of Amazon's EC2 Cluster Compute instances (consisting of 7040 cores). Results ranked the Amazon EC2 cluster 231 on the TOP500 super computer list [7].

As seen in the above examples, previous performance studies made use of scientific applications, profiling tools or LINPACK. Results from these studies indicate there are problems when running communication bound applications on the cloud. While the LINPACK result from the Amazon EC2 cluster instances indicates these problems are resolved, the EC2 HPC offering has not been studied through practical applications. In addition, cloud setup and cost of running scientific applications on the cloud has not been addressed.

It is because of these short fallings that a cloud benchmark is presented. Focus has been put on investigating the effects of network speed and virtualization on HPC optimized clouds. The financial cost of executing applications on the cloud is also examined. By basing this study on solving common scientific problems in bio-informatics and physics, a realistic case can be made for or against the use of cloud computing for scientific research. Comparisons are made between clouds and the currently used high performance clusters in order to quantify results.

III. APPLICATIONS

Scientific computing is a source of large scale problems. The amount of data collected in the fields of bio-informatics and physics has been exceptional, and data analysis can exceed the available computational time and storage. Cloud computing could be used to support large data analysis and solve large problems. A common application from each scientific field was chosen; in this way the measurements could be applicable to real life problems. This section describes the operation of applications used during the benchmarking study.

A. *Bio-informatics Application*

A patient's genome can be screened for cancers before any visible symptoms appear, and finding the inflicted subtype of cancer can lead to personalized cancer treatments. To facilitate these personalized treatments of cancer, signatures of cancer subtypes need to be collected. A common bio-informatics workflow used to find these subtypes involves building system models [8]. System models show the interaction of genes in a biological system, and are built by correlating genes together. Building a system model is an N × N problem, given a list of N genes; N correlations are required for each gene. This workflow consists of many steps including; normalization and filtering of data, statistically correlating genes and then visualizing these results in a network diagram.

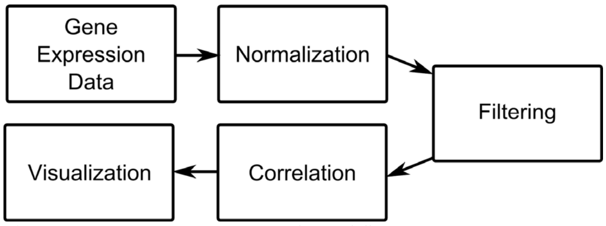

Fig. 1. A Common System Network Workflow.

The system network workflow presented in Fig. 1 makes use of data representing the amount of activated genes, also known as gene expression, in a biological sample. In order to find accurate relationships between genes, collecting both trait exhibiting and control expression datasets is necessary. Collecting this gene expression data involves multiple observations of genes in the biological system of interest. During this observation process human error can be introduced through uneven handling or scanning of samples. Normalization removes this bias by removing background noise from signal intensities and standardizing data so that distribution remains the same. Normalized data is then filtered, reducing the problem set by selecting genes that contain large variation. Correlation algorithms are then used to find the relationships between genes; commonly used correlation algorithms include Pearson's coefficient and Spearman's rho [9].

B. *Physics Application*

Data collected by particle accelerators such as synchrotrons and the Large Hadron Collider generate terabytes of data. By comparing simulations to collected results, it is possible to gain a better understanding of the laws that govern the universe [10]. We run a simulation of two disk galaxies colliding using an astrophysics application called GADGET [11]. This application is designed to simulate collision-less simulations and smoothed particle hydrodynamics on massively parallel computers. GADGET uses a combination of a physical mesh and tree based algorithms to simulate large range and small range particle interactions.

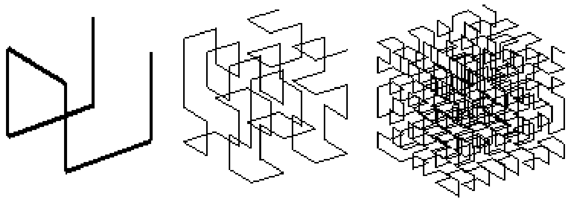

Fig. 2. 3D Representations of the Peano–Hilbert Curve.

Before each simulation step, physical mesh data decomposition is used to break the simulation area into

TABLE I
LIST OF BENCHMARKED COMPUTER PLATFORMS

Names	Nodes	Hypervisor	Platform	Hard Drive	CPU	RAM	Network	Interface
Amazon (Cluster)	8	Modified Xen: HVM	64-bit CentOS	Elastic Block Store	2 x Intel quad-core Nehalem (2.93 GHz)	23 GB	10Gb Ethernet	Web-based console. SSH.
Amazon (Large)	17	Modified Xen: Paravirtual	64-bit Ubuntu 9.10	Elastic Block Store	2 x Xeon equivalent (2.2 GHz)	7.5 GB	High I/O	Web-based console. SSH.
Amazon (Small)	17	Modified Xen: Paravirtual	64-bit Ubuntu 9.10	Elastic Block Store	2007 Xeon equivalent (1.6 GHz)	1.7 GB	Low I/O	Web-based console SSH.
vSphere Cloud	10	VMware	64-bit Ubuntu 9.10	Separate Drives	2.33 Ghz Intel Duel Core	2 GB	InfiniBand 10Gb	Web-based console, SSH. Remote Display.
InfiniBand Cluster	10	None	64-bit CentOS	Shared Drives	2.33 GHz Intel Quad Core Duo	8 GB	InfiniBand 10Gb	SSH.
HPCynergy	20	VMware	64-bit CentOS	Shared Drives	Virtual: Hexa-cores (2.33 GHz) Physical: Dual Quad Cores	8 GB	InfiniBand 10Gb	Web Interface. Web Service.

pieces. To achieve equal load balancing, GADGET makes use of the Peano–Hilbert curve to map 3D space onto a one dimensional curve. The Peano–Hilbert curve (see Fig. 2) is a space-filling curve variant which visits every point of a square grid. Once calculated, this curve is cut into pieces that define the individual domains. After the problem state has been reduced, it is distributed to multiple processors. Because this decomposition step occurs after every simulation step, load on processors are balanced.

In order to simulate the movement of galaxies the gravitational forces operating on close range particles need to be calculated. Calculation of force can be simplified by treating groups of similar particles as a single entity. In this way it is possible to summarize gravitational interactions between particles using a single force value. This force is calculated by adding together the mass of all particles in an area. While this method is quick, it is only accurate when particles are far away and will not work when particles are close. The accuracy of this method is improved through sub-division of the starting area.

IV. INTRODUCTION TO BENCHMARKED PLATFORMS

One physical machine and three cloud systems were used during this benchmarking. Naming conventions of the machines are as follows; the cluster is hereby referred to as the InfiniBand Cluster, while each cloud is referred to by the cloud management interface (vSphere [12], Amazon [13], HPCynergy [14] [15]). The vSphere and HPCynergy clouds are private clouds whereas Amazon is a public cloud. In terms of hardware, these computer platforms were chosen to be as similar as possible to each other, when possible utilizing the same pool of hardware. Of the four machines described below, HPCynergy, the vSphere and InfiniBand Cluster use the same hardware; the Amazon machines use their own individual hardware.

Despite the large effort taken to minimize hardware differences, some Amazon instances differ in the amount of cores per processor. Because of this variation, each process was mapped to a single core and when possible a single node. To validate the mapping process CPU usage was monitored during data collection, for example a duel core system with a single process would be using 50% capacity. This methodology was chosen as it is similar to that used by the cloud computers, in that virtual machines are mapped to physical hardware.

Three Amazon instance types [3] were tested; Small, Large and Cluster. It has been documented that Amazon uses a modified version Xen as the hypervisor. In each case the Amazon Elastic Block Store (an Amazon service which provides persistence storage of virtual hard-drive) was used to store the state of the deployed virtual machines. Amazon measures the performance of CPU's in Amazon Compute Units (ACUs); this is equivalent to an Intel Xeon chip. Each Amazon Small Compute instance contained 1 ACU and 1.7 GB RAM. Large instances contain four ACU and 7.5 GB of RAM. The Amazon Cluster Compute instances contain two Intel "Nehalem" quad-core CPU running at 2.98 GHz and 26 GB of RAM.

The second cloud used in this benchmarking was based on VMware virtualization technology. This private cloud made use of the same physical machines as the InfiniBand Cluster. A ten node virtual cluster was deployed through this VMware cloud, each with duel core processors running at 2.33 GHz. A 10 GB InfiniBand network was used to provide inter-node communication. VMware vSphere is used as the management software providing the ability to create, deploy and access virtual machines.

The third cloud used in this benchmarking was HPCynergy [14]. HPCynergy is a HPC cloud solution developed at Deakin University which incorporates a publishing service and broker. This cloud platform exposed VMware virtualized nodes running on the InfiniBand Cluster. A total of seventeen compute nodes were utilized through HPCynergy, each node containing a hexa-core processor running at 2.33 Ghz. A 10Gb InfiniBand network provided inter-node communication.

The InfiniBand Cluster used in this benchmarking is a bare-metal system consisting of 10 nodes each with an Intel Quad Core Duo processor running at 2.33 GHz. Each node utilizes 8 GB of RAM and runs a 64-bit version of CentOS to take advantage of this amount of RAM. As a machine dedicated to HPC, nodes are connected using 10 GB InfiniBand and a mounted network drive allows users to easily setup MPI applications. In terms of CPU speed and RAM size, this machine is equivalent to the documented specification of the Large Amazon instance. This machine differs from the Amazon instance having a faster network interconnect. Specifications of all platforms used in the following benchmarking are summarized in Table I.

V. Setting up the Cloud: Methodology

Setting up computer resources for High Performance Computing is both a time consuming task, and one that serves as an interruption to research. While the InfiniBand Cluster used in these benchmarking could be used once code had been compiled, the Amazon and vSphere clouds required modification to enable HPC. The HPCynergy cloud solution aims to reduce setup time by exposing systems which have middleware already setup.

Amazon and vSphere clouds required a number of steps including; transferring source code, configuring the compiler's dynamic linker, compiling the source code and any dependencies, configuring the sshd client, generating public and private keys, passing public keys to all nodes and creating a machineFile for MPI. The above steps were not required when setting up HPCynergy due to its unique interface. Like other clouds, HPCynergy monitors and acts as a broker to linked (physical and virtual) hardware. However instead of hiding the state and specification of hardware from the users, the opposite approach is taken. Users are informed of the software and underling (virtual) hardware specifications of each machine. This allows jobs to be optimized to the CPU architecture as well as minimizing the need to install specific libraries.

Some clouds had limitations which required additional setup time. The vSphere system did not contain any VM templates thus installation of the Ubuntu OS was required before operation. While all Amazon EC2 instances used in these benchmarks did not have common utilities such as the g++ compiler, the g77 compiler, vim or zip. Software compilation was more time consuming on the cloud systems. Missing library dependencies and compiler specific code meant that software would often fail during compilation.

Once each system was setup, input data and generated results had to be transferred from the user terminal to the cloud. Table II shows the total input/output transfer time and data size for each benchmarked system. For each benchmark, a total of 300 Mb was transferred between computers. Private clouds completed upload and download within seconds, however public Amazon clouds took many minutes. Results indicate that the time taken for data transfer is not just dependent on data size and network speed. Xen virtualization and differences in cloud interconnects can explain the variation between Amazon transfer times [3].

TABLE II
TOTAL DATA TRANSFER TIME

Computer Platforms	Input (Min)	Input Data Size	Output (Min)	Output Data Size
Amazon (Cluster)	3.8	85.2 Mb	4.3	231.2 Mb
Amazon (Large)	5.5	85.2 Mb	5.8	231.2 Mb
Amazon (Small)	6.8	85.2 Mb	23.8	231.2 Mb
vSphere Cloud	0.2	85.2 Mb	0.4	231.2 Mb
InfiniBand Cluster	0.2	85.2 Mb	0.4	231.2 Mb
HPCynergy	0.2	85.2 Mb	0.4	231.2 Mb

VI. Benchmarking

Comparisons made between collected results highlight the effects of virtualization and network latency of specific cloud platforms for high performance scientific computing. HPCynergy and Amazon's Cluster compute claim to address many of these weaknesses [3], [14]. HPCynergy is a software based solution while Amazon makes use of faster hardware. Benchmarking is used to prove that these HPC cloud platforms are feasible in regards to performance. To test performance, the system biology pipeline (Section III.A) and GADGET application (Section III.B) were run on a number of commercial cloud solutions, dedicated clusters, as well as virtual nodes discovered and used via HPCynergy. To ensure optimal performance, before analysis, input data was transferred to the local file system of each machine.

A. *Bio-informatics Benchmarking*

Performance of the system biology pipeline (described in Section III.A) was recorded from five machines, the Small and Large Amazon virtual clusters, the private vSphere cloud, the HPCynergy cloud and the InfiniBand Cluster. Results for each machine were measured up to four nodes; each test was run three times in order to ensure the validity of results.

As seen in Fig. 3, results show an almost linear increase of performance to available resources; this is expected as most of the system network workflow is embarrassingly parallel. When compared to physical hardware, the VMware based cloud shows a noticeable increase in required computational

 ISBN: 978-1-61208-153-3

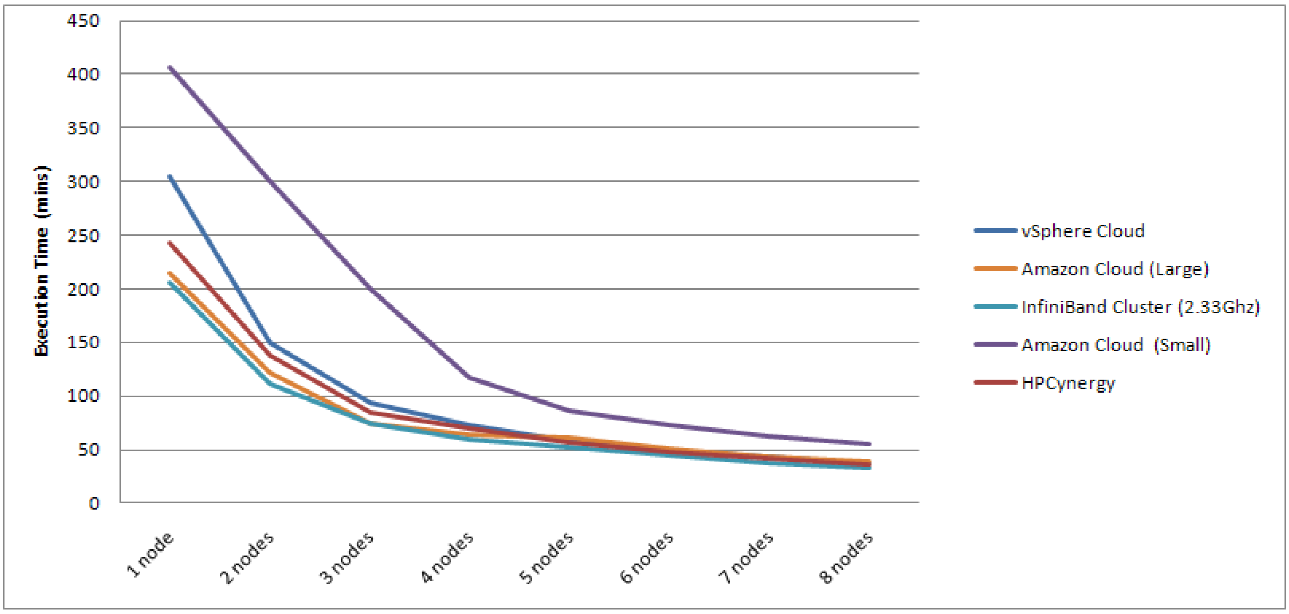

Fig. 3. IaaS Cloud Performance Comparison: Biological System Networks.

time. It is likely that this increase is due to virtualization overhead, in which part of the CPU is constantly being delegated to simulate the specified environment. Additional cloud services may also be responsible for decreased performance; this is seen in the HPCynergy platform which makes use of the same resource pool as the VMware cloud. When compared to the vSphere cloud, average performance is improved by 16%. The simple interface of HPCynergy allows for this improved performance, but it is not streamlined enough to match the physical hardware.

Additionally, collected results show an interesting relationship where the quicker a job runs the closer cloud performance matches physical hardware. This is due to virtualization overhead being distributed over many nodes. In the system biology pipeline, once job execution time became less than 35 minutes, virtualization overhead of clouds were indistinguishable from clusters.

In conclusion, different hypervisors and cloud service implementations have varying effects on performance. Amazon which uses a modified Xen hypervisor is very close to physical hardware, while the vSphere cloud which makes use of VMware virtualization suffered the most overhead. This virtualization overhead is minimised as jobs are spread across nodes.

B. *Physics Benchmarking*

The Small, Large and Cluster Amazon EC2 clouds, the private vSphere and HPCynergy clouds and an InfiniBand Cluster (see Section IV for extended specification details) were also utilized for the physics benchmarking. Benchmarking made use of full machine capacity, tests running up to 17 nodes. Each point was run three times in order to ensure the validity of results.

The results from this benchmarking can be seen in Fig. 4. As seen in the physical hardware results, the ideal performance of this GADGET benchmarking is a constant decrease as more compute nodes are added. The vSphere cloud, which runs on the same hardware, shows this shape with a similar offset seen in the bio-informatics study (Section A). Despite utilizing the same pool of resources and hypervisor, the HPCynergy solution sees an average performance improvement of 16% compared to the vSphere cloud. It is this simple interface of HPCynergy that allows for the improved performance results, but it is not streamlined enough to match the performance of the physical hardware.

Performance of the Amazon EC2 cloud varies depending on the instance type chosen. Performance of the Amazon Small instance shows a sharp computational increase at 2 nodes before performance becomes optimal at 3 nodes. The Amazon Large instance with higher I/O shows a similar early computational spike before optimizing at 5 nodes. Both the Small and Large Amazon EC2 cloud instances show an increase in computation time as more nodes are added past this optimal performance threshold. This relationship is an indication of a communication bottleneck, where each node is spending more time communicating then processing. Amazon's recently added Cluster Compute instance [13] has been optimized for running computation heavy applications. The performance of this instance shows a decrease in execution time mirroring other high speed clusters. This optimal performance is only guaranteed when allocating cluster instances at the same time. Because of this requirement the user loses one of the biggest draws to the cloud, the ability to elastically scale their applications.

Unlike the system biology problem presented in Section III.A, this N-body algorithm requires communication between nodes. Collected results from Amazon show that performance is not necessarily linked to amount of machines used. When running communication based applications, it is important that load is balanced between nodes and that

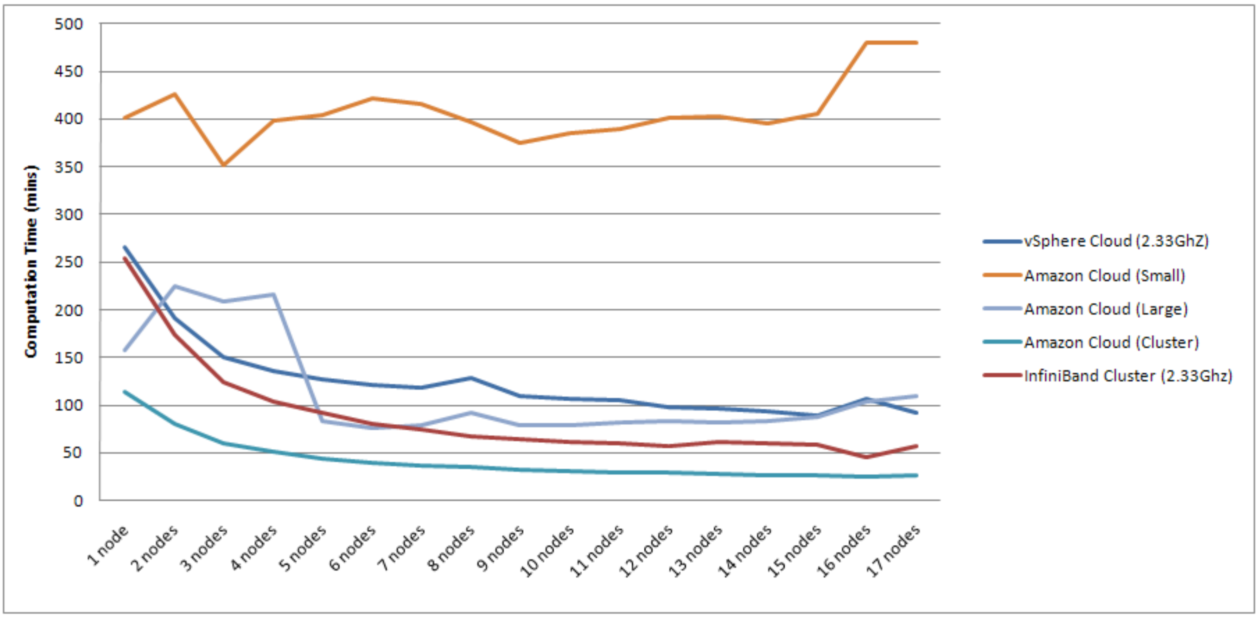

Fig. 4. IaaS Cloud Performance Comparison: N-body Simulations.

communication is minimized. If each node is communicating more than it is processing, the computation time will increase as resources are added. Cloud computers resources are highly distributed and performance of communication heavy applications can vary depending on the network architecture and the location of machines that have been allocated to the user.

VII. Cost Investigation

One of the big draws to the cloud is hardware scalability. Running a single machine for 5 hours costs the same as running 5 machines for 1 hour. Theoretically, this means the cost running an application should be the same regardless of time. This however may not be the case. Fig. 5 presents the cost per execution time of the Amazon instances run during the benchmark.

In terms of cost, the embarrassingly parallel bio-informatics application was the most efficient. While originally under-performing, the expected cost stabilization does occur in both the Small and Large Amazon instances at 5 nodes. Results from the physics benchmark did not show this trend. Running GADGET on the Small Amazon instance was wasteful, performance decreasing with each dollar spent. The large and cluster instances showed performance improvements with cost, the cluster instance scaling more consistently.

In conclusion, embarrassingly parallel applications are well suited to the pay on demand cloud model. Results show that execution time can decrease while maintaining the same total cost. Communication bound applications are not as cost efficient. Collected results show inconsistent performance per node and inconsistent cost-performance ratios. The main problem when utilizing the cloud for communication bound HPC applications is this performance unpredictability. Even the cluster instance, which showed the most consistent improvements did not show any hint of eventual cost stabilization.

VIII. Conclusion And Future Work

The results presented in this paper show that even standard public and even more private clouds can achieve performance similar to that of dedicated HPC clusters depending on the class of problem. When running embarrassingly parallel applications a near linear speed up is achievable and the results are comparable to those achieved on a cluster.

Clearly the effects of virtualization vary with the type of hypervisor used; Xen seems to have minimal performance effect on computation while VMware is noticeable. When running communication bound applications performance results vary. On the clouds with slow network speeds the N-body application achieved maximum performance at 5 nodes and then required compute time steadily increased due to communication overhead. The two clouds with HPC hardware (Amazon Cluster Compute instance, HPCynergy and VMware) showed the same decreasing performance trend as the InfiniBand Cluster. These performance results indicate that communication bound applications should be run only on clouds which provide high speed interconnect.

While some performance issues have been resolved, cloud setup is difficult and time consuming. A user must construct a virtual cluster and install analysis software. This setup process often starts through modifying of a pre-existing template. Templates can be difficult to utilize as they are often not documented, missing common dependencies (compilers, text editors, etc.) and may have a range of security access setups.

Benchmarking showed that transferring data to public clouds was a major issue. Compared to local clouds and clusters, public clouds increased data transfer requirements

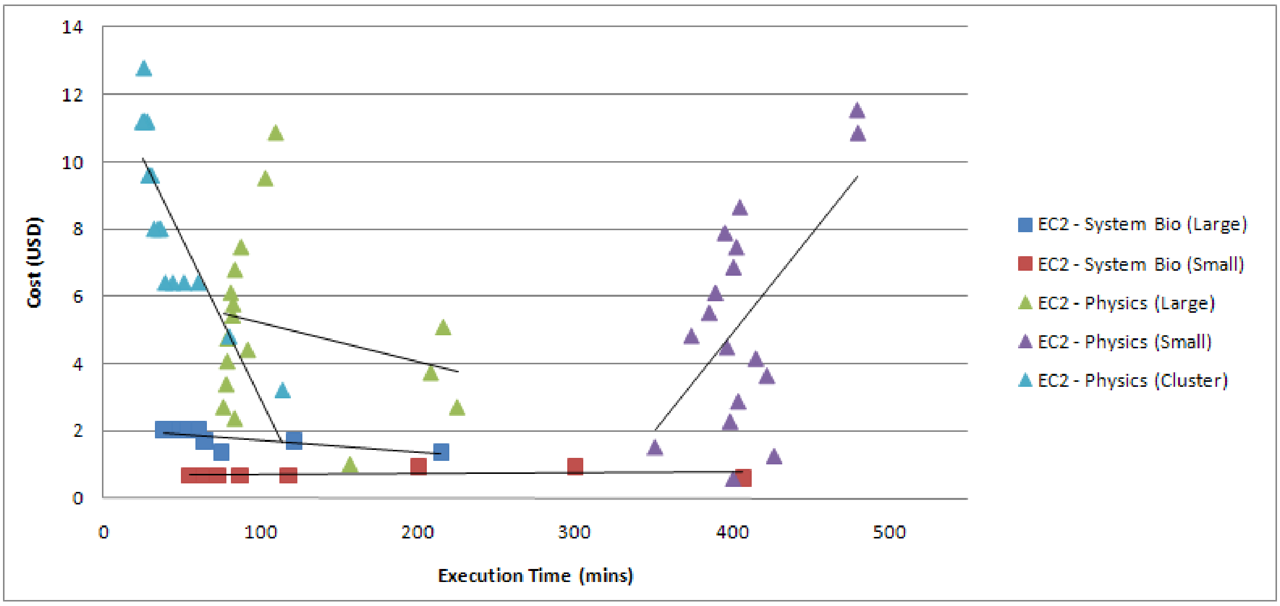

Fig. 5. Comparison of cost and execution time of the EC2 cloud.

by a factor of 25. This is problematic as the scientific applications described in this paper can make use and generate gigabytes of experimental data. At first glance the large transfer times are merely an artefact of the physical distance between cloud storage and user terminal. However collected data transfer results show significant variation between Amazon Cloud instances. This indicates that differences in cloud interconnects is also a concern, cloud storage and cloud instances often being separated. It is hoped that the adoption of faster broadband technologies should remove much of this data transfer delay.

Future work is planned to investigate the performance of clouds when running a wide range of applications. Of interest are other bio-informatics applications including; protein simulation and sequence alignment. With increased data, these applications will have profound effects in the fields of medicine and drug discovery.

It is also important to devise algorithms that take advantage of the cloud platform. To obtain maximum benefit from clouds, these algorithms must scale to large amounts of data and compute nodes while integrating solutions to minimise data transfer. It is possible to reduce the amount of input data by devising analysis methods which use compressed data. Another possibility is to devise cloud workflows which utilize the power of the user's desktop computer to perform data filtering and pre-processing. Currently we are investigating ways to stream data to the cloud; this allows faster processing turn-around (by minimizing idle compute time).

ACKNOWLEDGMENT

This work was supported in part by an Amazon Web Service Research Grant and the "Innovations through Broadband" FRC through the Deakin University, Faculty of Science and Technology contact.

REFERENCES

[1] D. R. Avresky, *et al.*, "High Performance Parallel Computing with Clouds and Cloud Technologies," in *Cloud Computing*. vol. 34, O. Akan, *et al.*, Eds., ed: Springer Berlin Heidelberg, 2010, pp. 20-38.
[2] W. Guohui, and T. S. E. Ng, "*The Impact of Virtualization on Network Performance of Amazon EC2 Data Center."* pp. 1-9.
[3] Amazon (2010) *Amazon EC2 Instance Types*. Accessed 24 September 2010, http://aws.amazon.com/ec2/instance-types/
[4] B. Langmead, *et al.*, "Cloud-scale RNA-sequencing differential expression analysis with Myrna," *Genome Biology,* vol. 11, p. R83, 2010.
[5] H.-L. Truong and S. Dustdar, "Cloud computing for small research groups in computational science and engineering: current status and outlook," *Computing,* vol. 91, pp. 75-91, 2011.
[6] Jeffery Napper and Paolo Bientinesi, *"Can Cloud Computing Reach the TOP500?"* Proceedings of the combined workshops on UnConventional high performance computing workshop plus memory access workshop (2009).
[7] Top500 (11/2010) *Amazon EC2 Cluster instances - TOP500*. Accessed 15 June 2010, http://www.top500.org/system/details/10661
[8] Khalil, I., Brewer, M.A., Neyarapally, T. and Runowicz, C.D. (2010) ***"****The potential of biologic network models in understanding the etiopathogenesis of ovarian cancer.**"* Gynecol Oncol. **116**(2):282-5
[9] J. L. Rodgers and W. A. Nicewander. *Thirteen ways to look at the correlation coefficient.* The American Statistician, **42**(1):59–66, February 1988.
[10] J S Bagla and T Padmanabhan (2008), *'Cosmological N-body simulations'*, Pramana, 49 (2), 161-192.
[11] Springel V (2005), *'The cosmological simulation code GADGET-2'*, MNRAS, submitted, astro-ph/0505010
[12] VMware (2010) *VMware vSphere 4: Private Cloud Computing, Server and Data Center Virtualization.* Accessed 15 January 2011 http://www.vmware.com/products/vsphere/
[13] Amazon (2010) *Amazon Elastic Compute Cloud*. Accessed 24 September 2010, http://aws.amazon.com/ec2/
[14] A. Goscinski and M. Brock. Toward dynamic and attributed-based publication, discovery and selection for cloud computing. Future Generation Computer Systems V. 26, I. 7, 2010.
[15] Andrzej Goscinski, Michael Brock and Philip Church. *"HIGH PERFORMANCE COMPUTING CLOUDS",* Cloud computing: methodology, system, and applications (2011). CRC, Taylor & Francis group.