ASEE's 123rd Annual • Conference & Exposition • New Orleans, LA • June 26-29, 2016

New Orleans JAZZED about Engineering Education

Paper ID #16060

HPC as a Service in Education

Dr. Hassan Rajaei, Bowling Green State University

Hassan Rajaei is a professor of computer science at Bowling Green State University, Ohio. His research interests include cloud computing, High Performance Computing (HPC), distributed simulation, parallel and distributed processing, communication networks, wireless communications, and virtual training environments. Rajaei received his Ph.D. from Royal Institute of Technology, KTH, Stockholm, Sweden, and he holds a M.S.E.E. from the University of Utah, and a BS from University of Tehran.

Mr. Saba Jamalian, Bowling Green State University

HPC as a Service in Education

Abstract

Advances in Cloud Computing have opened many chapters in Information Technology. Numerous service platforms offer clients of the cloud ease of use and flexibility of using the provided services. Education with billions of potential users worldwide is a major target. An emerging service called HPC-as-a-Service (HPCaaS) targets Science, Technology, Engineering, and Math (STEM) users. In this paper we discuss the use of HPCaaS platform in STEM education. We argue that such a service can significantly alleviate a major obstacle in teaching parallel programming for the STEM students.

Cloud computing provides unique benefits such as resource pooling, cost efficiency, availability, and large computational power. These features have attracted scientists, engineers, scholars, and the High Performance Computing (HPC) users like a magnet towards the Cloud. However, HPC programs often consume large number of collaborating processors to reduce the execution time, where synchronization between these processors and the communication overhead among them can become real challenge even on dedicated and special hardware, but worse on shared and virtualized platform like cloud. As a result, moving HPC applications to the cloud can adversely impact the abovenamed difficulties with additional issues, primarily virtualization, multi-tenancy, and network latency. One solution is a new cloud service known as HPC-as-a-Service.

In this paper, we present an HPCaaS platform called ASETS which uses Software Defined Networking (SDN) technologies to smooth the execution of parallel tasks in the cloud. Further, we provide application examples that could be used in a typical introductory parallel programing course. We argue that HPCaaS platform like ASETS can significantly benefit the users of HPC in the cloud as if their program is running on a dedicated hardware in their own laboratory. This is especially advantageous for students and educators who need not to deal with the underlying complexities of the cloud.

1. Introduction

Cloud Computing according to NIST[1] is a shared pool of configurable resources offering services with five essential characteristics; on-demand self-services, broad network access, resource pooling, measured service, and rapid elasticity. A sixth characteristic that is often not considered essential but rather as a side-effect of resource pooling regards multi-tenancy. Clouds can be deployed as private within an organization, public for general users, or hybrid of the two. Cloud is a service oriented architecture with three well-known service models: software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS) [1, 2]. Recently "as-a-Service" has become a trend where a particular service platform is prefixed to the term to provide additional services to the user community. The main idea is to use the flexibility of the cloud and its benefits. Furthermore, clouds maintain the software and applications up-to-date, providing a wide variety of choices and various platforms.

High Performance Computing (HPC) application domains cover Scientific & Engineering complex problems, to Defense, Finance, Pharmaceutical, and Big Data, just to name a few. This is a fast growing field, where numerous users desire move to the cloud. Nevertheless, HPC traditionally utilizes proprietary hardware with batch processing. HPC applications often run with direct access to the physical hardware (i.e. multi-core machines, grid, and supercomputers) that provides best achievable performance. As a result, the traditional HPC platforms are extremely fine-tuned to receive best performance of parallel processing. Moving HPC applications directly to the cloud can face performance penalty and thus extinguish the advantage gained from the cloud services [3, 4].

In this paper, we describe the benefits offered by the cloud for HPC applications and argue why HPC-as-a-Service (HPCaaS) would be a better choice. We compare different methodologies that have been used to better utilize the cloud infrastructure for HPC. Further, we identify several challenges facing HPC applications on the cloud, and provide potential solutions for them.

Cloud in educations enables the HPC students to have access to supercomputing resources on demand and in a cost efficient manner. The providers of HPCaaS, often own the service platform, administrate, and maintain the virtual resources. The platform providers may own the hardware or rent it from a cloud service provider. Nevertheless, the platform can have updated resources with the latest technology while the students and educators can benefit from the service and thus need not be concerned about the underlying mechanisms and complexities.

In addition, we present an HPCaaS platform called ASETS[5], which uses Software Defined Networking (SDN) technologies [6] to smooth the execution of parallel tasks in the cloud. Further, we provide application examples that could be used in a typical introductory parallel programing course. We argue that HPCaaS platform like ASETS can significantly benefit HPC users on the cloud. This is especially true for students and educators which do not need to deal with the underlying complexities of the cloud or the multiprocessors system they are using. Our solution makes it possible for the HPC students to enjoy using the cloud services and yet gain good performance on their parallel programing tasks. In addition, we provide execution performance of the given examples both on a private cloud as well as executing them on Amazon public cloud. We conclude that an HPCaaS platform can greatly facilitate learning parallel programing by the STEM students, especially in institutions where access to an on-site high performance computing facilities or a private cloud is either nonexistent or limited.

The rest of this paper is organized as follow: Section 2 presents related work, and Section 3 reviews cloud fundamentals whereas Section 4 describes HPC as a Service. Section 5 overviews parallel programing supported by HPCaaS platform. Section 6 describes ASETS, our solution to HPCaaS and Section 7 gives multiple HPC stimulating topics where the described service can be used. Concluding remarks and future work is provided in Section 8.

2. Related Work

Numerous work targeted HPCaaS in general but not focusing in its use in education in particular. While such a need in education is highly desirable, the literature review is not observing the demand and our work seems pioneering to address this necessity. Thus in this section, we review some work on general needs for HPCaaS and conclude that even these work could be expanded for educational purposes as well.

In one of the very early attempts of porting HPC applications to the cloud, Evangelino et al [7] evaluated Amazon EC2 for HPC purposes by using MPI benchmarks as well as a real-world scientific application (atmosphere-ocean simulation). They showed that cloud could not compete with supercomputers in terms of performance, though the results were encouraging at smaller scales comparing with low cost clusters. Furthermore, they concluded that interconnects and networking of the cloud are the major bottlenecks for the performance of HPC applications.

Gupta et al[8] recognized poor network performance, performance variation, and Operating System noise as the primary shortcomings of executing HPC applications in the cloud. In one study[9], they proposed an application-aware job scheduling system that optimizes HPC applications for the Cloud to improve the performance in terms of average turnaround time and throughput. In another study[10] they proposed a Virtual Machine (VM) placement strategy for Cloud systems to make the Cloud more compatible with HPC jobs. Their efforts indicate that Cloud is a viable platform for HPC applications, in particular for ones that are embarrassingly parallel and not communication-intensive.

AbdelBaky et al[11] explored a hybrid approach to dynamically scale resources of a supercomputer to the Cloud for a typical HPC application. The performance in the result system was neither reduced nor improved but the provided abstraction layer enabled HPC as a Service in the Cloud as a simpler yet more powerful alternative for HPC users and developers.

The Cloud platform in general has recently been utilized as a great tool to accelerate and enhance educational experiences. For instance, Alshuwaier et al[12] explored the applications of Cloud Computing in education and proposed a set of example use cases of cloud-based platforms for academic and educational purposes. Moreover, Marinela et al. in another study [13] argued that the Cloud seems to be the most efficient solution as an educational platform in terms of cost, particularly in the times of financial crisis. This paper, nevertheless, targets the application of Cloud for HPC education in form of the proposed HPCaaS.

3. Cloud Fundamentals

NIST and other organizations define Cloud Computing as Internet-based technology[2, 14, 15] which offers computational resources via a computer network and delivers flexible, scalable, and on-demand services to the end-users. It furnishes large storage capacity, network bandwidth, and vast processing powers[16, 17]. The technology aims at freeing the users from being attached to specific machines. Cloud Service Providers (CSPs) assure performance of applications through Service Level Agreement[18] (SLA). The services are classified into three basic layers—*applications*, *platforms*, and *infrastructures*—each serving different purposes and tasks[19].

The first layer is referred to *Infrastructure as a Service (IaaS)*. This layer can accommodate a complete IT infrastructure and deliver *Virtual Data Center as a Service (VaaS)*. At this level, CSPs often offer different services, for example, computing power represented by Virtual Machines (VMs), storages, virtual servers, networking services and more. The users can obtain the needed resources delivered to them via CSPs' portals. For example, Amazon Elastic Compute Cloud (Amazon EC2)[21] provides computing capacity via Amazon's data centers. IaaS offers a scalable infrastructure with the rapid provisioning feature. In addition, the users pay only for the utilized resources for the time period they needed eliminating overspending capital and initial setups. By integrating different cloud services to offer additional resources, institutions may no longer be worried about resource limitations. This characteristic can play a key role in Cloud-Based-Education (CBE) as virtual laboratories can be setup when they are needed and torn down when they no longer used.

The second layer is referred to *Platform as a Service (PaaS)*, which provides the ability to develop and deploy applications. CSPs often offer set of tools and services to help the developers to build their applications. Further, they can use plug-ins or prebuilt sub-programs for any Web-based applications which are compatible with the CSP's applications. PaaS offers services in different abstractions. The clients can use existing sets of deployed applications considered as high-level services, or build platforms using the provided operating systems and middleware.

The third layer is on the top of PaaS and is referred to *Software as a Service (SaaS)*. It aimed to deliver easy access to cloud applications through the Internet. In this layer, a CSP can provide a single instance on the Cloud for multiple users. Google Apps, a popular SaaS domain[22], provides variety of Web-Based applications for business, education, and government.

Based on the description above, an HPCaaS platform can be built on the second layer of the cloud, and on top of the IaaS fabric layer. The HPC platform layer in turn, will provide its services to the HPC applications on the SaaS layer. These services can be in form of Application Program Interfaces (API) like Google Apps, or better yet, in form of building blocks of HPC program examples, where students can use them as is, or expand them as they wish. It is here, that numerous sandboxes can be built for HPC STEM students to learn the objectives of the learning outcomes of a typical course in data science programming.

4. HPC as a Service

Cloud intends to provide utility computing by providing several resources as a service. The Service that support moving HPC applications to the cloud and utilizing cloud infrastructures without significant loss, and minimal interference by the user is called HPCaaS. Primary motivations for moving HPC applications to the cloud are in line with the typical benefits of Cloud Computing such as resource utilization, cost efficiency, flexibility, etc. Table 1 summarizes the comparison of the characteristics of an average on-premises HPC resource such as clusters or supercomputers with HPC provided by a public cloud (HPCaaS).

Table 1: On-Premises HPC resources vs. HPCaaS

Characteristic	On-premises HPC resource	HPCaaS
Scalability	Low	High
Flexibility	Low	High
Reliability	Low	High
Cost	CAPEX	OPEX
Setup Time	High	Low
Maintenance & Administration	expensive & complex	N/A
Resource Utilization	Low	High

4.1. Benefits of HPCaaS

In addition to general benefits of cloud computing, cloud is able to provide several unique advantages for the HPC community that is discussed in this section.

Ease of Maintenance and Administration

Majority of HPC users in the STEM related fields are not computer scientist, but rather engineers, biologists, physicists, or other academic scholars without in-depth computing background. They prefer not to get involved in the hassles of running and maintaining an HPC clusters or supercomputers. Nevertheless, currently most on-campus HPC users are forced to need a server administrator to maintain the "on-premises" hardware and software resources. In contrast, a cloud service provider preserves the infrastructure up-dated for running the applications keeping users away from the troubles of maintaining the resources [8]. Galaxy [23] is a good example of a cloud-based tool that provides a layer of transparency for computational research in the life sciences. The interactive web based tool makes it easy for biologists to use predefined software libraries and tools for their data intensive biomedical research. One of the very popular life science algorithm commonly used in Bioinformatics is BLAST. Wei Lu et al [24] utilized Microsoft Azure cloud infrastructure to implement and evaluate the performance of BLAST on the cloud. They proposed a cloud based architecture for BLAST that can be generalized to any cloud based infrastructure as well.

Efficiency in Resource Utilization

Cloud computing in general provides a dynamic and scalable infrastructure for business and web applications. In other words, based on the users and application demand, resources on the cloud can be dynamically scaled up or down. When it comes to HPC users in an academic setting, the variety of such spikes in the demand will increase. According to the timing of the scientific experiments, HPC users may not need the resources for a relatively long period of time or they may have a huge sudden demand for the resources time to time. As a result, utilizing cloud based resources and in particular benefiting from the pay-per-use model, will turn the capital expenses to operational expenses (CAPEX to OPEX)[25]. Cycle Computing is another example of transparent HPC on the cloud. They aim to provide utility supercomputing transparently on top of Amazon Infrastructure with an easy to use web based console. It makes it possible for scientists and HPC users to rent HPC resources worth of millions of dollars for a price of hundreds of dollars per hour[26].

Performance to Cost Ratio

A cloud-based virtual cluster often shows a performance penalty in comparison with a physical cluster with the same configuration. This penalty is due to several reasons where some are discussed in Section 4.2 below. However, it is a different story when it comes to the ratio of performance to cost [27]. The pay-as-you-go model and the scalability of the resources on the cloud can reduce the cost for running those of HPC applications with low communication overhead and I/O [8, 28].

4.2. Challenges Facing HPC Applications on the Cloud

Several researches and studies compared the performance of HPC applications on the cloud with on-premises infrastructure and reached to this conclusion that Clouds of today cannot compete with supercomputers[29, 30, 31, 32]. This section briefly describes primary reasons for such shortcomings.

Multi-tenancy

Multi-tenancy is one of the characteristics of the cloud. It is also one of the profit making features of the cloud for cloud providers. It enables cloud providers to share resources between multiple tenants. Degree of multi-tenancy refers to the number of tenants sharing a same resource on the cloud. By increasing the degree of multi-tenancy, cloud providers are able to overprovision the resources to users. Overprovisioning allows cloud providers to maximize benefits, though with the risk of reducing QoS[33]. Nevertheless, multi-tenancy is in direct contrast with what HPC needs. HPC applications demand direct access to dedicated hardware using some sort of batch scheduling while shared resources on the cloud complicates the performance of HPC applications[34].

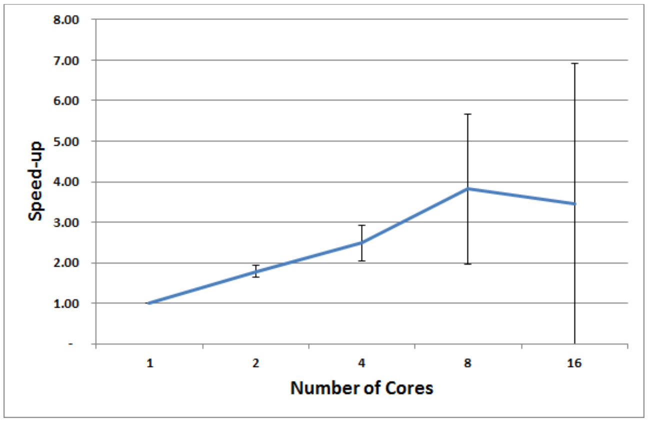

Figure 1: Speed-up for Matrix Multiplication benchmark running on Amazon EC2 instance

We conducted an experiment by running a Matrix Multiplication benchmark on a virtual instance of Amazon EC2 public cloud. We repeated the experiment for 15 times and Figure 1 shows the result for the achieved speed-up. The error bars represent the standard deviation of the results and indicate that by increasing the number of cores, the diversity of values we get in multiple experiments, increases. Figure 2 is the efficiency achieved for the experiments and the error bars are again the standard deviations. This experiment shows how performance of HPC applications on the cloud is not predictable due to the shared resource and multi-tenant environment of the cloud. Moreover, the cloud lowers the efficiency

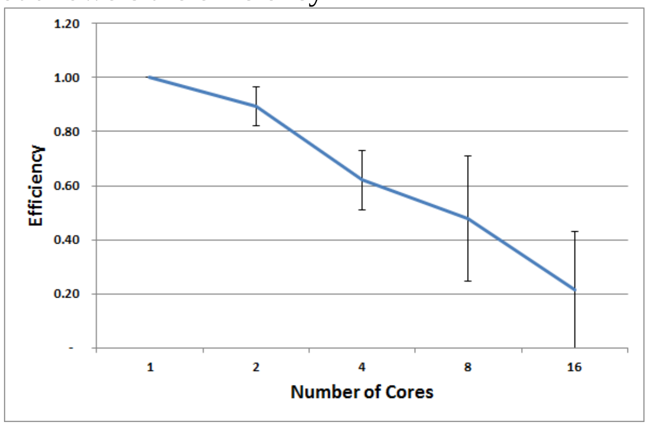

Figure 2: The efficiency achieved by running Matrix Multiplication benchmark on Amazon EC2 public cloud**.**

There is a relatively huge gap between the average and best performance of running Matrix Multiplication benchmark on Amazon EC2. This gap is due to the fact that several tenants are using a shared resource and the performance of the application depends on the number of simultaneous running applications. Other experiments such as [35] and [36] provide evidences that the multi-tenancy of the network is the major bottleneck and has the greatest influence in degrading the performance of HPC applications in the cloud.

Virtualization Overhead

Virtualization plays a key role in the cloud helping the cloud to have rapid elasticity, resource pooling, and flexibility. However, virtualization and in particular the hypervisor adds unwanted overhead by adding a software layer and preventing applications to have direct access to the hardware resources. This virtualization overhead is not the same for all types of hardware. For example, because of the hardware support, virtualization overhead for processors is significantly less than the overhead of network virtualization. For some hardware types such as GPUs, it is often more efficient to pass through GPUs than to have virtual GPUs [37].

Network Bandwidth and Latency

Network interconnects and I/O resources in the cloud are shared between several tenants. Therefore, the bandwidth and latency of the network may not be predictable. The bandwidth in most cases is much less than what is expected. We conducted an experiment by running Point-to-Point MPI benchmarks [38] on a 10Gig Amazon EC2 interconnect. Figure 3 shows the achieved results. This experiment indicates that the multi-tenant environment of the cloud lowers the bandwidth. Moreover, the latency of the network on the cloud is not stable. Therefore, we will see performance degradation for HPC applications and in particular data intensive ones.

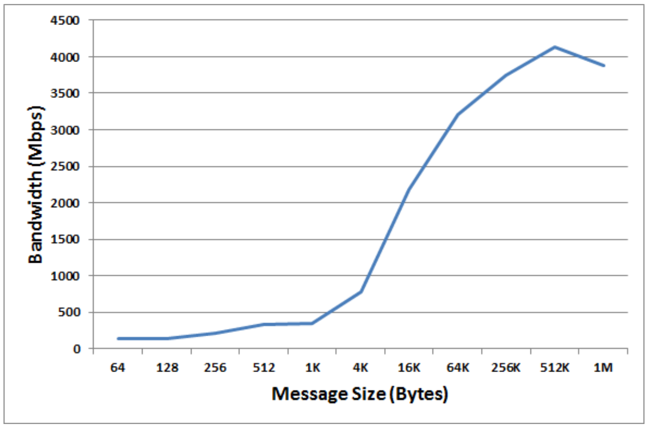

Figure 3: MPI Benchmark on 10G Amazon EC2 Virtual Network

5. Parallel Programming

One distinct advantage of Cloud-based platform for student learning is parallel programming which is the foundation of High Performance Computing (HPC) where the Cloud is ultimately resides. The availability of potentially unlimited processor power offers an outstanding platform for any parallel programming courses.

Cluster, Grid, and High Performance computing is one of the cornerstones of computer science curriculum. To teach students the HPC concept, one needs hands-on practices with different settings of parallel programming paradigm. With the availability of commodity cluster and multicore hardware, it is fairly easy to achieve the basic tool. Nevertheless, the initial costs, the setups, installations, and maintenance are challenging. Availability of an HPCaaS platform will overcome the lack of HPC infrastructure on campus and will offer students the service they need to practice their lab assignments when they need.

Students need to understand and practice with issues such as scalability, synchronization overhead, performance, partitioning, and load balancing of diverse applications. The same rule should apply whether students are using local clusters or the Cloud. However, using actual distributed servers in the cloud can improve students' understandings for issues related to the distributed system, thus improving the learning outcomes.

Numerous parallel models exist today in a parallel programming course. Examples include multi-threading (shared-memory), message-passing, distributed systems, SIMD vs. MIMD style, and recently GPU programming. Students using cloud-based education can exercise, for example, with a Message Passing Interface (MPI) and run their programs. Use of Amazon's EC2 is illustrated in Figure 4. Students may implement an application program using MPI[33]. Building MPI or OpenMP via OpenSSH[39] allows students to test, run, debug, and deploy their applications on a highly scalable distributed system. This process will expose the students with e.g. hidden bugs and other issues mentioned above to understand parallel programming complexities.

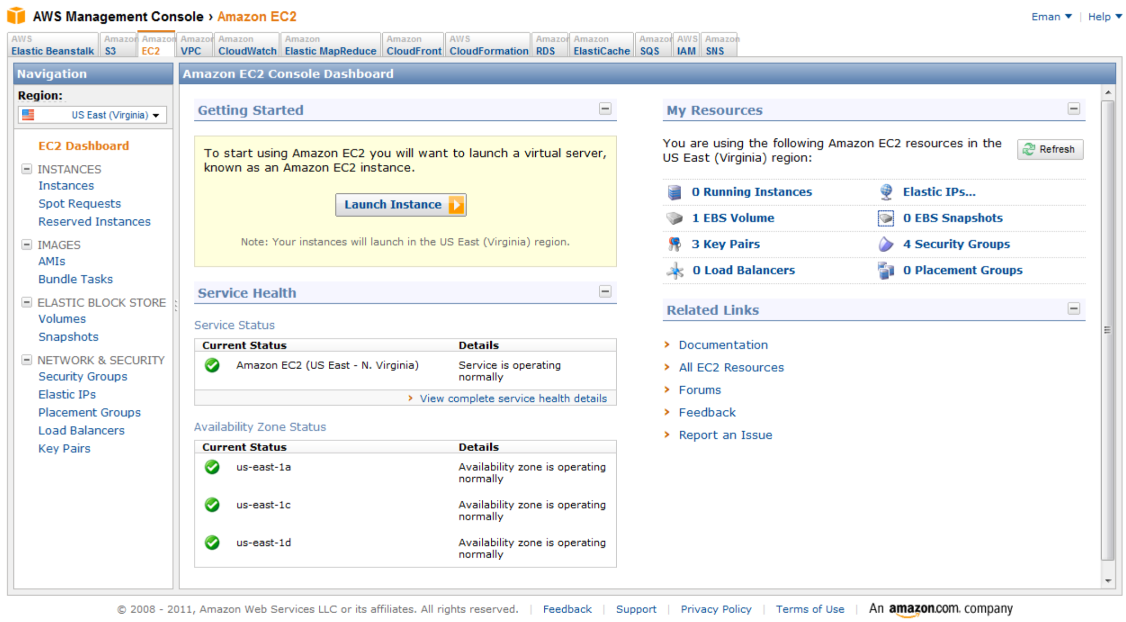

Figure 4: Amazon EC2[21] interface and services snapshot

6. Software-Defined Networking and HPCaaS

Software-Defined Networking (SDN) is an emerging technology in Computer Networks in which the data plane of the network is decoupled from the controller layer [6, 40]. In this architecture, instead of having intelligent routers powered by smart routing algorithms in the data plane, we have switches with assigned flow tables. The flow tables can be updated by the network controller. When a packet arrives in a switch in the data plane of the network, the switch first identifies the flow to which the packet belongs and then forwards the packet according to the flow table. If the flow could not be found in the flow table, then the packet will be forwarded up to the controller. In this way, the controller always has an overview of the whole network properties including the available bandwidths, possible congestions, etc. [40]. Figure 5 illustrates the general conceptual architecture of SDN.

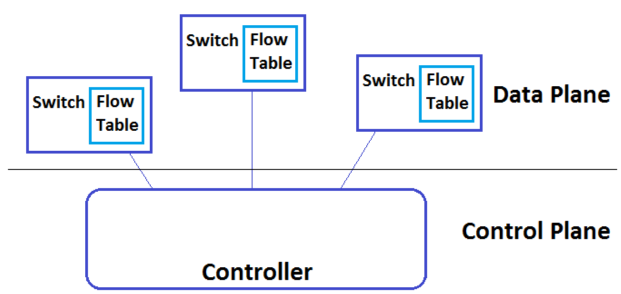

Figure 5: Conceptual Architecture of Software-Defined Networking

The idea of SDN fits very well with the virtual networking of the cloud where everything is virtually defined by software. An SDN controller in the cloud network enables us to be aware of all the links' bandwidths in runtime. Based on this idea, we have proposed a SDN Empowered Task Scheduling System (ASETS) [5]. Figure 6 represents the architecture of the system. Data for the tasks are stored in a shared file system. It is the responsibility of the task scheduler to assign tasks and transfer the appropriate data to the workers (VMs). The task scheduler runs SETSA (SDN Empowered Task Scheduler Algorithm) that takes the bandwidth properties of the network into the account when transferring data to the workers.

SETSA does not remove the multi-tenancy of the cloud, but rather mitigate its influence on the network bandwidth utilization. By increasing the degree of the multi-tenancy in the cloud, the variability of the network bandwidths increases. Our empirical analysis of SETSA shows that the algorithm in such cases can enhance the performance of HPCaaS in term of turnaround time up to 75% by better utilizing the network bandwidth [5]. Furthermore, Analyzing the performance of SETSA on two implementations of ASETS on a private OpenStack cloud and Amazon public cloud indicates that SETSA is capable of improving the performance of HPCaaS significantly up to 67% [41].

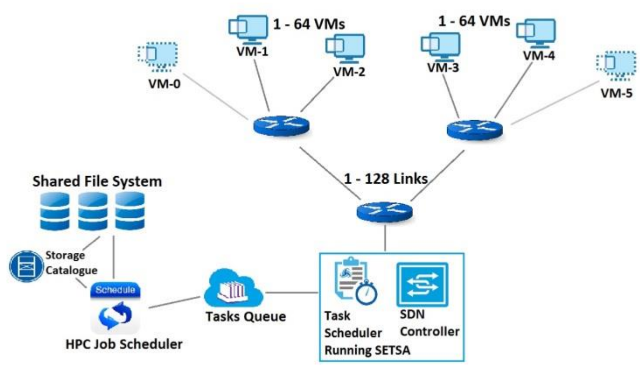

Figure 6: Overview of ASETS architecture

Our experiments with ASETS illustrated promising results to enable the Cloud become a feasible platform for running HPC tasks without suffering from performance degradation. This makes it possible for HPC educators and students to rely on Cloud-based infrastructure for their academic purposes. Furthermore, ASETS sheds lights for other challenging educational use cases of the Cloud such as Big Data and legacy I/O Intensive applications in which the Cloud Networking and Latency are the primary shortcomings.

7. HPC Educational Examples

High performance computing opens the doors for solving numerous fascinating scientific and engineering problems. The price of doing so, however, is the fight that one has to face in several fronts at the same time. In order for a HPC task to obtain the desired answer, the programmer needs first to be familiar with *parallel programming* and *load balancing* issues. After that, the programmer needs to model the application domain problem in form of a *parallel program*.

With a sandbox model supported by HPCaaS, multiple objectives of parallel processing can be practiced. Examples include, *load balancing*, *networking*, *operating systems*, and *parallel processing* to facilitate reaching the desired learning outcomes. In addition, students can learn about concurrent processes, communication and synchronization between the processes, as well as task scheduling and policies.

For parallel programming purpose, the Message-Passing Interface [19, 41] (MPI) library and OpenMP can be used. As a warm-up exercise, we can assign students with matrix multiplication benchmark. In this problem, we assume that we have a large matrix A and B to be multiplied and the result to be stored in matrix C. By varying the dimension of matrices A and B, as well as the number of processors, we can obtain a well-designed exercise where students can observe the following:

- Writing a simple parallel program to be executed on cloud using HPCaaS
- Partitioning the task between the existing number of Virtual Machines (VM) allocated for this exercise
- Scheduling assignment task of multiplication to virtual processors
- Choice of static vs. dynamic task assignments
- Impacts of method of assigning tasks to processors
- Impacts of load balancing to the execution of parallel task
- Impacts of task granularity in performance of the parallel jobs
- Performance of parallel execution and speedup
- Inter-process communications with e.g. *send, receive, scatter, gather, and broadcas*t
- Synchronization issues such as barrier, blocking and non-blocking send and receive
- Understanding the ratio of computation vs. communication
- Debugging issues in parallel execution.
- Compare homogenous and heterogeneous virtual parallel processing platforms
- Race conditions and their impacts on the integrity of the results
- Impacts of mutual exclusions on the performance and speedup efficiency
- Amdahl's Law in parallel programming

The matrix multiplication benchmark is fairly straightforward; nevertheless it provides an excellent training sandbox for HPC students and prepares them to perform other domain applications in a virtual cluster environment provided by HPCaaS platform.

8. Concluding Remarks and Future Work

This paper explored potential benefits of High Performance Computing as a Service, especially for educational use for STEM students. We argued that lack of sufficient infrastructure for teaching and learning parallel programing is a major hinder in widespread use of HPC. We advocate this shortcoming can be dramatically reduced by using HPCaaS platforms and services. Students and educators can benefit from ease of use and availability of the service with modest cost and exercise their learning objectives without interfering with other students resources.

In addition, we identified the benefits as well as the shortcomings of moving HPC applications to the cloud without proper measures. Our experiments confirm these shortcomings. We recognized innovations addressing these HPC challenges and proposed utilizing software-defined networking to improve the performance of HPC applications on the cloud.

Our proposed HPCaaS solution, ASETS, was successfully tested both on Amazon cloud and local private cloud. Nevertheless, we find a stronger need to make packaging the platform for students and make it possible to port it to their desired cloud together with number of benchmarks such as matrix multiplications with MPI and/or OpenMP to demonstrate the transparency and portability of the provided solution.

Bibliography

1. Peter Mell, Tim Grance, "The NIST definition of cloud computing." 2011.
2. Escalante, B. F. (2010). “Cloud Computing Fundamentals”. In Handbook of Cloud Computing. Springer
3. Moustafa AbdelBaky, Manish Parashar, Kirk Jordan, Hyunjoo Kim, Hani Jamjoom, A Zon-Yin Shae, Gergina Pencheva, Vipin Sachdeva, James Sexton, Mary Wheeler, Mary F. Wheeler, "Enabling High-Performance Computing as a Service," Computer, vol. 45, no. 10, pp. 72-80, 2012.
4. S. Azodolmolky, P. Wieder, R. Yahyapour, "Cloud computing networking: challenges and opportunities for innovations," IEEE Communications Magazine, vol. 51, no. 7, pp. 54-62, 2013.
5. Jalian, S, Rajaei, H, 2015, “ASETS: A SDN Empowered Task Scheduling System for HPCaaS On the Cloud”, in Proceedings of IEEE Int. Conf. on Cloud Engineering IC2E/SDS 20015.
6. Nick McKeown, "Software-defined networking," in INFOCOM keynote talk, 2009.
7. Evangelinos, C, Chris N. Hill, "Cloud Computing for parallel Scientific HPC Applications: Feasibility of Running Coupled Atmosphere-Ocean Climate Models on Amazon's EC2," in *The First Workshop on Cloud Computing and its Applications (CCA'08)*, 2008.
8. Gupta, A. Laxmikant V. Kalé, Dejan S. Milojicic, Paolo Faraboschi, Richard Kaufmann, Verdi March, Filippo Gioachin, Chun Hui Suen, Bu-Sung Lee, "Exploring the performance and mapping of HPC applications to platforms in the cloud," in *Proceedings of the 21st international symposium on High-Performance Parallel and Distributed Computing, ACM*, 2012.
9. A. Gupta and D. Milojicic, "Evaluation of HPC Applications on Cloud," in Open Cirrus Summit (OCS), 2011 Sixth, pp. 22-26, 2011.
10. A. Gupta, Kal\'e Laxmikant V, D. Milojicic, P. Faraboschi and S.M. Balle, "Hpc-aware vm placement in infrastructure clouds," in Cloud Engineering (IC2E), 2013 IEEE International Conference on, pp. 11-20, 2013
11. AbdelBaky, Moustafa and Tavakoli, Reza and Wheeler, Mary F and Parashar, Manish and Kim, Hyunjoo and Jordan, Kirk E and Sachdeva, Vipin and Sexton, James and Jamjoom, Hani and Shae, Zon-Yin and others, "Enabling high-performance computing as a service," Computer, pp. 72-80, 2012.
12. Alshuwaier, Faisal, Abdullah A. Alshwaier, and Ali M. Areshey. "Applications of cloud computing in education." Computing and Networking Technology (ICCNT), 2012 8th International Conference on. IEEE, 2012.
13. Mircea, Marinela, and Anca Ioana Andreescu. "Using cloud computing in higher education: A strategy to improve agility in the current financial crisis." Communications of the IBIMA 2011 (2011): 1-15.
14. Gong, C., Liu, J., Zhang, Q., Chen, H. & Gong, Z. (2010) “The Characteristics of Cloud Computing”, Parallel Processing Workshops (ICPPW), 2010 39th International Conference on 13-16 Sept. 2010, pp. 275 – 279..
15. NIST, National Institutes of Standards and Technology, http://www.nist.gov/itl/cloud/
16. M. Chowdhury. “Cloud Computing: Facts, Security, & Challenges.”, Univ of British Columbia, December 2009, www.aipath.com/mchowdhury_law447b.pdf
17. Gong, C., Liu, J., Zhang, Q., Chen, H. & Gong, Z. (2010) “The Characteristics of Cloud Computing”, Parallel Processing Workshops (ICPPW), 2010 39th International Conference on 13-16 Sept. 2010, pp. 275 – 279.
18. Buyya, R., Broberg, J. & Goscinski, A. (2011) “Cloud Computing, Principles and Paradigms”, Wiley & Sons.
19. Microsof Windows Azure Platform, 2011, www.microsoft.com/azure/default.mspx
20. Vouk, M., Averritt, S., Bugaev, M., Kurth, A., Peeler, A., Shaffer, H., Sills, E., Stein, S. & Thompson, J. (2008) “Powered by VCL - Using Virtual Computing Laboratory (VCL)”, Proc. of 2nd Int. Conference on Virtual Computing (ICVCI), 2008, RTP, NC
21. Amazon (2011) Amazon Elastic Compute Cloud (EC2). http://aws.amazon.com/ec2/
22. Google (2011b), Google Apps, http://code.google.com/appengine/

23. Jeremy Goecks, Anton Nekrutenko, James Taylor, "Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences," *Genome Biol*, vol. 11, no. 8, 2010.
24. Wei Lu, Jared Jackson, Roger Barga, "AzureBlast: a case study of developing science applications on the cloud," in *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing (HPDC '10), 413-420, New York, NY, USA, 2010.*
25. Wolfgang Gentzsch, "Workstations, Servers, Clouds: Comparing apples to apples," 17 December 2014. [Online]. Available: https://www.theubercloud.com/workstations-servers-clouds-comparing-apples-apples/. [Accessed 15 January 2015].
26. Tiffany Trader , "HPCWire: Advancing Drug Discovery with HPC Cloud," 10 July 2014. [Online]. Available: http://www.hpcwire.com/2014/07/10/advancing-drug-discovery-hpc-cloud/. [Accessed January 2015].
27. J. Napper, P. Bientinesi, "Can cloud computing reach the top500?," in *Proceedings of the combined workshops on UnConventional high performance computing workshop plus memory access workshop, ser. UCHPC-MAW '09. ACM*, 2009.
28. U.S. Department of Energy (DOE), "Magellan Final Report," 2011. [Online]. Available: http://science.energy.gov/~/media/ascr/pdf/program-documents/docs/Magellan_Final_Report.pdf. [Accessed December 2014].
29. Qiming He, Shujia Zhou, Ben Kobler, Dan Duffy, Tom McGlynn, "Case study for running HPC applications in public clouds," in *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing (HPDC '10). ACM*, New York, NY, 2010.
30. K.R. Jackson, L. Ramakrishnan, K. Muriki, S. Canon, S. Cholia, J. Shalf, Harvey J. Wasserman, N.J Wright, "Performance Analysis of High Performance Computing Applications on the Amazon Web Services Cloud," in *IEEE Second International Conference on Cloud Computing Technology and Science (CloudCom)*, Indianapolis, IN, USA, 2010.
31. Piyush Mehrotra, Jahed Djomehri, Steve Heistand, Robert Hood, Haoqiang Jin, Arthur Lazanoff, Subhash Saini, Rupak Biswas, "Performance evaluation of Amazon EC2 for NASA HPC applications," in *ACM 3rd workshop on Scientific Cloud Computing Date (ScienceCloud '12)*, New York, NY, USA, 2012.
32. Roberto R. Expósito, Guillermo L. Taboada, Sabela Ramos, Juan Touriño, Ramón Doallo, "Performance analysis of HPC applications in the cloud," *Future Generation Computer Systems*, vol. 29, no. 1, pp. 218-229, 2013.
33. R. Householder, S. Arnold, R. Green, "Simulating the Effects of Cloud-based Oversubscription On Data Center Revenues and Performance in Single and Multi-class Service Levels," in *IEEE CLOUD*, Anchorage, Alaska, 2014.
34. R. Iakymchuk, J. Napper, P. Bientinesi, "Underutilizing resources for HPC on clouds," 2010.
35. A. Gupta, D. Milojic, "Evaluation of HPC Applications on Cloud," in *Sixth Open Cirrus Summit (OCS)*, Atlanta, GA, 2011.
36. Abhishek Gupta, Laxmikant V. Kale, Filippo Gioachin, Verdi March, Chun Hui Suen, Bu-Sung Lee, Paolo Faraboschi, Richard Kaufmann, Dejan Milojicic, "The Who, What, Why and How of High Performance Computing Applications in the Cloud," in *Proceedings of the 5th IEEE International Conference on Cloud Computing Technology and Science, ser. CloudCom. vol 13*, 2013.
37. Wei-Shen Ou, Chao-Tung Yang, Yu-Tso Liu, Ching-Hsien Hsu, Hsien-Yi Wang, "On implementation of GPU virtualization using PCI pass-through," in Proceedings of the 2012 IEEE 4th International Conference on Cloud Computing Technology and Science, CLOUDCOM '12 , 2012.
38. "MVAPICH Benchmark over Infiniband," The Ohio State University, [Online]. Available: http://mvapich.cse.ohio-state.edu/benchmarks/. [Accessed January 2015].
39. OpenSSH http://www.openssh.com, 2011
40. Bruno Astuto A. Nunes, Marc Mendonca, Xuan-Nam Nguyen, Katia Obraczka, Thierry Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," *Communications Surveys and Tutorials, IEEE Communications Society, Institute of Electrical and Electronics Engineers (IEEE)*, vol. 16, no. 3, pp. 1617-1634, 2014.
41. Jamalian, S. Rajaei, H, 2015, “Data-Intensive HPC Tasks Scheduling with SDN to Enable HPC-as-a-Service”, in IEEE Cloud 2015 Conference Proceedings, July.
42. Dong, B., Zheng, Q., Yang, J., Li, H. & Qiao, M. (2009) “An e-Learning Ecosystem Based on Cloud Computing Infrastructure”, Ninth IEEE International Conference on Advanced Learning Technologies,