International Journal of Complex Systems – Computing, Sensing and Control
Vol. 3, No. 2, pp. 119-130, 2015
Copyright © 2015, TSI® Press
Printed in the USA. All rights reserved

A SOFTWARE DEFINED NETWORKING ARCHITECTURE FOR HIGH PERFORMANCE CLOUDS[1]

Paul Rad[2], Mo Jamshidi[2], Gilad Berman[3], and John J. Prevost[2]

[2]Electrical and Computer Engineering Department
The University of Texas at San Antonio
San Antonio, Texas 78249

[3]Technology and Engineering, Mellanox Technologies
Hakidma 26, Ofer Industrial Park,
Yokneam, Israel Zip Code 2069200
paul.ra@utsa.edu, moj@wacong.org, giladb@mellanox.com, jeff.prevost@utsa.edu

ABSTRACT—Multi-tenant clouds with resource virtualization offer elasticity of resources and elimination of initial cluster setup cost and time for applications. However, poor network performance, performance variation and noisy neighbors are some of the challenges for execution of high performance applications on public clouds. Utilizing these virtualized resources for scientific applications, which have complex communication patterns, require low latency communication mechanisms and a rich set of communication constructs. To minimize the virtualization overhead, a novel approach for low latency networking for HPC Clouds is proposed and implemented over a multi-technology software defined network. The efficiency of the proposed low-latency SDN is analyzed and evaluated for high performance applications. The results of the experiments show that the latest Mellanox FDR InfiniBand interconnect and Mellanox OpenStack plugin gives the best performance for implementing virtual machine based high performance clouds with large message sizes.

Key Words: InfiniBand; SR-IOV; Software Defined Network; Cloud Computing; High Performance Computing; OpenStack

1. INTRODUCTION

Clusters of independent processors are used for parallelization in a standard High Performance Computing (HPC) environment. HPC typically utilizes the Message Passing Interface (MPI) protocol to communicate between processes. In the traditional approach, applications are executed on compute clusters, super computers or Grid Infrastructure [6],[22] where the availability of resources is limited. High performance computing employs fast interconnect technologies to provide low communication and network latencies for tightly coupled parallel compute jobs. Compute clusters are typically linked by high-speed networks using either gigabit network switches or InfiniBand [20],[23]. Contemporary HPC grids and clusters have a fixed capacity and static runtime environment; they can neither elastically adapt to dynamic

[1] This work was supported, in parts by NSF grant number CNS-1419165 to the University of Texas at San Antonio; and time grants to access the Facilities of the Open Cloud Institute of University of Texas at San Antonio.

workloads nor allocate resources efficiently and concurrently among multiple smaller parallel computing applications [21],[22],[24],[25].

Cloud technology uses an infrastructure that involves a large number of computers connected through a network. Cloud-based services allow users to access resources easily and quickly by paying only for their usage of the resources. Cloud computing offers the benefits of utility-based pricing and the elastic pooling of resources, and it eliminates initial cluster setup cost and time [2]. However, poor network performance, virtualization overhead, low quality of service, and multiple noisy neighbor issues are some of the challenges for execution of real-time, high performance, tightly coupled, parallel applications on Cloud.

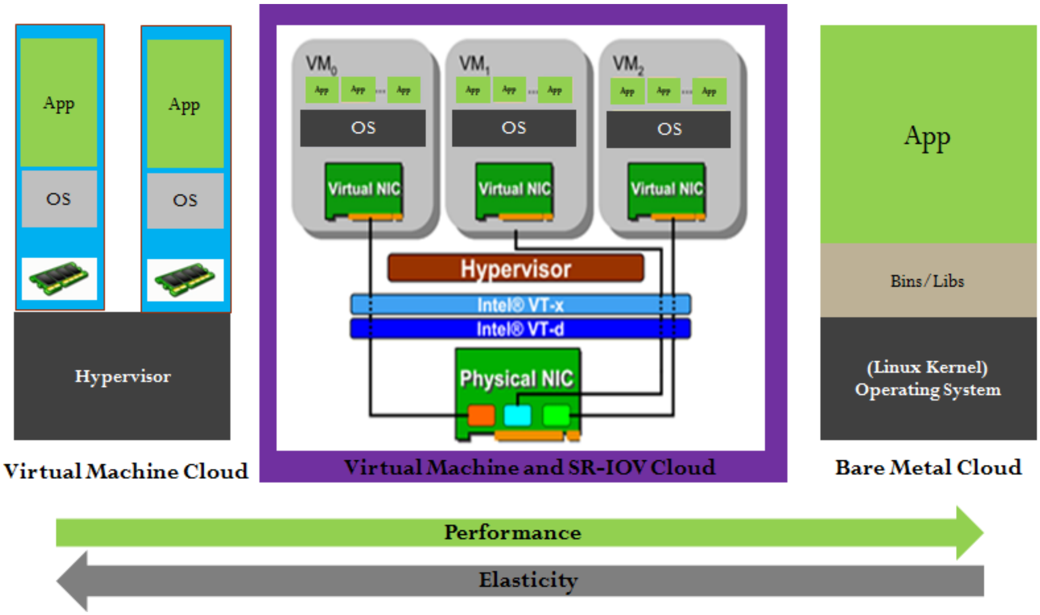

Figure 1. From Virtual Machine Based to Bare Metal Cloud

Traditional network architectures are ill-suited to meet the requirements of today's distributed research infrastructures. A low latency and reliable network built using software defined networking (SDN) among cloud servers is a key element for a cloud infrastructure to be capable of running scientific applications. In the SDN architecture, the control and data planes are decoupled, network management and state are logically centralized, and the underlying network infrastructure is abstracted from the applications. As a result, researchers gain unprecedented programmability, automation, and network control, enabling them to build highly scalable, flexible networks that readily adapt to changing business needs. SDN facilitates fast and reliable transfer of data and communication between cloud servers [1],[4]. InfiniBand is an interesting technology since it offers one of the highest throughputs and lowest latencies, guaranteeing both link Quality of Service (QoS) and scalability. It is often used in supercomputers and in high-performance computing environments [17]. One major challenge to overcome in the deployment of high-performance cloud network is the overhead introduced by virtual switches and virtual devices used and shared by the cloud servers. The Single Root I/O Virtualization (SR-IOV) interface, an extension to the PCI Express (PCIe) specification, overcomes the virtualization overhead by providing device virtualization through virtual functions that reside in the device [15]. This model allows the hypervisor to simply map virtual functions to cloud servers, which can achieve native device performance even without using pass through [5],[19]. The characterization of InfiniBand in bare-metal and virtualized environments has been thoroughly evaluated by the HPC and Virtualization communities [3],[5],[16]. Figure 1 illustrates different cloud stacks based on virtual machines, virtual machine with SR-IOV, containers and bare-mental. However, a comprehensive solution to support HPC applications with low-latency communication requirements levering virtual machine and SR-IOV for SDN is lacking.

The contribution of this paper is twofold. First, we introduce a dynamic configuration of InfiniBand software defined networking with SR-IOV virtualization using the OpenStack neutron plugin in a cloud environment. To the best of our knowledge this is the first paper to present a dynamic flexible low-latency SDN architecture for cloud to support high performance computing. Second, we present a performance evaluation of the proposed architecture using micro benchmarks and an HPC computation library.

In order to understand the latency and bandwidth performance implications of the proposed approaches on cloud resources, a broad performance analysis has been conducted using an OpenStack based cloud configured with low latency SDN using a Mellanox-neutron plugin. Throughout the paper, latency and bandwidth efficiency is defined as the percentage of latency and bandwidth in a virtualized environment compared with a non-virtualized environment utilitzing the same physical resources. To measure performance and efficiency, first we measured individual characterizations such as bandwidth and latency using the IB-verbs and the Intel MPI micro benchmarks [6] with different communication and computation characteristics. Second, we used an application level benchmark, such as the HPL Linpack, to measure the efficiency and the overall performance of a typical scientific application. Our results show that, when large messages used for communication among cloud servers with SR-IOV virtualization, the performance degradation due to network virtualization overhead is low (less than 5%). However, when small message sizes are used for communication, a reduction in performance can be expected compared to the standard HPC grid configuration.

The remainder of the paper is organized as follows. Section 2 provides background information, an overview of related work and our approach for a low latency software defined network for HPC clouds. Section 3 presents a brief introduction to the benchmarks we used and the results of our evaluations. Section 4 concludes the paper with directions for future work.

2. BACKGROUND AND RELATED WORK

The characterization of the InfiniBand in bare-metal and virtualized environments has been thoroughly evaluated by the HPC and Virtualization communities [3],[5],[16]. However, to the best of our knowledge, this is the first paper that offers dynamic configuration of InfiniBand software defined networking (SDN) with SR-IOV in a cloud environment. Our design is based on several existing building blocks, which we introduce in this section. Further, we present related work such as concepts for low latency software defined networking for HPC Clouds.

2.1 OpenStack Cloud Architecture

OpenStack is an open-source cloud management software, which consists of several loosely coupled services, designed to deliver a massively scalable cloud operating system [8] for building public or private clouds. To achieve this, all of the constituent services are designed to work together to provide a complete Infrastructure as a Service (IaaS). All the services collaborate to offer a flexible and scalable cloud solution using the available APIs [7],[11].

The OpenStack software consists of several loosely coupled services with well-defined APIs. While these APIs allow each of the services to use any of the other services, it also allows an implementer to switch out any service as long as they maintain the API.

The implementation described in this paper is based on the Juno release of the OpenStack distribution [8]. Here is a listing of the OpenStack services used in our experiments:

- OpenStack Identity Management ("Keystone") manages a directory of users, a catalog of OpenStack services, and a central authentication mechanism across all OpenStack components.
- OpenStack Compute ("Nova") provides virtual servers upon demand. Nova controls the cloud computing fabric, the core component of an infrastructure service.
- OpenStack Cell allows scaling in very large distributed heterogeneous infrastructures. The compute nodes in an OpenStack cloud are partitioned into groups called cells and cell structure enable distributed tree topology.
- OpenStack Network ("Neutron") provides a pluggable, scalable, and API-driven system for managing networks and IP addresses.
- OpenStack Block Storage ("Cinder") provides persistent block storage that compute instances use.
- OpenStack Image Service ("Glance") provides a catalog and repository for virtual disk images used in OpenStack Compute.
- OpenStack Object Storage ("Swift") provides scalable redundant storage software to store and retrieve object/blob data with a simple API. Swift is ideal for storing unstructured data that can grow without bound.

2.2 Software Defined Networking (SDN) Integration with OpenStack

Software Defined Networking is an emerging architecture that decouples the network control and the flow of packets in the data plane. This new approach makes network management dynamic and adoptable for the high-bandwidth and dynamic nature of today's highly scalable applications. SDN is a network technology that allows for a centralized programmable control plane to manage the entire data plane 12. SDN allows open API communication between the hardware and the operating system, and also between the network elements, both physical and virtualized, and operating system 13.

Integration of the SDN controller into Neutron using plug-ins provides centralized management and also facilitates network programmability of OpenStack networking using the APIs. Figure 2 illustrates the integration of the Mellanox Neutron plugin into OpenStack. The Mellanox Neutron plugin provides for the integration of Mellanox devices with the Neutron service. The Mellanox Neutron plugin creates and configures hardware vNICs based on SR-IOV virtual functions and enables each Virtual Machine vNIC to have its unique connectivity, security, and QoS attributes. The Neutron plugin enables switching in a Mellanox embedded switch in the Virtual Path Identifier (VPI) Ethernet/InfiniBand network adapter. Hardware vNICs are mapped to the guest VMs, through para-virtualization using a TAP device or directly as a Virtual PCI device to the guest via SR-IOV, allowing higher performance and advanced features such as remote direct memory access (RDMA). The OpenStack Neutron controller with the Mellanox plugin comprises of following elements:

- Neutron-server, a python daemon, is the main process of the OpenStack Networking that runs on the OpenStack Network Controller.
- Mellanox OpenStack Neutron Agent runs on each compute node, mapping between a VM vNIC (VIF) and an embedded switch port, thus enabling the VM network connectivity.
- The Mellanox Nova VIF driver is used together with the Mellanox Neutron plugin. This driver supports the VIF plugin by binding vNIC para-virtualized or SR-IOV with optional RDMA guest access to the embedded switch port.

- DHCP agent, a part of Neutron, provides DHCP services to tenant networks. This agent maintains the required DHCP configuration.
- L3 agent: This agent is responsible for providing Layer 3 and NAT forwarding to gain external access for virtual machines on tenant networks.

The Mellanox Neutron plugin is part of the upstream OpenStack release and provides unique value-added features such as transparent InfiniBand network management and configuration. The Mellanox Neutron plugin automatically configures the IB SR-IOV interface for the virtual machines and also assigns Pkey to the interface to provide multi tenancy and network isolation support.

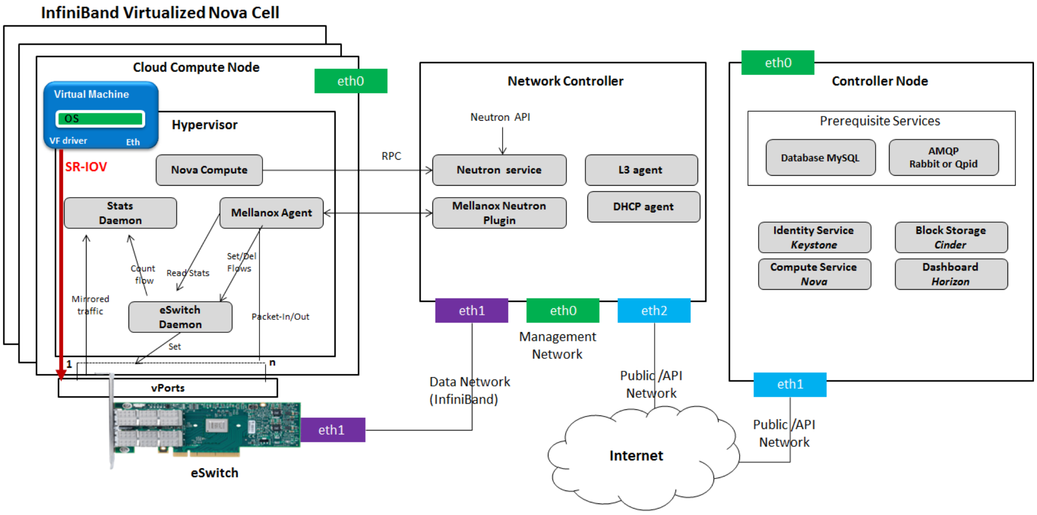

Figure 2. Integration of Mellanox Neutron Plugin

2.3 The InfiniBand and SR-IO Architecture

In this section we provide a short overview of InfiniBand followed by a description of the SR-IOV in the context of our research. We then continue drilling into the experiment and results.

2.3.1 InfiniBand Overview

InfiniBand is a high-performance network technology, which is in widespread use in low latency clusters [6]. Compared to network technologies such as Ethernet, IB has a substantial performance advantage through aggressive protocol offloading; all layers up to the transport layer are handled completely in network adapters with Remote Direct Memory Access (RDMA) over InfiniBand. RDMA is a zero-copy, CPU bypass technology for data transfer and is supported over standard interconnect protocols. RDMA allows applications to transfer data directly to the buffer of a remote application and therefore provides extremely low latency data transfers. The Operating System (OS) is involved only in establishing connections and registering memory buffers to ensure protection. Applications bypass the OS to trigger actual communication operations and poll for their completion, by directly accessing device memory. As a result, an application can handle complete send/receive cycles independently and without added latency from the intervention of the OS. Another capability of the InfiniBand stack, shown in Figure 3, is IP over InfiniBand (IPoIB) protocol. IPoIB protocol defines how to send IP packets using InfiniBand by creating a normal IP network interface. IPoIB impacts the functionalities and performance of the InfiniBand protocol stack in spite of this the user can deploy widespread set of TCP/IP based applications.

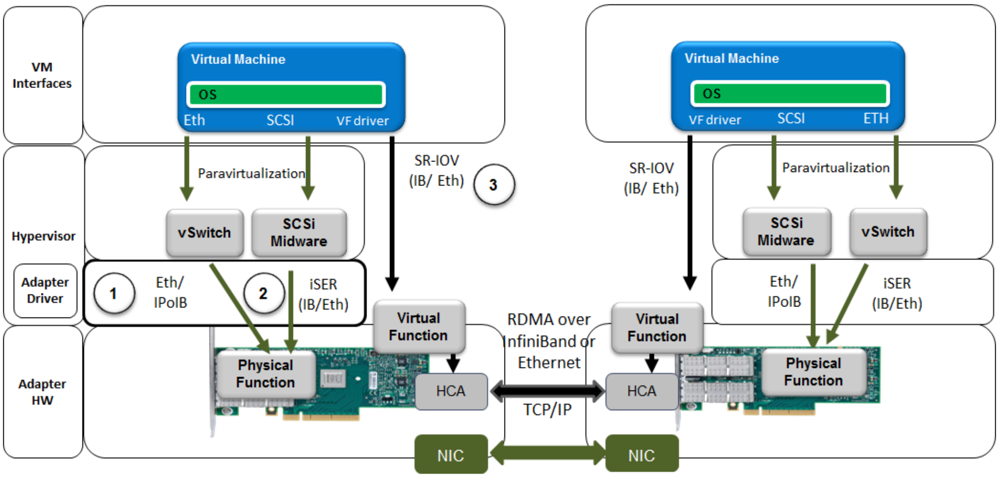

Figure 3. Three ways to leverage RDMA in a cloud environment

2.3.2 SR-IOV Overview

Single Root IO Virtualization (SR-IOV) allows a physical PCIe device to present itself as multiple devices on the PCIe bus. This technology enables a single adapter to provide multiple virtual instances of the device with separate resources. Mellanox ConnectX-3 adapters are capable of exposing virtual instances, called Virtual Functions, which can be provisioned separately. Each Virtual Function (VF) can be viewed as an additional device associated with the Physical Function (PF). In the host hypervisor, these VFs appear as a set of independent PCIe InfiniBand devices. In our InfiniBand Nova cell, each VF is directly associated to a cloud server that has the exclusive use of that function without using any device emulation in hypervisor [18]. And each PF and VF receives a unique PCI Express Requester ID that allows the I/O Memory Management Unit (IOMMU) to differentiate the traffic among VFs.

SR-IOV is commonly used in conjunction with the SR-IOV enabled hypervisor to provide virtual machines with direct hardware access to network resources, thereby improving performance. SR-IOV enables close to zero virtualization latency penalties through RDMA and Hypervisor bypass. Mellanox ConnectX-3 adapters are equipped with an onboard-embedded switch (eSwitch) and are capable of performing layer-2 switching for the different virtual machines running on the server. Higher performance levels can be achieved using eSwitch, since the switching is handled in hardware and reduces CPU overhead.

3. EXPERIMENTS

3.1 Experimental Setup

To evaluate Low-Latency Software Defined Network properties and performance for HPC clouds, we set up two child cloud cells, an InfiniBand Virtualized Nova Cloud cell and an Ethernet Virtualized Nova Cloud cell, under the top-level Open Cloud cell, the UTSA Cloud and Big Data Laboratory. The InfiniBand cell and the Ethernet cloud cell for the evaluation comprises of 16 high-performance Open Compute 12 servers with ConnectX-3 IB adapters and interconnected by

FDR 56Gigabit InfiniBand Mellanox switch. The InfiniBand cell comprises of 8 servers with two 10-core Intel(R) Xeon(R) CPU E5-2670 v2 @ 2.50GHz processors and 256 GB RAM. All servers run Cent OS 6.5 with Linux kernel 2.6 and KVM hypervisor kvm-kmod-3.2. We use the Havana version of OpenStack, whereby 8 servers are configured as OpenStack compute servers as shown in Figures 1 and 2.

3.2 Benchmarks

We have used IB-Verbs benchmarks for IB-Verbs level experiments. All MPI experiments were run using the Intel MPI benchmark 4.1.3.048 [13]. In this section, first we present the performance evaluation of proposed architecture compared to bare-metal servers. IB-Verbs and the Intel MPI Benchmark [13] with different communication and computation characteristics to understand the overhead of cloud resources with pass-through drivers [14]. Then we used an application level benchmark, such as the HPL Linpack, to measure the efficiency and the overall performance of a typical scientific application. Our report results were then averaged across multiple runs to ensure fair comparisons. Figure 5 shows virtualization efficiency calculated from the ratio of bandwidth and latency measurements of IB-Verbs communication between two cloud servers in different hosts and separate measurements of direct IB channel between two hosts for 2 bytes or larger messages.

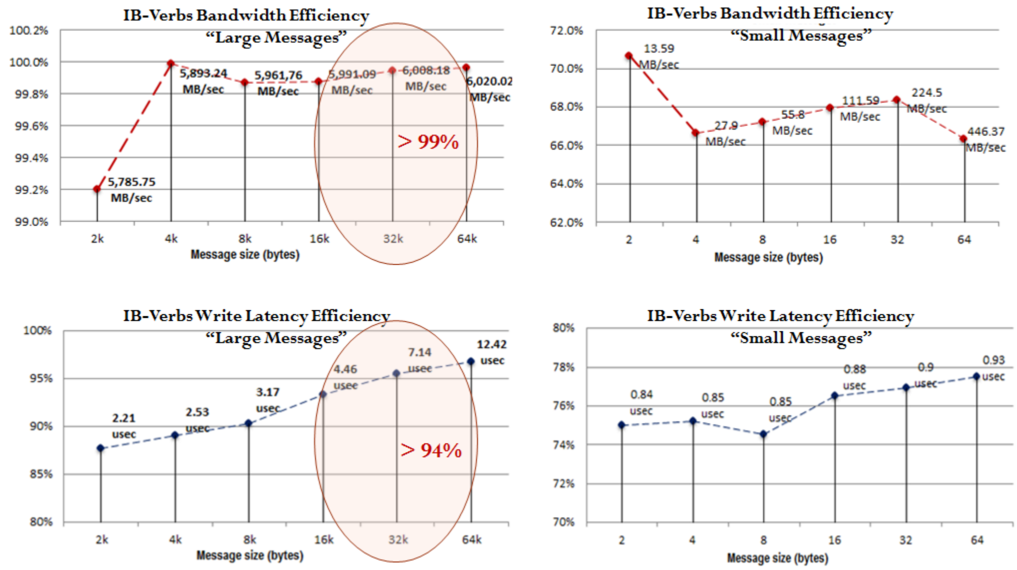

Figure 4. IB-Verbs latency and bandwidth efficiency for small and large messages

For larger message sizes, the difference becomes insignificant and the overhead in latency and bandwidth diminishes to nearly zero with very large message sizes. In this scenario the results are extraordinary, with cloud servers achieving the same network throughput and latency of the host. Figure 5 shows the virtualization efficiency calculated from the ratio of bandwidth and latency measurements of IB-Verbs communication between two cloud servers in different hosts and separate measurements of direct IB channel between two hosts for less than 64 bytes message sizes. When using the IB-verbs benchmark, we witness a big difference for small messages. For messages less than 64bytes, the extra latency caused by virtualization and SR-IOV is on the order of 30%. After micro benchmark IB-Verbs diagnostics tests between hosts and cloud servers, to scale the tests up and evaluate if the efficiency with the IB-Verbs is continued, we used the Intel MPI Benchmark to measure latency and bandwidth. Figure 5 represents virtualization efficiency calculated from the ratio of bandwidth and latency measurements of the Intel MPI Benchmarks between two cloud servers in different hosts and separate measurements of direct IB channel between two hosts with different message sizes. The results are similar to that of IB-Verbs latency.

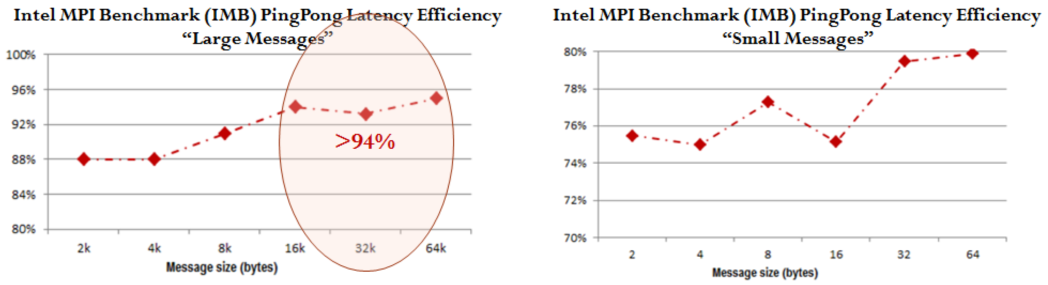

Figure 5. Intel MPI Benchmark (IMB) Ping-Pong latency efficiency

After testing the performance of the network using micro benchmarks, we used HPL Linpack, an application level benchmark, to measure the efficiency and the overall performance of a typical scientific application. To test the worst case scenario in our experiment, we deployed one cloud server per hypervisor and increase the number of processes while the HPL matrix size is kept constant. HPL performance is influenced by the amount of computation per process and message sizes. As the number of processes increase, the fraction of computation decreases linearly, while the message sizes also decrease proportionately. Table I represents the HPL Linpack Benchmark results.

Table I. HPL Linpack Benchmark

#of processes	*2Cloud Server Gflops*	*2Host Gflops*	*Performance Efficiency*
4	71.80	73	98%
8	142	147	97%
16	286	296	97%
20	353	370	95%
40	553	727	75%

Therefore, the impact of communication efficiency becomes higher as the number of processes increases. Also, the message sizes decrease, so there communication efficiency itself goes down as illustrated in IB-Verbs experiments. Therefore, it is a double-whammy effect as the number of processes is increased while the matrix size (the amount of computation) is kept constant. In a preliminary experiment, when large number of processes (40 processes) deployed on two cloud servers, we observed that performance efficiency is only about 75% of the performance with HPC grid configuration due to message size impact illustrated in IB-Verbs experiments. OpenStack Neutron plugin and SR-IOV are still in their early stages. Nevertheless, this low-latency SDN technology, with very little tuning, delivers almost 100% of efficiency in bandwidth and latency for large-message communications. However, the network virtualization and cloud management software will need to be tuned carefully to achieve high communication efficiency.

4. CONCLUSIONS AND FUTURE WORK

Both cloud platforms and traditional grid/cluster systems have different advantages and disadvantages in support of HPC applications. In this paper, InfiniBand Low Latency Software Defined Network for high performance cloud was studied in order to understand the performance characteristics of I/O bound scientific workloads deployed on public or private cloud infrastructures. The InfiniBand Low Latency Software Defined Network combined with SR-IOV is a new architecture proposed for the next-generation High Performance Clouds. This involves two important ideas. First, centralized programmable network control plane architectures with multi-protocol low latency plugins will replace today's proprietary network architectures for hyper-scale infrastructures. Second, cloud computing architectures are used for high performance computing. To the best of our knowledge this is the first paper to present a dynamic flexible low-latency networking architecture for clouds to support high performance computing. Another major contribution of this paper is the evolution of the proposed architecture with micro and application level benchmarks.

Our experimental results show that SDN can provide both operators and providers with unprecedented flexibility and centralized control in building the hyper-scale high performance cloud infrastructure required for scientific applications. The result of the experiment is exceptional, with cloud servers achieving the same network throughput and latency of the hypervisor for large message transfers. Our results for micro benchmarks show that there is about 0.3 ms of overhead for message set up introduced by SR-IOV Virtual Function; the additional cost per byte is negligible. Also, the overhead of the network control plane is negligible since it is required at the beginning to set up the SR-IOV VF functionality. Our Linpack experiments show that when computation is high per process and message sizes are large, the proposed cloud provides more than 95% of the bare-metal performance, which is consistent with the results from micro benchmarks. However the cloud performance can decrease measurably when the number of messages increase while message sizes decrease.

The proposed architecture is our first attempt to make cloud architecture suitable for HPC applications that requires high performance communication support. In the future, we will be further investigating the impact of the entire network stack and recently proposed OnMetal cloud by Rackspace for high performance computing.

ACKNOWLEDGEMENT

This work was supported, in parts by NSF grant number CNS-1419165 to the University of Texas at San Antonio; and time grants to access the facilities of the Open Cloud Institute (OCI) of University of Texas at San Antonio.

REFERENCES

1. Fred, H., M. Malik Salman, and G. Soudeh. "Open Flow as a Service." Retrieved April 21, 2014, Available: https://wiki.engr.illinois.edu/download/ attachments/197298167/ CloudFlow-3rd.pdf?version=6.
2. Gupta, A. and D. Milojicic. "Evaluation of hpc applications on cloud." In *Open Cirrus Summit (OCS), 2011 Sixth*, pp. 22-26. IEEE, 2011.
3. Vienne, J., J. Chen, MD Wasi-ur-Rahman, Nusrat S. Islam, Hari Subramoni, and Dhabaleswar K. Panda. "Performance analysis and evaluation of infiniband fdr and 40gige roce on hpc and cloud computing systems." In *High-Performance Interconnects (HOTI), 2012 IEEE 20th Annual Symposium on*, pp. 48-55. IEEE, 2012.
4. Hwang, K., G. C. Fox, and J. J. Dongarra (2012). Distributed and Cloud Computing: From Parallel Processing to the Internet of Things. Amsterdam: Morgan Kaufmann. Print.

5. Dong, Y., X. Yang, J. Li, G. Liao, K. Tian, and H. Guan. (2012). High performance network virtualization with SR-IOV. Journal of Parallel and Distributed Computing, 72(11), 1471-1480.
6. Jose, J., M. Li, X. Lu, K. C. Kandalla, M. D. Arnold, and D. K. Panda. "SR-IOV Support for Virtualization on InfiniBand Clusters: Early Experience." In *Cluster, Cloud and Grid Computing (CCGrid), 2013 13th IEEE/ACM International Symposium on*, pp. 385-392. IEEE, 2013.
7. Wuhib, F., R. Stadler, and H. Lindgren. "Dynamic resource allocation with management objectives—Implementation for an OpenStack cloud." In *Network and service management (cnsm), 2012 8th international conference and 2012 workshop on systems virtualiztion management (svm)*, pp. 309-315. IEEE, 2012.
8. OpenStack LLC, "OpenStack Cloud Administrator Guide", April 2014,http://docs.openstack.org/admin-guide-cloud/content/index.htm.
9. Shin, M.-K., K.-H. Nam, and H.-J. Kim. "Software-defined networking (SDN): A reference architecture and open APIs." In *ICT Convergence (ICTC), 2012 International Conference on*, pp. 360-361. IEEE, 2012.
10. Devlic, A., W. John, and P. Skoldstrom. "A use-case based analysis of network management functions in the ONF SDN model." In *Software Defined Networking (EWSDN), 2012 European Workshop on*, pp. 85-90. IEEE, 2012.
11. OpenStack: Open Source Cloud Computing Software. http://www.opencompute.org.
12. Open Compute Project. https://www.openstack.org.
13. Intel MPI Benchmarks. https://software.intel.com/en-us/articles/intel-mpi-benchmarks/
14. IP over InfiniBand. http://www.ietf.org/html.charters/ipoib-charter.html.
15. Santos, J. R., Y. Turner, G. J. Janakiraman, and I. Pratt. "Bridging the Gap between Software and Hardware Techniques for I/O Virtualization." In *USENIX Annual Technical Conference*, pp. 29-42. 2008.
16. Koop, M. J., J. K. Sridhar, and D. K. Panda. "TupleQ: Fully-asynchronous and zero-copy MPI over InfiniBand." In *Parallel & Distributed Processing, 2009. IPDPS 2009. IEEE International Symposium on*, pp. 1-8. IEEE, 2009.
17. Grun, P. (2010). Introduction to infiniband for end users. White paper, InfiniBand Trade Association
18. Intel LAN Access Division (2010). PCI-SIG SR-IOV Primer: An Introduction to SR-IOV Technology, White paper
19. Huang, J., X. Ouyang, J. Jose, Md Wasi-ur-Rahman, H. Wang, M. Luo, H. Subramoni, C. Murthy, and D. K. Panda. "High-performance design of hbase with rdma over infiniband." In Parallel & Distributed Processing Symposium (IPDPS), 2012 IEEE 26th International, pp. 774-785. IEEE, 2012.
20. Jackson, K. R., L. Ramakrishnan, K. Muriki, S. Canon, S. Cholia, J. Shalf, H. J. Wasserman, and N. J. Wright. "Performance analysis of high performance computing applications on the amazon web services cloud." In Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on, pp. 159-168. IEEE, 2010.
21. Vecchiola, C., S. Pandey, and R. Buyya. "High-performance cloud computing: A view of scientific applications." In Pervasive Systems, Algorithms, and Networks (ISPAN), 2009 10th International Symposium on, pp. 4-16. IEEE, 2009.
22. Foster, I., Y. Zhao, I. Raicu, and S. Lu. "Cloud computing and grid computing 360-degree compared." In Grid Computing Environments Workshop, 2008. GCE'08, pp. 1-10. IEEE, 2008.
23. P. Rad, R. V. Boppana, P. Lama, G. Berman, and M. Jamshidi "Low-Latency Software Defined Network for High Performance Clouds", the 10th 2015 IEEE International System of Systems Engineering Conference, May 2015.

24. P. Rad, V. Lindberg, J. Prevost, W. Zhang, and M. Jamshidi "ZeroVM: Secure Distributed Processing for Big Data Analytics" , World Automation Congress (WAC) 2014.
25. P. Rad, M. Muppidi, A. S. Jaimes, S. S. Agaian, and M. Jamshidi, "A Novel Image Encryption Method to Reduce Decryption Execution Time in Cloud", the 9th 2015 IEEE International Systems Conference, April 2015.

ABOUT THE AUTHORS

Paul Rad (High performance Cloud Group Chair at Cloud Advisory Council) received his first B.S. degree from Sharif University of Technology in Computer Engineering in 1994. He received his master in computer science from the University of Texas at San Antonio, where he graduated Magna Cum Laude in December 1999. Currently, he is the Chief Research Officer and Assistant Director of Open Cloud Institute (OCI) at the University of Texas, San Antonio, TX, USA. He has advised over 200 companies on virtualization and cloud computing and he has over 70 industry and academic technical publications. He holds 12 US patents on clustering, virtualization, cloud computing and big data analytics with over 200 citations by top 100 leading technology companies such as Microsoft, IBM, Cisco, Amazon Technologies, HP, and VMware. In 2014, he founded the Cloud and BigData Laboratory at the University of Texas, San Antonio, TX, USA as the first Open Compute Project (OCP) Certification Laboratory in the North America. In 2015, he co-founded the Open Cloud Institute where he also leads the industry and academic research cooperation at the University of Texas, San Antonio, TX, USA. He is currently involved in research on cloud computing architecture and cloud federation, big data analytics architecture, image processing & machine learning, hyper-scale computing, container clustering and micro-service application architectures for cloud native apps.

Prof. Mo M. Jamshidi (Fellow IEEE, Fellow ASME, A. Fellow-AIAA, Fellow AAAS, Fellow TWAS, Fellow NYAS, Member Hungarian Academy of Engineering, Member, Russian Academy of NL Systems). He holds honorary doctorate degrees from University of Waterloo, Canada, 2004 and Technical University of Crete, Greece, 2004 and Odlar Yourdu University, Baku, Azerbaijan, 1999. Currently, he is the Lutcher Brown Endowed Distinguished Chaired Professor at the University of Texas, San Antonio, TX, USA. He has been an adviser to NASA (including 1st MARS Mission), USAF Research Laboratory, USDOE and EC/EU (Brussels). He has advised over 70 MS and nearly 55 PhD students. He has over 730 technical publications including 68 books (11 text books), research volumes, and edited volumes in English and a few foreign languages, and over 6500 science citations. He is the Founding Editor or co-founding editor or Editor-in-Chief of 5 journals including IEEE Control Systems Magazine and the IEEE Systems Journal. He has received numerous honors and awards, including five from the IEEE, among others. He is currently involved in research on system of systems engineering, cyber-physical systems, with emphasis on open cloud computing, robotics, UAVs, autonomous vehicles, biological and sustainable energy systems, including smart grids and big data analytic.

Gilad Berman received his first B.S. degree from Tel Aviv University in computer science in 1999. Currently, he is a staff system engineer at Mellanox Technology. He is working as a system and solution engineer at the business development department of Mellanox, focusing on new and emerging technologies. Before working for Mellanox, he worked for IBM Deep

Computing as an HPC architect. In his role at IBM, he designed and led the implementation of some of the largest supercomputers in Europe at the time. He has advised companies on high performance computing. He is currently involved in research on cloud computing, SR-IO, and Infiniband network.

John J. Prevost received his first B.S. degree from Texas A&M in Economics in 1990. He received his second B.S. degree in Electrical Engineering from the University of Texas at San Antonio, where he graduated Magna Cum Laude in December 2009. In 2012 he received his M.S. degree in Electrical Engineering, also from the University of Texas at San Antonio along the way to earning his Ph.D. in Electrical Engineering in December, 2013. He is currently an Assistant Research Professor in the department of Electrical and Computer Engineering at UTSA. In 2015, he co-founded the Open Cloud Institute where he also leads the Research and Education thrust area. Prior to his academic appointment, He served many roles as a technical leader in the computer hardware and software industry. He has served as Director of Product Development, Director of Information Systems and Chief Technical Officer. He is an active consultant in the areas of complex systems and maintains strong ties with industry leaders. His is a member of Tau Beta Pi, Phi Kappa Phi and Eta Kappa Nu Honor Societies, and has been a member of IEEE since 2006. His current research interests include energy aware cloud optimization, cloud controlled robotics, cloud based communications, and quantum cloud computing.