

Evaluating the Suitability of Commercial Clouds for NASA’s High Performance Computing Applications: A Trade Study

S. Chang¹, R. Hood¹, H. Jin², S. Heistand¹, J. Chang¹, S. Cheung¹, J. Djomehri¹, G. Jost³, D. Kokron¹
NASA Advanced Supercomputing Division, NASA Ames Research Center

Executive Summary

NASA’s High-End Computing Capability (HECC) Project is periodically asked if it could be more cost effective through the use of commercial cloud resources. To answer the question, HECC’s Application Performance and Productivity (APP) team undertook a performance and cost evaluation comparing three domains: two commercial cloud providers, Amazon and Penguin, and HECC’s in-house resources—the Pleiades and Electra systems.

In the study, the APP team used a combination of the NAS Parallel Benchmarks (NPB) and six full applications from NASA’s workload on Pleiades and Electra to compare performance of nodes based on three different generations of Intel Xeon processors—Haswell, Broadwell, and Skylake. Because of export control limitations, the most heavily used applications on Pleiades and Electra could not be used in the cloud; therefore, only one of the applications, OpenFOAM, represents work from the Aeronautics Research Mission Directorate and the Human and Exploration Mission Directorate. The other five applications are from the Science Mission Directorate.

In addition to gathering performance information, the APP team also calculated costs as of May 2018 for the runs. In the case of work done on Pleiades and Electra, it used the “full cost” of running, based on the Project’s total annual budget (including all hardware, software, power, maintenance, staff, and facility costs, etc.). In the case of the commercial cloud providers, the team calculated only the compute cost of each run, using published rates and including publicly-known discounts as appropriate. Other infrastructure costs of running in the cloud—such as near-term storage, deep archive storage, network bandwidth, software licensing, and staffing costs for security and security monitoring, application porting and support, problem tracking and resolution, program management and support, and sustaining engineering—were not considered in this study. These “full cloud costs” are likely significant.

While hardware differences across the three domains make it difficult to compare performance in an apples-to-apples manner, the APP team can make some general observations nonetheless. All runs on HECC resources were faster, and sometimes significantly faster, than runs on the most closely matching Amazon Web Services (AWS) resources. The largest differences are most likely due to Amazon’s lack of a true high-performance processor interconnect, which provides a communication fabric between cores on different compute nodes.

For some of the NPBs, Penguin On-Demand (POD) resources yielded faster runs; for others it had slower runs than HECC. Differences in the processor interconnects and communication library are likely the causes of the performance differences. For the

¹ An employee of CSRA LLC, a General Dynamics Information Technology company.

² Point of contact; please direct any correspondence about the work to haoqiang.jin@nasa.gov.

³ An employee of Supersmith.

application benchmarks, performance of resources at Penguin always lagged behind similar resources at HECC.

In all cases, the full cost of running on HECC resources was less than the lowest-possible compute-only cost of running on AWS. To run the full set of the NPBs, AWS was 5.8–12 times more expensive than HECC, depending on the processor type used. For the full-sized applications, AWS was in the best case 1.9 times more expensive.

The NPB runs at POD were 4.7 times more expensive than equivalent runs at HECC. The full-sized applications were 5.3 times more expensive.

Based on this analysis of current performance and cost data, this study finds:

[Finding 1](#): Tightly-coupled, multi-node applications from the NASA workload take somewhat more time when run on cloud-based nodes connected with HPC-level interconnects; they take significantly more time when run on cloud-based nodes that use conventional, Ethernet-based interconnects.

[Finding 2](#): The per-hour full cost of HECC resources is cheaper than the (compute-only) spot price of similar resources at AWS and significantly cheaper than the (compute-only) price of similar resources at POD.

[Finding 3](#): Commercial clouds do not offer a viable, cost-effective approach for replacing in-house HPC resources at NASA.

While the study shows conclusively that it would not be cost effective to run the entire HECC workload on the commercial cloud, there may be cases where cloud resources would prove to be a cost-effective supplement to HECC resources. For example, it may make economic sense to use the cloud to provide access to resources that would be underutilized at HECC, such as GPU-accelerated nodes used for data analytics, physics-based modeling and simulation, or machine/deep learning. It also may make sense to offload some of the HECC workload that would run reasonably well on cloud resources—namely single-node jobs—in order to reduce wait times for remaining HECC jobs. In this case, the economic argument for running small jobs in the cloud is based on the opportunity cost of keeping those jobs at HECC. This analysis leads to the fourth finding:

[Finding 4](#): Commercial clouds provide a variety of resources not available at HECC. Several use cases, such as machine learning, were identified that may be cost effective to run there.

As a result of this study, the APP team has identified and started work on three actions:

[Action 1](#): Get a better understanding of the potential benefits and costs that might accrue from running a portion of the HECC workload in the cloud. This has two parts:

- *Determine the scheduling impact to large jobs of running 1-node jobs in house.*
- *Understand the performance characteristics of jobs that might be run on the cloud.*

[Action 2](#): Define a comprehensive model that allows accurate comparisons of cost between HECC in-house jobs and jobs running in the cloud.

[Action 3](#): Prepare for a broadening of services offered by HECC to include a portion of its workload running on commercial cloud resources. This has two parts:

- *Conduct a pilot study, providing some HECC users with cloud access.*
- *Develop an approach where HECC can act as a broker for HPC use of cloud resources, including user support and an integrated accounting model.*

1. Introduction

Background

After years of development, technologies for cloud computing have become mature, and clouds are being used in a variety of problem domains, including physics-based simulations. There are currently many commercial cloud providers that have the potential to host High Performance Computing (HPC) workloads, including Amazon Web Services (AWS), Penguin On-Demand (POD), Microsoft Azure, and Google Cloud. In addition, several other vendors such as Rescale have begun packaging and reselling cloud resources. To make their offerings attractive, the resellers usually provide a user interface for submitting jobs that simplifies the choice of destination for a job, including the possibility of using in-house computing resources.

The cloud providers and resellers have been aggressive about marketing their products to HPC consumers, usually with analyses of how much money can be saved by moving HPC workloads to the cloud. They make a compelling case for cost efficiency in situations where HPC demand is highly variable. In the case of an HPC facility with a uniformly high utilization rate, however, their argument is less clear and is worthy of investigation.

The Department of Energy undertook such an investigation in 2011 with their Magellan Project [1]. Their comprehensive study found that while clouds have certain features that are attractive for use cases requiring on-demand access to computing resources, the performance of cloud resources was severely lacking for typical HPC workloads with moderate to high levels of communication or I/O. They conclude: *“Our detailed performance analysis, use case studies, and cost analysis shows that DOE HPC centers are significantly more cost-effective than public clouds for many scientific workloads, but this analysis should be reevaluated for other use cases and workloads.”*

The High-End Computing Capability (HECC) project’s Application Performance and Productivity (APP) team has periodically evaluated the claims of cloud providers to be an effective substitute for HPC for NASA. In 2011, APP compared the performance of similar computational resources from the Nebula cloud at NASA Ames, AWS, and HECC’s Pleiades system. The study [2,3,4] used a variety of benchmarks, including full applications from the Pleiades workload, and established that the performance of Nebula for HPC applications was worse than that of the HPC instances of Amazon EC2, which in turn was worse than that of Pleiades, particularly at higher core counts.

A major contributor to the poor performance of Nebula was virtualization of hardware resources. In addition, performance on both cloud platforms suffered from rudimentary processor interconnects that, while cost effective for conventional IT applications, are not sufficient for NASA’s HPC workload. 90% or more of that workload is from physics-based Modeling and Simulation (M&S) applications that run on more than one node. By spreading out the work to cores on multiple nodes—potentially a thousand or more—the computationally intensive applications can be sped up to run in a reasonable amount of time. Such multi-node applications typically need to exchange data between cores running on different nodes and use a Message Passing Interface (MPI) library for that purpose [5]. Most of the HPC applications running at NASA exhibit “tightly-coupled” communication patterns, meaning that there is a significant amount of communication interleaved with the computational components. In tightly-coupled executions, the rudimentary interconnects provided by many commercial cloud providers cannot keep up, and application performance is adversely impacted.

Since 2011, the APP team has been monitoring performance of AWS resources. In 2013, POD resources were included in the tests. From 2014, internal studies of cloud resources looked at new features vendors were providing to see how performance compared to equivalent HECC hardware. Evaluations included aspects such as I/O speeds and batch scheduler features sets. The various tests done in this time frame confirmed the original finding that cloud-based resources could not match the performance of Pleiades on the benchmarks used.

In this new study, the APP team is evaluating the proposition that modern cloud-based resources can be *cost-effective* in comparison to in-house HPC resources. This study differs from previous ones in that it introduces cost into the investigation and it considers circumstances under which the most cost-effective strategy for delivering HPC cycles might include the use of cloud-based resources.

Goals of the Study

The main question to be addressed in this study is whether commercial cloud resources should be used as part of a cost-effective implementation of HECC. Specifically, its first goal is to determine whether commercial cloud resources would make a viable, cost-effective substitute for running all of the current HECC workload. In the event that is not the case, there are two additional goals of the study. One is to determine under what conditions commercial cloud resources might make a viable, cost-effective supplement for running some of the current HECC workload. The other is to provide guidance for a follow-up study to identify which elements of the HECC workload could be offloaded to a commercial cloud and what steps would be necessary to add cloud-based resources to HECC.

2. Approach

Evaluation Workload

The common benchmarks used by HECC's APP team in recent years for architecture evaluation include:

- the NAS Parallel Benchmarks [6], a set of codes that are portable and useful for showing performance of different types of full applications,
- the NAS Technology Refresh (NTR) procurement suite, a set of full-sized applications dating to 2007, and
- HECC's Standard Billing Unit (SBU⁴) suites—both the original one from 2011 [7] and a more recent version from 2017—which have been used to establish the billing rates for HECC resources.

All of the codes in these suites use MPI for inter-process communication. This suite of codes is sized well below the average job workload size on HECC systems, which is currently between 2048 and 4096 ranks, each of which would run on a separate core.

Since the codes would be running on public clouds, the Computational Fluid Dynamics (CFD) codes OVERFLOW, FUN3D, and USM3D, which are in the NTR and SBU suites, could

⁴ The HECC project and the [NASA Center for Climate Simulation](#) (NCCS) use the SBU for allocating and tracking computer resource usage across dissimilar architectures. Representative codes are run on each architecture and their run times compared to the baseline values calculated on a Pleiades Westmere node, whose SBU rate is defined to be 1.0. For example, an Electra Skylake node has an SBU rate of 6.36, as it was measured to be 6.36 times more effective at processing the benchmark suite than a Pleiades Westmere node.

not be used because they are export controlled (ITAR/EAR99). The open source application OpenFOAM was used to represent the CFD members instead. The various benchmarks used in the study are described in the remainder of this section.

NAS Parallel Benchmarks (NPBs)

The NPBs are a set of benchmarks designed to help evaluate the performance of parallel supercomputers. Among the benchmarks, eight of them, including BT, CG, EP, FT, LU, MG, IS, and SP, mimic the computation and data movement in CFD applications. Class C and Class D comprise the third and second largest problem sizes for each benchmark. Two of the benchmarks, BT and SP, require the number of MPI ranks to be a perfect square; the other six require a power of two. For example, Class C runs of BT and SP are performed using 16, 25, 36, 64, 121, 256, 484, and 1024 MPI ranks. For the other six benchmarks, Class C runs are performed using 16, 32, 64, 128, 256, 512, and 1024 ranks.

ATHENA++

ATHENA++ is an astrophysical magneto-hydrodynamics (MHD) code in C++. It is fairly heavily used on Pleiades and is a candidate for future SBU suites. Runs with 512, 1024 and 2048 MPI ranks were attempted at AWS and HECC; not all the AWS runs were successful.

ECCO (MITgcm)

One of the largest users of SBUs in the Science Mission Directorate on Pleiades is the Estimating the Circulation & Climate of the Ocean (ECCO) project. The primary code used in the project is based on the MIT general circulation model (MITgcm), a numerical model designed for study of the atmosphere, ocean, and climate. The test case used for this study is one that was included in the NTR1 (NAS Technology Refresh) benchmarks. Runs with 120 and 240 MPI ranks were performed at AWS and HECC. In the rest of this study, “ECCO” is the label used for the MITgcm results and analysis.

Enzo

Enzo is one of the benchmarks in SBU2 Suite. It is a community-developed, adaptive mesh refinement simulation code, designed for rich, multi-physics hydrodynamic astrophysical calculations. Runs with 196 MPI ranks were performed using the SBU2 dataset.

FVCore

FVCore is one of the benchmarks in SBU1 Suite. It is obtained by extracting the dynamic core of the fvGCM algorithm in GEOS-5, used for Earth science research. The term “fvGCM” has been historically used to refer to the model which was developed in the 1990s at NASA’s Goddard Space Flight Center. The grid is a Cubed-Sphere, so there is no singularity at the poles, and it can be partitioned in ways to test the communication performance between regions as well as the computational performance on the core. The vertical direction was set to be 26 layers, and for each layer the horizontal grid is divided into 6 tiles (6 faces in a cube). Runs with 1176 MPI ranks were performed at AWS and HECC using the SBU1 dataset.

WRF / NU-WRF

The Weather Research and Forecasting (WRF) application is an observation-driven regional Earth system modeling and assimilation system at satellite-resolvable scale. Runs with the WRF SBU1 benchmark used 384 MPI ranks.

NASA-Unified Weather Research and Forecasting (NU-WRF) is one of the benchmarks in the SBU2 Suite. Runs with 1700 MPI ranks with the SBU2 dataset were attempted; there was a problem running on AWS Skylake nodes, however.

OpenFOAM

OpenFOAM is an open source Computational Fluid Dynamics (CFD) application. The Channel395 test case from the OpenFOAM tutorial was used for this study. Runs with 48, 144, and 288 MPI ranks were made on Haswell-based nodes at HECC and AWS.

Evaluation Systems

The HECC production systems currently have 6 different generations of Intel Xeons from Westmere through Skylake. This study compares HECC resources based on the three most recent processor types—Haswell, Broadwell, and Skylake—to similar systems available from cloud providers.

In-house solution: HECC Pleiades and Electra systems

Processor Types and Interconnect:

Three types of Intel Xeon processors were tested:

1. Haswell – Intel Xeon E5-2680v3 CPU @2.5 GHz with 24 cores and 128 GiB of memory per node, and InfiniBand 4X-FDR 56-Gbps Interconnect
2. Broadwell – Intel Xeon E5-2680v4 CPU @2.4 GHz with 28 cores and 128 GiB of memory per node, and InfiniBand 4X-FDR 56-Gbps Interconnect
3. Skylake – Intel Xeon Gold 6148 CPU @2.40GHz with 40 cores and 192 GiB of memory per node, and InfiniBand 4X-EDR 100-Gbps interconnect

Storage Resources:

HECC provides:

- (i) 680 GB of space for a /nasa filesystem that hosts the public software modules,
- (ii) six /home filesystems each about 2 TB, and
- (iii) six Lustre shared /nobackup filesystems with sizes between 1.7 PB and 20 PB.

The testing for this study was performed mainly with /home7 and /nobackupp8.

Operating System:

SUSE Linux Enterprise Server 12 operating system, provided and supported by HPE, was used on these systems.

MPI Library:

MPI applications running on HECC systems are required to use the HPE MPT for its proven scaling performance and prevention of network instability.

Job Scheduling:

Test jobs are submitted from one of the seven Pleiades front-end systems (*pfe[21-27]*) and scheduled to run by the PBS batch job scheduler.

Commercial cloud offering: AWS public cloud available through NASA CSSO/EMCC⁵

The public cloud resources available for this study were from the AWS US West (Oregon) region. The following instance types were used in this study.

Processor Types and Interconnect:

Three types of instances based on Intel Xeon processors were tested:

⁵ All NASA acquisition of cloud-based resources is required to go through the agency OCIO's Cloud Services Service Office (CSSO), which has developed the Enterprise Managed Cloud Computing (EMCC) framework.

1. c4.8xlarge – Haswell processors, Intel Xeon E5-2666 v3 @2.90 GHz with 18 cores, 60 GiB of memory, no local temporary storage per node, and 10-Gbps interconnect
2. m4.16xlarge – Broadwell processors, Intel Xeon E5-2686v4 @2.30 GHz with 32 cores, 256 GiB of memory, no local temporary storage per node, and 25-Gbps interconnect
3. c5.18xlarge – Skylake processors, Intel Xeon Platinum 8124M @3.0 GHz with 36 cores, 144 GiB of memory, no local temporary storage per node, and 25-Gbps interconnect

Storage Resources:

A 50-GB EBS volume was configured for /nasa and /home filesystems and five 250-GB EBS volumes were bundled together and exported as an NFS /nobackup for testing.

Operating System:

Amazon Linux AMI, which has all AWS specific tunings, changes and drivers needed to run well, was chosen for this study. The AWS cost is smaller using this OS compared to, for example, CentOS or SLES OS.

MPI Library:

There is no HPE MPT available on the AWS cloud. Intel-MPI was used for all MPI applications. A user-supplied license is required to build Intel-MPI applications on AWS.

Job Scheduling:

Without a job scheduler in the test environment, a script called spot_manager was created and used on a front-end instance (hostname *nasfe01*) for requesting and managing compute instances used for each run.

Commercial cloud offering: POD

The POD resources available for this study are from a “Proof of Concept” arrangement where the usage was tracked but no actual charges made. The following resources were used in this study.

Processor Types and Interconnect:

Three types of Intel Xeon processors were tested:

1. Haswell Processors in MT1 location - Intel Xeon E5-2660v3 CPU @2.6 GHz with 20 cores and 128 GiB of memory per node, and InfiniBand 4X-QDR 40-Gbps Interconnect
2. Broadwell Processors in MT2 location - Intel Xeon E5-2680v4 CPU @2.4 GHz with 28 cores and 256 GiB of memory per node, and Intel Omni-Path 100-Gbps Interconnect
3. Skylake Processors in MT2 location (early access prior to public release) - Intel Xeon Gold 6148 CPU @2.40 GHz with 40 cores and 384 GiB of memory per node, and Intel Omni-Path 100-Gbps interconnect

Storage Resources:

Both MT1 (NAS filesystem) and MT2 (Lustre filesystem) have about 500 TB of total storage space visible to public. 1 TB was allocated for this testing.

Operating System:

POD configures all MT1 resources with CentOS Linux version 6 and MT2 resources with CentOS Linux version 7.

MPI Library:

There is no HPE MPT available on POD. Intel-MPI is available via their pre-installed Intel compiler modules. However, a user-supplied license is required to use the Intel compiler and MPI library. There are also multiple modules of OpenMPI.

Job Scheduling:

A PBS-like batch job system is available. The commands *qsub*, *qstat*, and *qdel* can be used to manage jobs by each user.

Cost Basis

For each application in the workload, the cost associated with each run was calculated as follows:

HECC Cost Calculation:

A job run on HECC systems was charged in SBUs [8] which are calculated from

- (i) a conversion factor for each processor type (Haswell 3.34; Broadwell 4.04; Skylake 6.36),
- (ii) the number of dedicated nodes used, and
- (iii) the duration of the job in hours.

The current cost of 1 SBU is \$0.16 [8], which is calculated as the total annual HECC costs divided by the total number of SBUs promised to the Mission Directorates in FY18. Note that the total HECC costs utilized in this calculation includes:

- the total annual HECC budget which covers the costs of operating the HECC facility including hardware and software costs, maintenance, support staff, facility maintenance, and electrical power costs, and
- a \$1 Million facility depreciation cost (based on 30-year amortization of the initial capital cost of \$30 Million for the building hosting the resource).

In the current modularized approach to expand the facility beyond the current building, the costs of the containers to host the HPC hardware is also being paid by the project. Thus, the total annual budget being used in the above cost per SBU calculation also includes the cost of infrastructure facility development and expansion.

AWS Cost Calculation:

Because of the difficulty of assessing front-end, storage and bandwidth, technical support, and CSSO/EMCC overhead costs discussed in Appendix I, a full-blown cost for using the AWS resources to run each job is not available for this study. Only the cost of using the compute instances is presented. Depending on the AWS region and the purchase type, availability and cost of an instance type can vary significantly. For example, spot instances are not available for GovCloud. For regions where they are available, spot price fluctuates; it can go above the on-demand price or go lower than 30% of the on-demand price. Thus, four representative (with Amazon Linux OS) prices will be calculated:

- on-demand price of the US-West (Oregon),
- a sample spot price based on 30% of the US-West (Oregon) on-demand price (see Appendix I),
- on-demand price of the GovCloud, and
- a sample pre-leasing price estimated as 70% of the GovCloud on-demand price (see Appendix I).

Table 1 summarizes the various hour-rates of the instance types used in this study. Note that c5 instances are not yet offered for the GovCloud, so no pricing information is available.

POD Cost Calculation:

Similar to AWS, it is difficult to assess the POD login node and storage costs on a per-job basis. Thus, a full-blown cost for using the POD resources to run each job is not available for this study. Only the cost of using the compute resources is presented. For the Haswell, Broadwell, and Skylake, the published rates as of May 2018 of \$0.09, \$0.10, and \$0.11 per core-hour, respectively, will be used in this report. Note that government and volume discounts may be available.

Although POD advertises per-core charging, it turns out that requesting fewer cores than the maximum number of cores per node is not allowed for multi-node jobs. Essentially, the cost of the whole node will be charged. That is, \$1.80 per 20-core Haswell node, \$2.80 per 28-core Broadwell node, and \$4.40 per 40-core Skylake node.

3. Results

Performance and Cost

One of the goals of the study was to determine if clouds can provide a cost-effective substitute for running the HECC workload. To this end, a suite of non-ITAR workloads including the NPBs, Athena++, ECCO, Enzo, FVCORE, WRF/nuWRF, and OpenFOAM, was used to assess the performance and cost effectiveness on AWS and POD versus HECC systems.

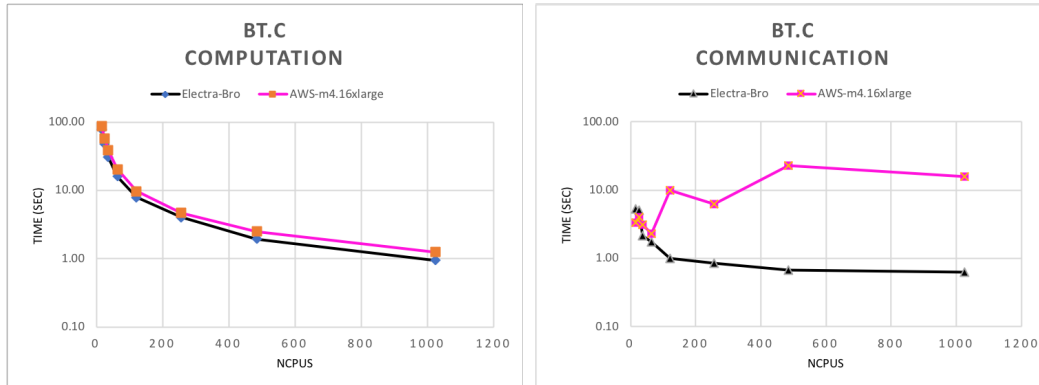
The NPB Class C benchmark runs, with core-count ranging from 16 to 1024 cores, were performed to check scaling performance. In comparison with HECC systems, the results show that the computation performance scales well for runs on AWS, while communication

| | HECC | AWS (Costs are Compute-Only [†]) | | | | | POD |
|--------------|-----------|--|-----------|-------------|--------------------|----------------------|--------------|
| Model/Cores | Full Cost | Instance Name | On-Demand | Spot Price | GovCloud On-Demand | GovCloud Pre-Leasing | Compute Only |
| Haswell/18 | | m4.16xlarge | \$1.591 | \$0.477 | \$1.915 | \$1.341 | |
| Haswell/20 | | | | | | | \$1.800 |
| Haswell/24 | \$0.534 | | | \$0.636* | | | \$2.160* |
| Broadwell/28 | \$0.646 | | | \$0.840* | | | \$2.800 |
| Broadwell/32 | | c4.8xlarge | \$3.200 | \$0.960 | \$4.032 | \$2.822 | |
| Skylake/36 | | c5.18xlarge | \$3.060 | \$0.918** | N/A | N/A | |
| Skylake/40 | \$1.018 | | | \$1.020* ** | | | \$4.400 |

**AWS spot prices in blue are approximations for comparison purposes and are pro-rated based on relative core counts versus HECC. The POD 24-core Haswell price has been similarly converted for comparison to HECC.*
***This estimated AWS spot price for Skylake is unlikely to be available—see Appendix I.*
[†]Full cost information unavailable, but expected to be significantly higher than compute-only cost.

Table 1: Hourly costs of compute nodes used in study.

generally does not, especially across multiple instances. For example, the following graphs show the scaling behavior for the BT.C benchmark:



The poor communication scaling indicates that the 25-Gbps interconnect (maximum of current AWS instances) hinders efficient communication between MPI processes across multiple AWS instances. Details of the Class C NPB runs can be found in Appendix III (Figure 2 and Table 6). The scaling results for the Class D NPBs can also be found there (Tables 7 and 8).

Table 2 compares the cost of running the Class D benchmarks on Intel Broadwell-based nodes at HECC, AWS, and POD. For each benchmark the scaling runs are summarized on a single line, showing the total time and the total number of nodes required by all runs and the total cost for the node hours used by the runs. (See Table 7 in Appendix III for cost details for each scaling run.) In the case of HECC, a “full cost” is used, as was detailed in the Cost Basis discussed above.

Table 2 shows an estimate for the compute charges if the AWS 30% spot pricing is available (\$16.42). It also shows an estimate, \$48.26, for running in a pre-leased block of nodes on the government resources at AWS (i.e. 70% of the cost of AWS Gov on-demand instances). In the best case when spot pricing is used, AWS is 5.8 times more expensive. At the published price, POD is 4.7 times more expensive than HECC.

Table 3 shows similar comparative data for Skylake-based nodes at HECC and AWS. Note that the full cost of the NPB Class D runs on HECC Skylakes is more than 12 times cheaper than the compute-only cost on AWS.

| Benchmark | # of HECC or POD Broadwell nodes | # of AWS m4.16xlarge instances | Total HECC time (sec) | HECC full cost | Total AWS time (sec) | AWS Oregon compute cost | AWS Gov compute cost | Total POD time (sec) | POD compute cost |
|--|----------------------------------|--------------------------------|-----------------------|----------------|----------------------|-------------------------|----------------------|----------------------|------------------|
| bt.D | 47 | 40 | 135.37 | \$0.40 | 327.3 | \$5.55 | \$6.99 | 168.30 | \$2.37 |
| cg.D | 140 | 120 | 192.01 | \$0.99 | 629.75 | \$15.12 | \$19.05 | 150.68 | \$3.41 |
| ep.D | 140 | 120 | 10.62 | \$0.04 | 17.11 | \$0.30 | \$0.38 | 13.84 | \$0.25 |
| ft.D | 29 | 24 | 95 | \$0.20 | 552.59 | \$5.32 | \$6.70 | 58.50 | \$0.59 |
| is.D | 29 | 24 | 10.89 | \$0.02 | 58.09 | \$0.56 | \$0.71 | 7.09 | \$0.08 |
| lu.D | 140 | 120 | 147.3 | \$0.62 | 633.12 | \$17.07 | \$21.51 | 185.14 | \$3.64 |
| mg.D | 140 | 120 | 17.59 | \$0.08 | 52.07 | \$1.03 | \$1.30 | 19.61 | \$0.38 |
| sp.D | 47 | 40 | 152.11 | \$0.46 | 555.84 | \$9.77 | \$12.31 | 171.54 | \$2.46 |
| Total Cost | | | | \$2.81 | | \$54.72 | \$68.94 | | \$13.18 |
| Estimated AWS spot cost (30% of on-demand cost) | | | | | | \$16.42 | | | |
| Estimated AWS pre-leasing cost (70% of US-Gov cost) | | | | | | | \$48.26 | | |

Table 2: Selected NPB class D performance and cost using Broadwell processors on HECC (28 cores/node), POD (28 cores/node) and AWS (32 cores/instance).

| Benchmark | # of HECC Skylake nodes | # of AWS c5.18xlarge instances | Total HECC time (sec) | HECC full cost | Total AWS time (sec) | AWS Oregon compute cost |
|--|-------------------------|--------------------------------|-----------------------|----------------|----------------------|-------------------------|
| bt.D | 33 | 37 | 100.35 | \$0.31 | 238.81 | \$3.27 |
| cg.D | 46 | 52 | 65.05 | \$0.23 | 1984 | \$30.20 |
| ep.D | 46 | 52 | 8.5 | \$0.03 | 8.58 | \$0.09 |
| ft.D | 46 | 52 | 50.3 | \$0.17 | 1118.4 | \$14.02 |
| is.D | 46 | 52 | 4.93 | \$0.02 | 164.5 | \$2.20 |
| lu.D | 46 | 52 | 107.12 | \$0.36 | 327.69 | \$4.64 |
| mg.D | 46 | 52 | 13.36 | \$0.04 | 71.99 | \$0.88 |
| sp.D | 33 | 37 | 121.84 | \$0.35 | 383.5 | \$5.68 |
| Total Cost | | | | \$1.50 | | \$60.98 |
| Estimated AWS spot cost (30% of on-demand cost) | | | | | | \$18.29 |

Table 3: Selected NPB class D performance and cost using Skylake processors on HECC (40 cores/node), and AWS (36 cores/instance).

Table 4 shows similar data for the application benchmarks described in Section 2, comparing the full cost of running on Haswell-based nodes at HECC to the compute-only cost on similar nodes at AWS. Like Tables 2 and 3, it aggregates multiple scaling runs into a single line for each benchmark; the full data can be found in Appendix III (Table 9). Table 5 shows the data for two applications run on HECC and POD, using Haswell-, Broadwell-, and Skylake-based nodes at both sites. In this case, multiple runs are not being aggregated into a single line.

In the best case, where spot pricing is available to run the entire workload, the compute-only AWS cost is still 1.9 times more than the full cost of running on HECC in-house resources. With a pre-leasing option on GovCloud, the AWS compute cost is about 5.2 times that of HECC full cost. The POD compute cost is about 5.3 times that of the HECC full cost.

Analysis

The compute-only cost of using the AWS US-West (Oregon) spot instances to run the workloads is still a few times more expensive than the full-blown HECC cost. For example, as seen in Table 2, for NPB class D on the Broadwell nodes, using spot pricing costs \$16.42 (compute-only) on AWS and \$2.81 (full-blown) on HECC, a ratio of 5.8x. For the NPB class D on the Skylake nodes, as seen in Table 3, it costs \$18.29 (compute-only) on AWS, which is ~12x of the HECC full-blown cost of \$1.50. For the 6 full-sized applications using Haswell nodes, as depicted in Table 4, the compute-only cost on AWS, \$141.77, is 1.9x of the HECC full-blown cost of \$76.18.

| Benchmark | Case | # of HECC Haswell nodes | # of AWS c4.8xlarge instances | HECC time (sec) | Total HECC full cost | AWS time (sec) | Total AWS Oregon compute cost | Total AWS Gov compute cost |
|--|------------|-------------------------|-------------------------------|-----------------|----------------------|----------------|-------------------------------|----------------------------|
| ATHENA++ | SBU2 | 129 | 171 | 3445 | \$29.51 | 3672 | \$127.11 | \$153.00 |
| ECCO | NTR1 | 15 | 21 | 185 | \$0.19 | 313 | \$1.41 | \$1.69 |
| Enzo | SBU2 | 9 | 11 | 1827 | \$2.44 | 2266 | \$11.02 | \$13.26 |
| FVCore | SBU1 | 49 | 66 | 1061 | \$7.72 | 1104 | \$32.20 | \$38.76 |
| nuWRF | SBU2 | 71 | 95 | 529 | \$5.58 | 1302 | \$54.66 | \$65.80 |
| OpenFOAM | Channel395 | 20 | 27 | 27500 | \$30.74 | 51430 | \$246.31 | \$296.46 |
| Total Cost | | | | | \$76.18 | | \$472.71 | \$568.96 |
| Estimated AWS spot cost (30% of on-demand cost) | | | | | | | \$141.77 | |
| Estimated AWS pre-leasing cost (70% of US-gov cost) | | | | | | | | \$398.27 |

Table 4: Selected MPI applications performance and cost using Haswell processors on HECC (24 cores/node) and AWS (18 cores/instance).

| APP | Case | NCPUS | # of HECC nodes | # of POD nodes | HECC time (sec) | Total HECC full cost | POD time (sec) | Total POD compute cost |
|-------------------|------|-------|-----------------|----------------|-----------------|----------------------|----------------|------------------------|
| ENZO (HAS) | SBU2 | 196 | 9 | 10 | 1827 | \$2.44 | 2355 | \$11.78 |
| ENZO (BRO) | SBU2 | 196 | 7 | 7 | 1625 | \$2.04 | 1870 | \$10.18 |
| ENZO (SKY) | SBU2 | 196 | 5 | 5 | 1519 | \$2.15 | 1751 | \$10.70 |
| WRF (HAS) | SBU1 | 384 | 16 | 20 | 1243 | \$2.95 | 1802 | \$18.02 |
| WRF (BRO) | SBU1 | 384 | 14 | 14 | 1225 | \$3.08 | 1436 | \$15.64 |
| WRF (SKY) | SBU1 | 384 | 10 | 10 | 1069 | \$3.02 | 1352 | \$16.52 |
| Total Cost | | | | | | \$15.68 | | \$82.84 |

Table 5: Enzo and WRF performance and cost using Haswell, Broadwell, and Skylake processors on HECC and POD. HECC: Haswell – 24 cores/node, Broadwell – 28 cores/node, Skylake – 40 cores/node; POD: Haswell – 20 cores/node, Broadwell – 28 cores/node, Skylake – 40 cores/node.

The ratios are even higher when using the AWS US-West (Oregon) on-demand, the US-Gov on-demand, or pre-leasing US-Gov instances (see Appendix I for more details). In addition, AWS does not offer spot instances in their government cloud. Paying for on-demand instances on their public or government cloud or pre-leasing option would not be cost effective.

The additional cost of utilizing front-ends, filesystems, data transfer, software licenses, and support on AWS is harder to quantify on a per-job basis. One can however, examine the total charge in different categories. For example, the total AWS charge for the month of December 2017 in conducting the benchmark evaluation was \$1944, of which \$1068 was spent on using various compute instances, \$136 on disk space usage, \$738 on having the front-end available (at a \$4.256 per hour rate), and about \$2 on data transfer. Therefore, there was approximately an 82% ($\$1944/\$1068=1.82$) extra beyond the compute instances cost. Similarly, for January 2018, there was a 97% ($\$1654/\840) extra cost on top of the compute instances cost. Since the workloads used in this evaluation neither use very much storage nor perform large file transfers, it is anticipated that in a production environment, this overhead due to storage and file transfer will probably increase. On the other hand, the overhead due to the use of the front-end will likely decrease if there are many users sharing the front-end system. Also, since the AWS resources used for this evaluation are through CSSO/EMCC, there will be an additional CSSO/EMCC overhead (which includes fees paid to AWS for support services and other operational costs of CSSO/EMCC), which was not yet reported and thus not included in this cost assessment.

For POD, there is a front-end cost and storage cost in addition to the compute cost. Storage is \$0.10 per GB-month, but government discounts will likely apply. There is no cost for data transfer or getting technical support. POD also offers additional volume discounts, but the specifics would not be known until HECC negotiates a cloud services contract with POD.

Even though the POD MT2 resources may provide competitive performance, it is not cost effective compared to running on the HECC resources. For example, as seen in Table 2, the total cost of running the various NPB class D benchmarks on POD Broadwell nodes (\$13.19, compute-only) is 4.7 times that of the HECC Broadwell full-blown cost (\$2.81). The two full-sized application benchmarks, Enzo and WRF, when run on three node types were ~5.3 times higher on POD than the full-blown HECC costs (see Table 5). Note that the storage cost incurred on POD was minimal during this evaluation period. Given that the cost of POD login node and file transfer will likely be free and there will be volume and government discounts with storage, it is expected that the extra cost on top of compute in a production environment will be smaller with POD than with AWS.

Based on the results presented above, the HECC systems are not only much better in their performance but also more cost effective than existing AWS and POD offerings. Neither AWS nor POD would be a viable substitute to HECC for running the traditional HPC MPI (i.e. multi-node) applications.

Usability and Potential Technical Issues

If HECC is going to offer cloud-based resources, then usability is a major concern. The study examines multiple areas in that regard. The complete findings can be found in Appendix IV. The key findings are summarized here:

- Porting of NASA’s traditional HPC MPI applications to the cloud is not trivial and the process can encounter frequent failures due to:
 - (i) system and runtime configuration differences, especially when an application requires a large number of libraries (e.g., for the cases of GEOS-5 from the SBU2 benchmark suite and WRF),
 - (ii) failures to run for some cases of an application for unknown issues (e.g., ATHENA++), and
 - (iii) unexpected poor performance whose cause needs further investigation (e.g., NPB CG and FT, WRF, OpenFOAM).

(See Appendix IV for details.) The time, effort, and cost for NASA scientists, HECC staff, or commercial cloud support staff to investigate and resolve such issues can be significant when migrating to the cloud. In addition, the use of licensed software libraries will increase the cost of running in the cloud in addition to running at HECC.

- AWS offers many operating system options. Since SLES 12 is the current OS used on HECC systems, it is the first choice for trying on AWS since it is likely to introduce fewer porting issues. However, on AWS, SLES OS is more expensive than the basic Amazon Linux OS used in this evaluation. POD offers only CentOS.
- AWS offers no pre-installed software stack. POD has pre-installed more than 250 software modules for HPC use. For commercial software, users have to rely on using licenses provided by HECC or they must bring their own to either AWS or POD.
- AWS does not provide a job scheduling system. The evaluation on AWS used a script-based method created by HECC staff to request, check, and stop instances. POD uses the Moab/Torque management system, which is very similar to the PBS batch scheduler used on HECC systems.
- Given the additional work needed for porting, it may be worthwhile to use “container” technology, such as Docker, to package a job’s image so that the job can run on both HECC and the cloud. Getting this technology to work properly and satisfy NASA security requirements still requires significant development and testing.
- Enabling a hybrid environment, where job bursting can occur from HECC to the cloud, will require a working job scheduling system. Altair has PBS 18 in beta test with Microsoft Azure. Adaptive Computing advertises availability of their Moab/NODUS cloud bursting solution on AWS, Google cloud, AliCloud, Digital Ocean and others. HECC has yet to test the feasibility of these technologies.
- Authorization to Operate (ATO) will be needed to operate on cloud resources such as those at AWS or POD. The best option is likely to extend the NAS moderate security plan, under which HECC operates, to include the use of resources from

specific commercial cloud vendors. At that point, export-controlled (ITAR/EAR99) data would be permitted.

Other Considerations

Another consideration for using clouds is that they may be able to provide access to state-of-the-art resources, such as GPUs, that are not readily available at HECC. To this end, the study found:

- POD lagged behind AWS in offering the Intel Skylake processors. AWS released Skylake on November 7, 2017 for use on their public cloud. POD released Skylake for general use on March 12, 2018. Note, however, that the AWS Skylake processors are in high demand and it can take a long time to get spot instances if one does not want to pay the on-demand price. Also note that no Skylake processors are yet available (as of May 2018) through AWS government cloud.
- AWS lags behind POD in offering higher-performance interconnect. The POD MT2 site already offers 100-Gbps Omni-Path interconnect while AWS currently offers up to 25-Gbps interconnect.
- POD lags behind AWS in offering new GPU processors. AWS has Nvidia M60, K80, and V100 while POD has only Nvidia K40. Currently, HECC also has Nvidia K40.
- POD offers Intel KNL but AWS does not. HECC also has Intel KNL.

4. Findings for NASA's Current HPC Workload

Note that the findings presented in this section reflect the performance and cost at the time of the study (May 2018). Changes in cloud offerings, especially with regard to pricing, may necessitate a reexamination in the future.

Performance

All runs on HECC resources were faster, and sometimes significantly faster, than runs on the most closely matching Amazon resources. The largest differences are most likely due to Amazon's lack of a true high-performance interconnect for processors.

For some of the NPBs, Penguin resources yielded faster runs; for others it had slower runs than HECC. Differences in the processor interconnects and MPI library are likely the causes of the performance differences. For the application benchmarks, performance of resources at Penguin always lagged behind similar resources at HECC. These performance differences lead to the first finding:

Finding 1: *Tightly-coupled, multi-node applications from the NASA workload take somewhat more time when run on cloud-based nodes connected with HPC-level interconnects; they take significantly more time when run on cloud-based nodes that use conventional, Ethernet-based interconnects.*

Cost

With the exception of Skylake processors, Table 1 in Section 2 shows that the base compute-only costs of a node-hour at AWS using spot pricing is more expensive than the full cost of a node-hour on a similar resource at HECC. Skylakes are new and, as shown in Appendix I, the spot pricing estimate is very optimistic. Jobs paying that price are likely to be evicted before completion because of the high demand for the nodes.

Table 1 also shows that the compute-only cost of resources at POD is 4.0–4.3 times more expensive than the full cost for similar resources at HECC. These observations lead to the second finding:

Finding 2: *The per-hour full cost of HECC resources is cheaper than the (compute-only) spot price of similar resources at AWS and significantly cheaper than (compute-only) price of similar resources at POD.*

Cost Effectiveness

In all cases, the full cost of running on HECC resources was less than the lowest-possible compute-only cost of running on AWS. To run the full set of the NPBs, AWS was 5.8–12 times more expensive than HECC, depending on the processor type used. The full-sized applications were in the best case 1.9 times more expensive.

The NPB runs at Penguin were ~4.7 times more expensive than equivalent runs at HECC. The full-sized applications were ~5.3 times more expensive.

The average use of HECC resources is for jobs using 4,000 cores and running for more than 60 hours. At least three-fourths of the workload is by multi-process jobs using tightly coupled communications and the SBU suite has been defined to reflect that usage. The requirements of the HECC workload together with the cost and performance data in this report leads to the third finding:

Finding 3: *Commercial clouds do not offer a viable, cost-effective approach for replacing in-house HPC resources at NASA.*

This finding is based on the expectation that other cloud providers can provide on-demand resources only at or above the spot-price level of AWS. In addition, providers who do not use HPC-level node interconnects will face similar performance penalties as seen on AWS.

Clouds as a Supplement to HECC Resources

While it is not cost effective to use clouds to *replace* in-house HPC resources at NASA, there may be circumstances where it makes economic sense to use them for *augmenting* in-house resources. In identifying candidate uses for clouds it is worthwhile to start with an examination of why HECC costs are so low.

The main factor that keeps HECC costs low is its very high overall average utilization rate—more than 80% of the theoretical maximum number of SBUs in 2017 were delivered to users. Coupled with low hardware acquisition costs and a relatively low ratio of staff-to-HW infrastructure, it is very difficult for cloud vendors to compete with HECC's cost per node-hour. Multiple cloud vendors have acknowledged this fact with comments such as, "If you (HECC) can keep systems 70% utilized or more, then our pricing cannot compete with yours."

Another factor that helps drive down HECC costs is its standard practice to add compute capacity without increasing staffing costs. This pattern is expected to continue with the NAS Facility Expansion effort, which is adding additional floor space and compute resources to HECC. Over time, the plan calls for doubling the space available and equipping it with compute resources, with no additional staffing costs.

Using utilization and responsiveness as drivers, there are two areas where clouds might be useful to HECC:

- When HECC has the need for resources whose long-term demand is sporadic or unknown, it may be more cost effective to use the cloud for those requirements. For example, testing a new architecture often requires a period of high demand followed by a lengthy period of almost no activity. Other examples involve users with requirements for GPU-accelerated nodes, such as for machine learning. Here, the demand is likely long-term, but the extent of the demand is unknown at present.
- When the running of many loosely-coupled (or serial) jobs significantly impacts the scheduling of the HECC's primary workload—large, tightly-coupled computations—then it may be worth paying the price of running the small jobs in the cloud to improve responsiveness for all users.

This analysis and the observation that commercial cloud providers offer many more types of compute resources than HECC is able to (see Appendix I) lead to the fourth finding:

Finding 4: *Commercial clouds provide a variety of resources not available at HECC. Several use cases, such as machine learning, were identified that may be cost effective to run there.*

5. Discussion and Further Work

This section discusses potential cloud use cases in further detail, which leads to some actions the APP team is taking to improve NASA's ability to deliver compute capacity in the most cost-effective way possible.

Using Cloud Resources When User Demand is Unknown or Sporadic

One of the scenarios where it may make sense for HECC to use the cloud rather than acquiring its own resources is when a resource would be lightly or sporadically utilized. For example, HECC is continually procuring small systems to test the usefulness of new technologies for NASA applications such as the many-integrated cores architecture-based Intel Xeon Phi processor or the ARM processor. Unless, these new technologies are seen to perform very well on a large number of applications, their utilization may be limited to a small number of users even after the testing phase is completed. In such cases, where ongoing, long-term demand for the systems is uncertain, HECC should evaluate the cost effectiveness of using cloud-based resources instead of acquiring its own in-house systems.⁶

Another scenario is the use of GPUs for modeling and simulation (M&S) applications. During the time that user demand for GPU cycles for M&S is developing, HECC should consider using cloud-based resources for those jobs. One advantage for users would be access to newer hardware types than HECC has in-house. For example, AWS offers newer GPU processors than either HECC or POD. Although the AWS GPU price is expensive (see Appendix I) and multi-node GPU jobs would face the sort of performance penalties that multi-CPU jobs face, it may still be more cost effective than regularly acquiring new in-house GPUs and certainly provides the ability to opportunistically access as needed.

Similarly, data analytics applications, e.g., machine-learning algorithms, have been shown to run effectively on GPUs. However, since the demand for GPU cycles for such applications is still developing, HECC should consider running such jobs in the cloud. Since the computations are usually loosely coupled, cost effectiveness would likely benefit from

⁶ Note that given relatively high cost of commercial clouds (at least 1.9 times on AWS for applications), utilization of in-house resource has to be below 42% before it becomes cost-effective to use a commercial cloud for the resource.

running on nodes with many GPUs. Such nodes are particularly expensive and would represent a large investment that might only be used a fraction of the time. Running in the cloud could be the most effective solution for such a workload until such time as demand is high enough to justify an on-premises resource. Note that running data analytics applications in the cloud may also face a performance penalty depending on where the data is located. For example, a data analytics application using datasets already at HECC would face additional cost in accessing that data when running on cloud resources.

In any of the above cases, when user demand grows to the point that it is more economical to run on premises, then bringing a resource in-house can be justified using purely an economic argument.

Using Cloud Resources to Improve Responsiveness of HECC In-House Systems

Opportunity Cost of Running Small Jobs on Pleiades/Electra

Experience with the porting and performances of multi-node MPI jobs on the cloud leads to the conclusion that they are not as well suited as single-node jobs may be. A significant portion of usage of HECC in-house resources is by single-node jobs. In 2017, 1.87M one-node jobs (or subjobs) from NASA's Science Mission Directorate (SMD) consumed a total 9M SBUs on Pleiades and Electra. They represented about 85% of the total number of jobs on those systems but only 3% of the total SBU usage.

A question that then naturally comes up in light of these numbers is how the scheduling of the one-node jobs from SMD would affect the other 15% of the jobs on the system. In particular, do wider jobs, i.e. ones using a larger number of processors, get delayed significantly because of the existence of one-node jobs?

While it seems that the relatively small usage (3%) could be accomplished by finding "holes" in the scheduling of wider jobs, the sheer volume of single-node jobs makes the questions worth investigating.

Analysis of Leasing Cloud Resources for Small Jobs

If the productivity impact of leaving all of the small jobs on HECC in-house resources is found to be significant, NASA might consider leasing resources in the cloud in order to run those jobs. Some tradeoff analysis of using leased resources was therefore conducted.

Analysis of PBS historical data shows that the majority of one-node jobs on HECC systems belong to NASA SMD. In addition, it is likely that the SMD jobs have less stringent security requirements. When the effects of delaying the scheduling of wider jobs is considered, migrating some of the one-node SMD jobs from HECC to a commercial cloud might prove to be cost effective for the whole HECC workload. To this end, the study conducted some experiments to determine the utilization percentage and waiting time on the cloud resource that would be seen by HECC when single node jobs are moved from in-house resources to a fixed-size, leased resource in the cloud. Historical job submission data from all one-node SMD jobs in 2017 was used. The size of the leased resource was varied from 2 to 8 racks of 72 nodes each. The results shown in Figure 1 clearly indicate that it would be difficult to size a lease to handle all single node jobs from SMD and have the leased resource both be responsive to user requirements and be well utilized. For example, a lease of 4 racks (288 nodes) would be about 95% utilized but would have an average waiting time of more than 750 hours. If 5 racks (360 nodes) were leased, the wait time would drop to about 150 hours but it would cost 25% more. Thus, any efficient solution to run single node jobs on a leased

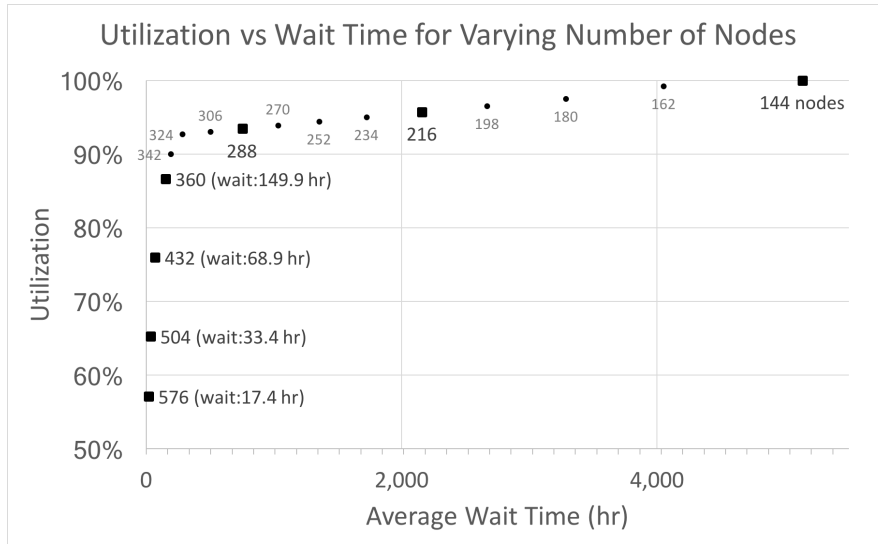


Figure 1: Results of simulations where all one-node jobs in SMD that were submitted in CY 2017 are run instead on leased cloud resources, demonstrating how the size of the leased resource (in number of nodes) gives rise to a trade off in utilization of the leased resource vs. the average wait time for each job.

resource would need to have a fallback (in the on-demand cloud or on in-house resources) to reduce waiting times when demand exceeds the capability of the leased resources.

A cost comparison of using 144 Haswell nodes for one year among HECC-in-house, AWS and POD is summarized below. Note that these Haswell nodes are not identical and the costs presented should be used with caution when drawing conclusions. See Appendix III for more detailed information on how the costs are obtained.

- HECC: using 144 existing Haswell nodes (each with 24 cores, 128 GiB of memory), “unlimited” amount of storage space and multiple front-end nodes costs \$674K.
- AWS: pre-leasing up-front 144 c4.8xlarge instances (each with 18-core, 60 GiB of memory per instance) with the Amazon Linux OS in US West (Oregon) region, 100 TB EBS gp2 volumes, 1 m4.16xlarge as login-node, costs \$1.35M. Pre-leasing instances with other OS (such as SLES, CentOS, etc.), no-up-front, or partial-up-front, or from a different AWS region, or using EFS instead of EBS will cost more. For example, pre-leasing no-up-front 144 c4.8xlarge instances in the GovCloud region, 100 TB EFS, 1 m4.16xlarge as login-node, costs \$1.92M. Note that data transfer cost is not counted. Additional overhead from CSSO/EMCC is not included in the estimate.
- POD: reserving 144 Haswell nodes (each with 20 cores, 128 GiB of memory) and 100 TB storage costs \$2.39M. Note that this cost does not take into account of possible discounts for government agencies, high-volume usage, long-term contracts or dedicated resources that POD will offer. Also, data transfer is free.

As detailed above, the cost of pre-leasing commercial cloud resources for one-node jobs is at least double the cost of using existing HECC in-house resources. However, it may be that the commercial cloud pricing will become more competitive in the future. NASA should monitor performance and costs as the market changes. NASA should also evaluate the cost effectiveness of running its own small-job cluster in-house. The per-node cost of such a cluster would likely be lower than HECC’s typical cluster because the node interconnect would not need to support tightly-coupled communications.

Further Work

With a goal of helping HECC use the most cost-effective way to meet NASA's HPC requirements, the APP team has identified and begun work on three actions:

Action 1: *Get a better understanding of the potential benefits and costs that might accrue from running a portion of the HECC workload in the cloud. This has two parts:*

- *Determine the scheduling impact to large jobs of running 1-node jobs in house.*
- *Understand the performance characteristics of jobs that might be run on the cloud.*

HECC is already in the process of conducting research into how the one-node jobs affect scheduling. HECC's systems team is simulating how scheduling would have occurred over a historical period of job submissions if single-node jobs from SMD were run on cloud resources instead of HECC in-house systems. The result of this study will quantify an "opportunity cost" of delayed scheduling of wide jobs in order to run one-node jobs in-house. That information can be used to help decide whether moving the one-node jobs is worth the likely increase in cost to do so.

In addition, HECC will develop benchmarks to represent workflows that might be run on the cloud. In the case where cloud resources are used to satisfy a sporadic demand, the benchmarks will be used periodically to evaluate the cost of jobs in the cloud and on potential in-house resources to determine when demand has reached that tipping point. The new benchmarks will also be used together with the ones from this study to extend performance evaluations to providers such as Google and Microsoft and to keep results current as providers have new offerings.

In general, such an evaluation for moving cloud computations to in-house resources should take into account factors such as:

- User demand, current and potential, for the resource
- Up-front cost of acquiring and maintaining the resource in house versus the cost of acquiring the resource via a commercial cloud provider
- Need to test the resource within HECC environment, e.g., to test its interaction with the in-house file system
- Possibility of repurposing the hardware after the completion of testing

Action 2: *Define a comprehensive model that allows accurate comparisons of cost between HECC in-house jobs and jobs running in the cloud.*

If HECC is using the cloud as a cost-effective way to provide access to resources that would have low utilization rates, it will periodically need to evaluate when demand is sufficient to warrant purchasing hardware and bringing the computations on premises. When comparing cloud costs to projected in-house costs, it will be important to use apples-to-apples metrics. This study compares the most pessimistic cost of running on HECC resources to the most optimistic cost of running on AWS or POD. The full cost of running on AWS needs to add charges for storage, network bandwidth, and staffing support. It may need to use a higher price for compute resources as well if spot pricing is not available. The full cost of running on POD may need to add charges for staffing support and for storage beyond the modest amount included for free.

An alternative way of comparing costs would be to use a *marginal cost* for computing at HECC. In this approach, the additional costs associated with adding resources to existing HECC resources (mostly capital costs for compute racks) would be used. Then, the

additional cost would be divided by the expected number of SBUs that equipment would deliver to arrive at the marginal cost of providing additional SBUs.

As an example, if HECC added one HPE E-Cell with 288 Skylake nodes, the estimated marginal cost for the new SBUs provided would be in the range of \$0.09–0.10/SBU, depending on the cost of additional storage and network hardware required. This assumes that the new equipment would be utilized at a rate of 80% over a three-year period. This rate is about 60% of the full cost rate per SBU⁷. Before using this approach to make financial decisions, the true marginal costs associated with storage, networking, and support need to be determined.

In comparison, using a spot-pricing estimate of 30% of the on-demand rate, an equivalent number of Skylake node hours at AWS would cost 60–80% more than at HECC. (Note that the spot-pricing rate is *less* than the rate for leasing resources long term from AWS—see Appendix I.) Using POD resources would cost more than 6 times the marginal HECC cost, but government and volume discounts would lower that factor somewhat.

In addition, in order to accurately estimate costs of running in the cloud, HECC needs to get a better understanding of storage and network I/O bandwidth requirements as they interact with cloud offerings. HECC also needs to understand the requirements for a deep tape archive and the long-term preservation of data and how that could be integrated with the cloud.

Action 3: *Prepare for a broadening of services offered by HECC to include a portion of its workload running on commercial cloud resources. This has two parts:*

- *Conduct a pilot study, providing some HECC users with cloud access.*
- *Develop an approach where HECC can act as a broker for NASA's HPC use of cloud resources, including user support and an integrated accounting model.*

There are a number of areas for work to integrate cloud-based resources into HECC production:

- While NASA's CSSO/EMCC has existing agreements with AWS, HECC should evaluate other potential suppliers of cloud resources—potentially through an RFP. Note that NASA requires that money be funneled through CSSO/EMCC, so that organization would likely need to be involved in the procurement.
- Using cloud-based resources in production will require modifications to the NAS Security Plan under which the HECC resources operate. NAS/HECC should extend their moderate plan to cloud-based resources from the vendors NASA chooses to work with. This will likely involve having those vendors be responsible for some of the security controls.
- HECC needs to consider how its users would access cloud-based resources. While many vendors, especially resellers, offer web-based job submission mechanisms, HECC users are accustomed to script-based job submissions. HECC needs to perform requirements analysis on a group of users to determine if web-based systems are preferable before investing money in a software license.

⁷ The assumption of equipment being in production for three years is pessimistic when compared to typical HECC practices. It is more likely that the equipment be in use for 5+ years, thereby lowering the marginal cost of an SBU substantially—probably to something in the range of \$0.05–0.06, depending on costs incurred for maintenance after three years.

- HECC also needs to consider how to support cloud users, especially with setting up images and porting applications.

In considering the moving of some of the HECC workload to the cloud, the pilot study is pursuing the following:

- Studying the workflows of candidate SMD one-node applications to identify and resolve potential issues for migration.
- Investigating the feasibility of using container technology, e.g. Charliecloud, for moving one-node jobs between HECC and AWS, POD or other cloud offerings.
- Testing the cloud bursting capability with the PBS 18 and/or Moab/NODUS job scheduler as a mechanism for HECC users to submit jobs to run in the cloud.
- Designing mechanisms for accounting for cloud usage that integrate cleanly into HECC current accounting user interface.

To act as a broker for NASA’s HPC use of clouds, HECC’s approach needs to include:

- Processes for identifying cloud-appropriate workflows, both from current HECC users and others who approach HECC asking for help,
- Processes, documentation, and training for porting applications to run effectively in the cloud, and
- Mechanisms—potentially through a user portal—for non-HECC users to incorporate cloud usage as part of their workflow.

6. Conclusions

The APP team used a combination of the NAS Parallel Benchmarks (NPB) and six full applications from NASA’s workload on Pleiades and Electra to compare performance of nodes based on three different generations of Intel Xeon processors—Haswell, Broadwell, and Skylake. With the exception of an open-source CFD code standing in for export-controlled codes that could not be run in the cloud, the full applications represent typical work done across NASA’s Mission Directorates.

In addition to gathering performance information, the APP team also calculated costs for the runs. In the case of work done on Pleiades and Electra, the “full cost” of running jobs was determined. In the case of the commercial cloud providers, the team calculated only the compute cost of each run, using published rates and including publicly-known discounts as appropriate. Other infrastructure costs of running in the cloud—such as near-term storage, deep archive storage, network bandwidth, software licensing, and staffing costs for security and security monitoring, application porting and support, problem tracking and resolution, program management and support, and sustaining engineering—were not considered in this study. These “full cloud costs” are likely significant.

Results show that large applications with tightly coupled communications perform worse on cloud resources than on similar resources at HECC. In addition, per-hour use of cloud resources is more expensive than the full cost of using similar resources at HECC. Taken in combination, the data as of May 2018 leads to the conclusion that *commercial clouds do not offer a viable, cost-effective approach for replacing in-house HPC resources at NASA.*

While it is not cost effective to use clouds to *replace* in-house HPC resources at NASA, there may be circumstances where it makes economic sense to use them for *augmenting* in-house resources. This study identified three actions for HECC in anticipation of clouds being useful for some of the HECC workload. The main themes of the actions are for NASA to better

understand the impact and cost of running in the cloud, define a comprehensive model for more accurate comparisons with HECC in-house resources, and to prepare for its likely eventual use for a certain fraction of the agency's HPC workload.

Acknowledgments

This study was funded by the NASA High-End Computing Capability, including work under task ARC013.11.01 on contract NNA07CA29C.

Penguin Computing made their cloud-based resources available to our evaluation at no cost; in addition, they provided access to their Skylake processors prior to public release. The authors appreciate their involvement.

The authors would like to thank R. Biswas, B. Ciotti, L. Hogle, P. Mehrotra, and W. Thigpen for their helpful comments and suggestions.

Acronyms

| | |
|-------|---|
| AMI | Amazon Machine Image |
| AWS | Amazon Web Services |
| CSSO | (NASA) Computer Services Service Organization |
| EAR99 | A classification of items subjecting to Export Administration Regulations |
| EBS | (Amazon) Elastic Block Store |
| EC2 | (Amazon) Elastic Compute Cloud |
| EFS | (Amazon) Elastic File System |
| EMCC | (NASA) Enterprise Managed Cloud Computing |
| HECC | (NASA) High-End Computing Capability |
| HPC | High Performance Computing |
| HPE | Hewlett Packard Enterprise |
| ITAR | International Traffic in Arms Regulations |
| M&S | (Physics-Based) Modeling & Simulation |
| MPI | Message Passing Interface, used for parallel programming |
| NAS | “NASA Advanced Supercomputing Division” or “Network Attached Storage” (depending on context) |
| NFS | Network File System |
| PBS | Portable Batch System |
| POD | Penguin On-Demand |
| SBU | (HECC) Standard Billing Unit |
| SMD | (NASA) Science Mission Directorate |

References

1. U.S. Department of Energy. The Magellan Report on Cloud Computing for Science. Office of Advanced Scientific Computing Research (ASCR), December 2011.
2. P. Mehrotra, R. Hood, J. Chang, S. Cheung, J. Djomehri, S. Heistand, H. Jin, S. Saini, J. Yan, and R. Biswas. Evaluating NASA's Nebula Cloud for Applications from High Performance Computing. NASA Advanced Supercomputing, 2011.
3. P. Mehrotra, J. Djomehri, S. Heistand, R. Hood, H. Jin, A. Lazanoff, S. Saini, and R. Biswas, "Performance Evaluation of Amazon EC2 for NASA HPC Applications," the 21st International ACM Symposium on High-Performance Parallel and Distributed Computing (HPDC'12), Delft, the Netherlands, June 18-22, 2012.
4. S. Saini, S. Heistand, H. Jin, J. Chang, R. Hood, P. Mehrotra, and R. Biswas, "Application Based Performance Evaluation of Nebula, NASA's Cloud Computing Platform," the 14th International Conference on High Performance and Communications (HPCC'12), Liverpool, United Kingdom, June 25-27, 2012.
5. Message Passing Interface Standard, Version 3.1, <https://www.mpi-forum.org/docs/mpi-3.1/mpi31-report.pdf>.
6. NAS Parallel Benchmarks: <https://www.nas.nasa.gov/publications/npb.html>
7. SBU Suite
2011:https://www.hec.nasa.gov/news/features/2011/sbus_042111.html
8. Standard Billing Units, <https://www.hec.nasa.gov/user/policies/sbus.html>
9. Amazon whitepaper April 2017, "Overview of Amazon Web Services", <https://aws.amazon.com/whitepapers/overview-of-amazon-web-services/>
10. AWS current instance types. <https://aws.amazon.com/ec2/instance-types/>
11. Amazon whitepaper December 2016, "AWS Storage Services Overview", <https://aws.amazon.com/whitepapers/storage-options-aws-cloud/>
12. AWS Support Plans, <https://aws.amazon.com/premiumsupport/compare-plans/>
13. Pricing information for AWS on-demand and reserved instances, <https://aws.amazon.com/ec2/pricing/reserved-instances/pricing/>
14. AWS Accelerated Computing, <https://aws.amazon.com/blogs/aws/new-p2-instance-type-for-amazon-ec2-up-to-16-gpus/>
15. AWS EBS Volume Types: <https://aws.amazon.com/ebs/details/#VolumeTypes>
16. AWS EFS Pricing: <https://aws.amazon.com/efs/pricing/>
17. AWS GovCloud (US) Data Transfer Pricing, <https://aws.amazon.com/govcloud-us/pricing/data-transfer/>
18. AWS Data Transfer Costs and How to Minimize Them, <https://datapath.io/resources/blog/what-are-aws-data-transfer-costs-and-how-to-minimize-them/>
19. What is AWS Direct Connect?
<https://docs.aws.amazon.com/directconnect/latest/UserGuide/Welcome.html>
20. Penguin Computing On-Demand (POD) HPC Cloud, <https://www.penguincomputing.com/hpc-cloud/pod-penguin-computing-on-demand/>
21. POD Pricing, <https://www.penguincomputing.com/hpc-cloud/pricing/>
22. Private communication with POD staff, Jan and Feb 2018.
23. Moab/NODUS Cloud Bursting Solution, <http://www.adaptivecomputing.com>
24. AWS Public Sector Contract Center <https://aws.amazon.com/contract-center/>

Appendix I – Amazon Web Services

Resource Overview

The AWS Cloud [9] offers more than 90 AWS services in four main categories: computing power, storage options, networking, and databases. Within each category, there are multiple options to choose from for matching different workloads. For example, in the computing power category, AWS EC2 provides resources in instances where an instance is a virtual server. Loosely speaking, the allocation of AWS resources in units of instances is equivalent to HECC’s resource allocation in units of nodes. The chart from [10] below shows the currently available AWS EC2 instance families and instance types. Four instance families—general purpose, compute-optimized, memory-optimized, and storage-optimized—currently use Intel Xeon processors exclusively. For the compute-optimized instances, the c3, c4, and c5 instances use Ivy Bridge, Haswell and Skylake processors, respectively. The accelerated computing family currently provides the Nvidia Tesla processors (K80, M60 and V100) in addition to the Intel Xeon processors on the instance.

| Instance Family | Current Generation Instance Types |
|-----------------------|---|
| General purpose | t2.nano t2.micro t2.small t2.medium t2.large t2.xlarge t2.2xlarge m3.medium m3.large m3.xlarge m3.2xlarge m4.large m4.xlarge m4.2xlarge m4.4xlarge m4.10xlarge m4.16xlarge m5.large m5.xlarge m5.2xlarge m5.4xlarge m5.12xlarge m5.24xlarge |
| Compute optimized | c3.large c3.xlarge c3.2xlarge c3.4xlarge c3.8xlarge c4.large c4.xlarge c4.2xlarge c4.4xlarge c4.8xlarge c5.large c5.xlarge c5.2xlarge c5.4xlarge c5.9xlarge c5.18xlarge |
| Memory optimized | r3.large r3.xlarge r3.2xlarge r3.4xlarge r3.8xlarge r4.large r4.xlarge r4.2xlarge r4.4xlarge r4.8xlarge r4.16xlarge x1.16xlarge x1.32xlarge x1e.xlarge x1e.2xlarge x1e.4xlarge x1e.8xlarge x1e.16xlarge x1e.32xlarge |
| Storage optimized | d2.xlarge d2.2xlarge d2.4xlarge d2.8xlarge h1.2xlarge h1.4xlarge h1.8xlarge h1.16xlarge i2.xlarge i2.2xlarge i2.4xlarge i2.8xlarge i3.large i3.xlarge i3.2xlarge i3.4xlarge i3.8xlarge i3.16xlarge |
| Accelerated computing | f1.2xlarge f1.16xlarge g2.2xlarge g2.8xlarge g3.4xlarge g3.8xlarge g3.16xlarge p2.xlarge p2.8xlarge p2.16xlarge p3.2xlarge p3.8xlarge p3.16xlarge |

As seen in the following table obtained through the spot_manager script created by HECC staff, each instance type is characterized by:









- (i) amount of memory,
- (ii) number of cores,
- (iii) number of GPU processors included,
- (iv) whether there is a local temporary storage included and its size,
- (v) network performance and
- (vi) whether launching the instances in a placement group is allowed.

There are two types of placement groups: “cluster” group for clustering instances into a low-latency group, and “spread” group for spreading instances across underlying hardware to reduce risks of simultaneous failures. For network performance, most instance types are configured with 10 Gbps or less. A few instance types are configured with 25-Gbps interconnect.

| Type | Memory | Cores | GPUs | Local Temp Storage | Network Perf | Enhanced Networking | Placement Group |
|------------|-----------|----------|------|--------------------|--------------|---------------------|-----------------|
| d2.xlarge | 30.5 GiB | 2 cores | | 6000 GiB | Moderate | Yes | Yes |
| d2.2xlarge | 61.0 GiB | 4 cores | | 12000 GiB | High | Yes | Yes |
| d2.4xlarge | 122.0 GiB | 8 cores | | 24000 GiB | High | Yes | Yes |
| d2.8xlarge | 244.0 GiB | 18 cores | | 48000 GiB | 10 Gigabit | Yes | Yes |
| r3.large | 15.25 GiB | 1 cores | | 32 GiB | Moderate | Yes | Yes |
| r3.xlarge | 30.5 GiB | 2 cores | | 80 GiB | Moderate | Yes | Yes |
| r3.2xlarge | 61.0 GiB | 4 cores | | 160 GiB | High | Yes | Yes |
| r3.4xlarge | 122.0 GiB | 8 cores | | 320 GiB | High | Yes | Yes |
| r3.8xlarge | 244.0 GiB | 16 cores | | 640 GiB | 10 Gigabit | Yes | Yes |

| | | | | | | | |
|--------------|------------|----------|----|-----------|------------------|-----|-----|
| i3.large | 15.25 GiB | 1 cores | | 475 GiB | Up to 10 Gigabit | Yes | Yes |
| i3.xlarge | 30.5 GiB | 2 cores | | 950 GiB | Up to 10 Gigabit | Yes | Yes |
| i3.2xlarge | 61.0 GiB | 4 cores | | 1900 GiB | Up to 10 Gigabit | Yes | Yes |
| i3.4xlarge | 122.0 GiB | 8 cores | | 3800 GiB | Up to 10 Gigabit | Yes | Yes |
| i3.8xlarge | 244.0 GiB | 16 cores | | 7600 GiB | 10 Gigabit | Yes | Yes |
| i3.16xlarge | 488.0 GiB | 32 cores | | 15200 GiB | 25 Gigabit | Yes | Yes |
| c3.large | 3.75 GiB | 1 cores | | 32 GiB | Moderate | Yes | Yes |
| c3.xlarge | 7.5 GiB | 2 cores | | 80 GiB | Moderate | Yes | Yes |
| c3.2xlarge | 15.0 GiB | 4 cores | | 160 GiB | High | Yes | Yes |
| c3.4xlarge | 30.0 GiB | 8 cores | | 320 GiB | High | Yes | Yes |
| c3.8xlarge | 60.0 GiB | 16 cores | | 640 GiB | 10 Gigabit | Yes | Yes |
| r4.large | 15.25 GiB | 1 cores | | | Up to 10 Gigabit | Yes | Yes |
| r4.xlarge | 30.5 GiB | 2 cores | | | Up to 10 Gigabit | Yes | Yes |
| r4.2xlarge | 61.0 GiB | 4 cores | | | Up to 10 Gigabit | Yes | Yes |
| r4.4xlarge | 122.0 GiB | 8 cores | | | Up to 10 Gigabit | Yes | Yes |
| r4.8xlarge | 244.0 GiB | 16 cores | | | 10 Gigabit | Yes | Yes |
| r4.16xlarge | 488.0 GiB | 32 cores | | | 25 Gigabit | Yes | Yes |
| p3.2xlarge | 61.0 GiB | 4 cores | 1 | | Up to 10 Gigabit | Yes | Yes |
| p3.8xlarge | 244.0 GiB | 16 cores | 4 | | 10 Gigabit | Yes | Yes |
| p3.16xlarge | 488.0 GiB | 32 cores | 8 | | 25 Gigabit | Yes | Yes |
| x1.16xlarge | 976.0 GiB | 32 cores | | 1920 GiB | 10 Gigabit | Yes | Yes |
| x1.32xlarge | 1952.0 GiB | 64 cores | | 3840 GiB | 25 Gigabit | Yes | Yes |
| x1e.32xlarge | 3904.0 GiB | 64 cores | | 3840 GiB | 25 Gigabit | Yes | No |
| t2.large | 8.0 GiB | 1 cores | | | Low to Moderate | No | No |
| t2.xlarge | 16.0 GiB | 2 cores | | | Moderate | No | No |
| t2.2xlarge | 32.0 GiB | 4 cores | | | Moderate | No | No |
| p2.xlarge | 61.0 GiB | 2 cores | 1 | | High | Yes | Yes |
| p2.8xlarge | 488.0 GiB | 16 cores | 8 | | 10 Gigabit | Yes | Yes |
| p2.16xlarge | 732.0 GiB | 32 cores | 16 | | 25 Gigabit | Yes | Yes |
| g3.4xlarge | 122.0 GiB | 8 cores | 1 | | Up to 10 Gigabit | Yes | Yes |
| g3.8xlarge | 244.0 GiB | 16 cores | 2 | | 10 Gigabit | Yes | Yes |
| g3.16xlarge | 488.0 GiB | 32 cores | 4 | | 25 Gigabit | Yes | Yes |
| m4.large | 8.0 GiB | 1 cores | | | Moderate | Yes | Yes |
| m4.xlarge | 16.0 GiB | 2 cores | | | High | Yes | Yes |
| m4.2xlarge | 32.0 GiB | 4 cores | | | High | Yes | Yes |
| m4.4xlarge | 64.0 GiB | 8 cores | | | High | Yes | Yes |
| m4.10xlarge | 160.0 GiB | 20 cores | | | 10 Gigabit | Yes | Yes |
| m4.16xlarge | 256.0 GiB | 32 cores | | | 25 Gigabit | Yes | Yes |
| c4.large | 3.75 GiB | 1 cores | | | Moderate | Yes | Yes |
| c4.xlarge | 7.5 GiB | 2 cores | | | High | Yes | Yes |
| c4.2xlarge | 15.0 GiB | 4 cores | | | High | Yes | Yes |
| c4.4xlarge | 30.0 GiB | 8 cores | | | High | Yes | Yes |
| c4.8xlarge | 60.0 GiB | 18 cores | | | 10 Gigabit | Yes | Yes |
| c5.large | 4.0 GiB | 1 cores | | | Up to 10 Gbps | Yes | Yes |
| c5.xlarge | 8.0 GiB | 2 cores | | | Up to 10 Gbps | Yes | Yes |
| c5.2xlarge | 16.0 GiB | 4 cores | | | Up to 10 Gbps | Yes | Yes |
| c5.4xlarge | 32.0 GiB | 8 cores | | | Up to 10 Gbps | Yes | Yes |
| c5.9xlarge | 72.0 GiB | 18 cores | | | 10 Gigabit | Yes | Yes |
| c5.18xlarge | 144.0 GiB | 36 cores | | | 25 Gigabit | Yes | Yes |

Similarly, there are multiple storage services [11] to choose from as shown in this chart:

| | | |
|---|---|--|
|  | Amazon Simple Storage Service (Amazon S3) | A service that provides scalable and highly durable object storage in the cloud. |
|  | Amazon Glacier | A service that provides low-cost highly durable archive storage in the cloud. |
|  | Amazon Elastic File System (Amazon EFS) | A service that provides scalable network file storage for Amazon EC2 instances. |
|  | Amazon Elastic Block Store (Amazon EBS) | A service that provides block storage volumes for Amazon EC2 instances. |
|  | Amazon EC2 Instance Storage | Temporary block storage volumes for Amazon EC2 instances. |
|  | AWS Storage Gateway | An on-premises storage appliance that integrates with cloud storage. |
|  | AWS Snowball | A service that transports large amounts of data to and from the cloud. |
|  | Amazon CloudFront | A service that provides a global content delivery network (CDN). |

Among these services, AWS EBS volumes can be used for persistent local storage. Data in EBS volumes cannot be shared between instances unless exported as an NFS filesystem from the instance the EBS volume is mounted on. The EBS bandwidth has limits depending on the volume and instance sizes. EFS is an NFS filesystem; thus, the data is available to one or more EC2 instances and across multiple Availability Zones (AZs) in the same region. S3 is a scalable and durable object-based storage system; data in S3 can be made accessible from any internet location. Point in time snapshots of EBS and EFS volumes can be taken and stored in S3. AWS Glacier provides long-term archival storage.

The AWS services are available in more than 16 regions around the world, each with a few AZs. Each region is completely independent. The two regions closest to the HECC facility are US West (N. California) and US West (Oregon). There is also a region marked as AWS GovCloud (US) which is designed to address stringent U.S. government security and compliance requirements. A region has multiple AZs. Each AZ is isolated, but the AZs in a region are connected through low-latency links.

Pricing

Usage on AWS is charged on three fundamental characteristics: compute, storage, and data transfer out. There is no charge for inbound data transfer or for data transfer between other Amazon Web Services within the same region. In addition, AWS offers four different custom support plans [12]: basic, developer, business, and enterprise. The basic support plan is included but has no access to technical support resources and beyond.

Cost of compute instances

There are four purchase types for AWS EC2 instances:

- On-demand instances: full price. The customer pays for compute capacity by the hour with no long-term commitments. Amazon may change the price once a year.
- Reserved instances: discounted rate from full price. Instances are required to be reserved for 1–3 years. May get volume discounts up to 10% when you reserve more.
- Spot instances: bid price. The customer bids for unused instances and prices fluctuate about every 5 minutes. Note that it is possible for the spot price to go over the on-demand price. Spot instances can be interrupted by EC2 with 2 minutes of notification when EC2 needs the capacity back.
- Dedicated hosts: on-demand price for instances on physical servers dedicated for your use

Prices [13] vary with AWS regions, OS (categorized as Amazon Linux, RHEL, SLES, Windows, etc.; RHEL and SLES are more expensive than Amazon Linux.), number of cores, memory, and other factors. Usage is billed on one-second increments, with a minimum of 60 seconds.

The following three charts show sample pricing for a c4.8xlarge instance with the Linux OS in three regions: (i) GovCloud (US), (ii) US West (N. California), and (iii) US West (Oregon). In each chart, reserved and on-demand pricings for standard 1-year term and convertible 1-year term are shown. The convertible 1-year term allows for flexibility to use different instance families, OS, etc. As seen in these charts, pricing for US West (Oregon) region is lower than the GovCloud (US) region while the US West (N. California) region is the most expensive among the three.

Pricing for GovCloud (US)

c4.8xlarge

| STANDARD 1-YEAR TERM | | | | | |
|-------------------------|---------|-----------|--------------------|------------------------|------------------|
| Payment Option | Upfront | Monthly* | Effective Hourly** | Savings over On-Demand | On-Demand Hourly |
| No Upfront | \$0 | \$883.30 | \$1.210 | 37% | \$1.915 per Hour |
| Partial Upfront | \$5055 | \$421.21 | \$1.154 | 40% | |
| All Upfront | \$9907 | \$0 | \$1.131 | 41% | |
| CONVERTIBLE 1-YEAR TERM | | | | | |
| Payment Option | Upfront | Monthly* | Effective Hourly** | Savings over On-Demand | On-Demand Hourly |
| No Upfront | \$0 | \$1015.43 | \$1.391 | 27% | \$1.915 per Hour |
| Partial Upfront | \$5803 | \$483.26 | \$1.324 | 31% | |
| All Upfront | \$11373 | \$0 | \$1.298 | 32% | |

Pricing for US West (N. California)

c4.8xlarge

| STANDARD 1-YEAR TERM | | | | | |
|-------------------------|---------|-----------|--------------------|------------------------|------------------|
| Payment Option | Upfront | Monthly* | Effective Hourly** | Savings over On-Demand | On-Demand Hourly |
| No Upfront | \$0 | \$993.53 | \$1.361 | 32% | \$1.993 per Hour |
| Partial Upfront | \$5690 | \$474.14 | \$1.299 | 35% | |
| All Upfront | \$11152 | \$0 | \$1.273 | 36% | |
| CONVERTIBLE 1-YEAR TERM | | | | | |
| Payment Option | Upfront | Monthly* | Effective Hourly** | Savings over On-Demand | On-Demand Hourly |
| No Upfront | \$0 | \$1142.45 | \$1.565 | 21% | \$1.993 per Hour |
| Partial Upfront | \$6528 | \$543.85 | \$1.490 | 25% | |
| All Upfront | \$12795 | \$0 | \$1.461 | 27% | |

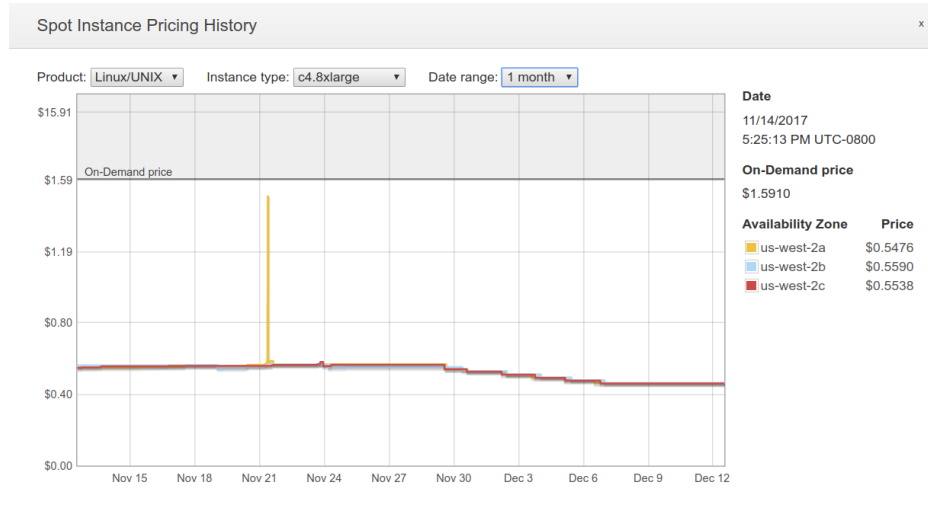
Pricing for US West (Oregon)

c4.8xlarge

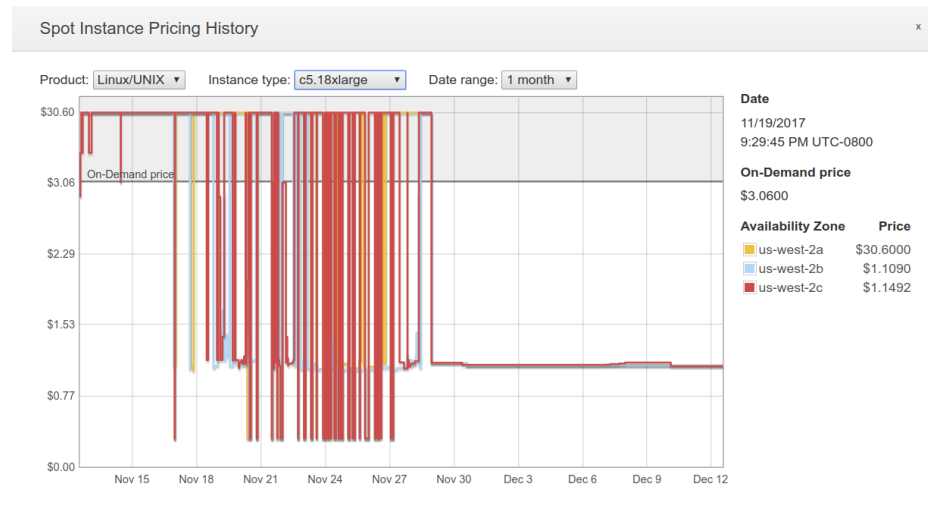
| STANDARD 1-YEAR TERM | | | | | |
|-------------------------|---------|----------|--------------------|------------------------|------------------|
| Payment Option | Upfront | Monthly* | Effective Hourly** | Savings over On-Demand | On-Demand Hourly |
| No Upfront | \$0 | \$735.84 | \$1.008 | 37% | \$1.591 per Hour |
| Partial Upfront | \$4231 | \$352.59 | \$0.966 | 39% | |
| All Upfront | \$8293 | \$0 | \$0.947 | 40% | |
| CONVERTIBLE 1-YEAR TERM | | | | | |
| Payment Option | Upfront | Monthly* | Effective Hourly** | Savings over On-Demand | On-Demand Hourly |
| No Upfront | \$0 | \$846.07 | \$1.159 | 27% | \$1.591 per Hour |
| Partial Upfront | \$4836 | \$402.96 | \$1.104 | 31% | |
| All Upfront | \$9478 | \$0 | \$1.082 | 32% | |

Note that spot instances are not offered for the GovCloud (US). Pricing for spot instances in the AWS public clouds changes frequently. The two charts below show sample pricing for a c4.8xlarge instance and a c5.18xlarge instance in the US West (Oregon) region over a 1-month period. When the demand is lowest, one can get a deep discount using spot instances versus on-demand instances. The tricky part of the game is to predict when the demand will be low.

- Spot pricing for a c4.8xlarge instance



- Spot pricing for a c5.18xlarge instance



Cost of GPU instances

Current AWS Accelerated Computing instances [14] include Intel Xeon E5-2686 v4 (Broadwell) CPU and various GPUs. Pricing in the following chart is on a per-hour basis:

| Instance Type | CPU Model | # CPU Cores | CPU Memory | GPU Model | # of GPUs | GPU Memory | Oregon On-demand | Gov On-demand |
|---------------|---|-------------|------------|------------|-----------|------------|------------------|---------------|
| p3.2xlarge | Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz | 4 | 61 GiB | Tesla V100 | 1 | 16 GiB | \$3.06 | \$3.67 |
| p3.8xlarge | Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz | 16 | 244 GiB | Tesla V100 | 4 | 64 GiB | \$12.24 | \$14.69 |
| p3.16xlarge | Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz | 32 | 488 GiB | Tesla V100 | 8 | 128 GiB | \$24.48 | \$29.38 |
| p2.xlarge | Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz | 2 | 61 GiB | Tesla K80 | 1 | 12 GiB | \$0.90 | \$1.08 |
| p2.8xlarge | Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz | 16 | 488 GiB | Tesla K80 | 8 | 96 GiB | \$7.20 | \$8.64 |
| p2.16xlarge | Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz | 32 | 732 GiB | Tesla K80 | 16 | 192 GiB | \$14.40 | \$17.28 |
| g3.4xlarge | Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz | 8 | 122 GiB | Tesla M60 | 1 | 8 GiB | \$1.14 | \$1.32 |
| g3.8xlarge | Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz | 16 | 244 GiB | Tesla M60 | 2 | 16 GiB | \$2.28 | \$2.64 |
| g3.16xlarge | Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz | 32 | 488 GiB | Tesla M60 | 4 | 32 GiB | \$4.56 | \$5.28 |

Cost for storage space

Charge for storage is based on the amount allocated, not the amount actually used. Pricing is tiered; it is cheaper per gigabyte when more storage is allocated. If no long-term storage such as S3 is needed, the options are EBS and EFS:

EBS

This chart [15] from AWS shows the description, characteristics, and pricing of various EBS volume types.

Amazon EBS Volume Types

The following table shows use cases and performance characteristics of current generation EBS volumes:

| Volume Type | Solid State Drives (SSD) | | Hard Disk Drives (HDD) | |
|--------------------------------|---|--|--|--|
| | EBS Provisioned IOPS SSD (io1) | EBS General Purpose SSD (gp2)* | Throughput Optimized HDD (st1) | Cold HDD (sc1) |
| Short Description | Highest performance SSD volume designed for latency-sensitive transactional workloads | General Purpose SSD volume that balances price performance for a wide variety of transactional workloads | Low cost HDD volume designed for frequently accessed, throughput intensive workloads | Lowest cost HDD volume designed for less frequently accessed workloads |
| Use Cases | I/O-intensive NoSQL and relational databases | Boot volumes, low-latency interactive apps, dev & test | Big data, data warehouses, log processing | Colder data requiring fewer scans per day |
| API Name | io1 | gp2 | st1 | sc1 |
| Volume Size | 4 GB - 16 TB | 1 GB - 16 TB | 500 GB - 16 TB | 500 GB - 16 TB |
| Max IOPS**/Volume | 32,000 | 10,000 | 500 | 250 |
| Max Throughput/Volume | 500 MB/s | 160 MB/s | 500 MB/s | 250 MB/s |
| Max IOPS/Instance | 80,000 | 80,000 | 80,000 | 80,000 |
| Max Throughput/Instance | 1,750 MB/s | 1,750 MB/s | 1,750 MB/s | 1,750 MB/s |
| Price | \$0.125/GB-month \$0.065/provisioned IOPS | \$0.10/GB-month | \$0.045/GB-month | \$0.025/GB-month |
| Dominant Performance Attribute | IOPS | IOPS | MB/s | MB/s |

EFS

AWS EFS functions as a shared filesystem, which provides a common data source for workloads and applications running on more than one EC2 instance. EFS filesystems can be mounted on on-premises servers to migrate data over to EFS. This enables cloud-bursting scenarios.

With AWS EFS, a customer pays only for the amount of file system storage they use per month. There is no minimum fee and there are no set-up charges. There are no charges for

bandwidth or requests. Pricing for EFS storage in US West (Oregon) is \$0.30/GB-month [16]. There is also a charge of \$0.01/GB for syncing data to EFS.

Note: getting accurate usage accounting per user is going to be extremely difficult in a shared environment. Thus, the cost of storage space is probably best treated as an overhead of the operation.

Cost for transferring data

In most cases, AWS only charges for transfer data out of an AWS service.

AWS web sites only list the data transfer pricing [17] for the AWS GovCloud (US) region. This chart shows a sample pricing for transfer data from AWS S3 to internet:

| Amazon S3, Amazon SNS, Amazon SQS, Amazon SWF, Amazon DynamoDB | Price/GB |
|--|----------------------------|
| Data Transfer IN From | |
| All data transfer in | \$0.00 |
| Data Transfer OUT To | |
| Amazon EC2 into the AWS GovCloud (US) Region | \$0.00 |
| Another AWS Region or Amazon CloudFront | \$0.03 |
| Internet: | |
| • First 10 TB / month | \$0.155 |
| • Next 40 TB / month | \$0.115 |
| • Next 100 TB / month | \$0.090 |
| • Next 350 TB / month | \$0.065 |
| • Next 524 TB / month | contact us |
| • Next 4 PB / month | contact us |
| • Greater than 5 PB | contact us |

There are also costs for data transfers between EC2 instances, between AZs in the same region, and between different AWS regions [18].

The outbound data transfer is aggregated across multiple Amazon services such as Amazon EC2, Amazon S3, etc., and then charged at the outbound data transfer rate. This charge appears on the monthly statement as “AWS Data Transfer Out”.

Also, outbound data transfer costs are reduced when going over a direct connection into a customer site. There is the cost of having the direct connect [19] as well as a small charge of all data going over the direct connect. The benefit of using a direct connect is small in terms of cost but huge in terms of security. There is a 1-Gbps direct connection between NASA’s CSSO/EMCC and AWS.

The cost for out-bound data transfer is generally small enough that it is better included as overhead of the operation.

Appendix II – Penguin On Demand (POD)

Resource Overview

Penguin Computing offers on-demand public cloud (i.e., POD) and private cloud resources [20]. POD uses bare-metal (non-virtualized), InfiniBand- and OmniPath-based clusters in two locations, referred to as MT1 and MT2, which are accessible through a single POD Portal. Each location has its own localized storage. There are high-speed interconnects to facilitate easy migration of data from one location to another.

MT1 location provides Intel Westmere, Sandy Bridge, Haswell and Sandy Bridge+Nvidia K40 processors with InfiniBand QDR 40-Gbps interconnect and high-speed Network Attached Storage volumes. No storage quota is set by default.

MT2 location provides Intel Broadwell, Skylake (released on March 12, 2018), and KNL, all with Intel Omni-Path 100-Gbps interconnect and a Lustre filesystem. No storage quota is set by default.

POD also offers customized large filesystems such as GPFS if needed.

Different login nodes in each location can be created. Four choices of login nodes are available. POD also offers a Scyld Cloud Workstation, which is a remote, 3D-accelerated visualization solution for pre- and post-processing tasks on POD.

For the private cloud, the Penguin team will install and configure the cluster for the customer and take care of all operational and maintenance tasks. Such private clouds can be in the form of a portion of POD's public cloud on a monthly or yearly basis or be designed and hosted with a specific configuration tailored to meet the customer's needs.

Pricing

POD pricing information, except for their Intel KNL, is published online [21].

One free login node per location is allowed per management account (note that there could be multiple users under one management account) per POD location. Other types of login nodes are charged per server hour.

Login Node Pricing

| LOGIN NODE TYPE | CONFIGURATION | PRICE PER HOUR |
|-----------------|-------------------|----------------|
| pod.free | 1 core, 256MB RAM | FREE |
| pod.1x2 | 1 core, 2 GB RAM | \$0.09 |
| pod.2x4 | 2 core, 4 GB RAM | \$0.18 |
| pod.4x8 | 4 core, 8 GB RAM | \$0.36 |

A free login is provided for access to the compute cluster. Larger, more powerful login nodes can also be launched on-demand and charged by the second. Login nodes can be powered on/off to control usage and charges.

For most compute resources, charging based on per-core-hour is advertised. Usage of the Nvidia K40 resources is based on node hours.

Compute Pricing

| QUEUE | COMPUTE NODES | CORES/NODE | RAM/NODE | HIGH SPEED INTERCONNECT | STORAGE | HOUR |
|--------------|---|------------|----------|-------------------------|--------------------|---------------|
| Free Compute | 24 Cores for 5 Min Maximum Jobs | | | QDR InfiniBand | Private NAS Volume | FREE |
| S30 | Dual Intel(r) Xeon(r) Gold 6148 (Skylake) | 40 | 384 GB | Intel Omni-Path | Lustre File System | \$0.11 |
| B30 | Dual Intel® E5-2600v4 Series (Broadwell) | 28 | 256 GB | Intel Omni-Path | Lustre File System | \$0.10 |
| T30 | Dual Intel® E5-2600v3 Series (Haswell) | 20 | 128 GB | QDR InfiniBand | Private NAS Volume | \$0.09 |
| M40 | Dual Intel® X5600 Series (Westmere) | 12 | 48 GB | QDR InfiniBand | Private NAS Volume | \$0.08 |
| H30 | Dual Intel® E5-2600 Series (Sandy Bridge) | 16 | 64 GB | QDR InfiniBand | Private NAS Volume | \$0.08 |

| QUEUE | GPU NODES | CORES GPUS/NODE | RAM/NODE | HIGH SPEED INTERCONNECT | STORAGE | PRICE/GPU NODE HOUR** |
|-------|-----------------------------|-----------------|----------|-------------------------|--------------------|-----------------------|
| H30G | Dual NVIDIA® Tesla® K40 GPU | 2 GPUs/Node | 64 GB | QDR InfiniBand | Private NAS Volume | \$2.32 |

A user’s home directory comes with 1 GB of free storage. Storage beyond 1 GB is charged \$0.10 per GB-month. There is no charge for data transfer in and out of POD.

Storage Pricing

| STORAGE TYPE | DESCRIPTION | CHARGE SCHEME | PRICE/GB/MONTH |
|--------------------|---|-----------------------|----------------|
| Lustre File System | Parallel file system via 100 Gbps Intel® Omni-Path, N+1 redundant storage | Charged by Usage | \$0.10 |
| Private NAS Volume | Per-user volumes via 10 GigE network Distributed, N+1 redundant storage | Charged by Allocation | \$0.10 |

Storage is charged per GB, per month, and tracked daily.
All POD storage is accessed through high speed networks for best overall performance.

Technical support via pod@penguincomputing.com is free.

POD may offer discounts for government agencies, volume, long term contract and dedicated resources [22]; thus the published public prices of \$0.09, \$0.10, and \$0.11 per-core-hour for POD Haswell, Broadwell, and Skylake, respectively, and the \$0.10 per GB-month storage cost could be reduced.

Appendix III – Performance and Cost

Cost Basis

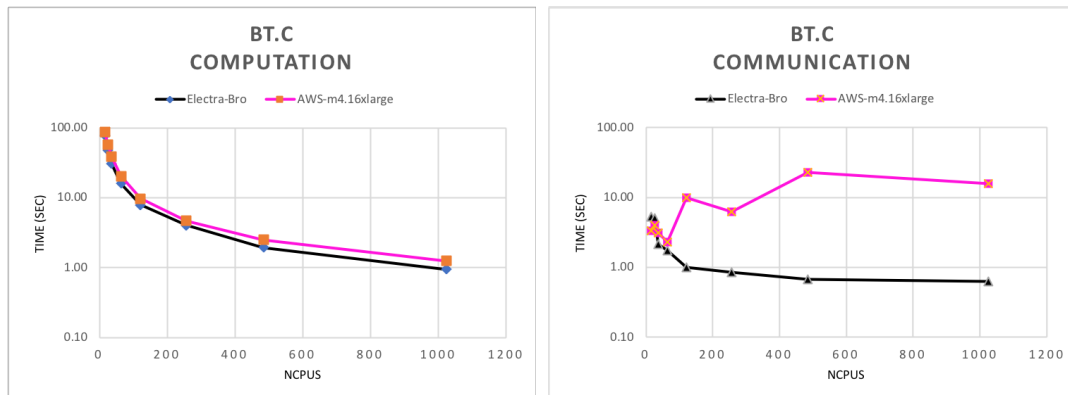
See Section 2 for the cost basis used for this study.

Performance and Cost of Workloads

NPB Scaling Study

The NPBs are commonly used to check the scaling performance of a system. Runs with the Broadwell processors offered by HECC (2.4 GHz, 28-core, 128 GiB, 56 Gbps), AWS (2.3 GHz, 32-core, 256 GiB, 10 Gbps) and POD (2.4 GHz, 28-core, 256 GiB, 100 Gbps) were performed. Since many of the NPB benchmarks require 2ⁿ MPI ranks and would fit nicely in a 32-core node, using the AWS 32-core Broadwell instances results in needing fewer AWS instances and possibly fewer occurrences of MPI inter-node communication than using the 28-core HECC or POD Broadwell nodes for some runs.

To examine the scaling performance, runs using the eight Class C micro-benchmarks (BT, CG, EP, FT, IS, LU, MG, and SP) from 16-CPU count up to 1,024-CPU count were performed on HECC and AWS. The timing of each run was separated into two components – computation and communication. The study’s results show that the scaling performance of the computation component, as represented by BT.C on the lower left graph, of both HECC and the AWS Broadwell is quite good:



On the contrary, the scaling performances of the communication component of HECC and AWS Broadwell (as shown on the right above) are drastically different. Figure 2 shows the communication performance for all NPB Class C benchmarks on HECC Broadwell and AWS Broadwell. With the only exception of the EP “embarrassingly parallel” benchmark, the AWS communication times are substantially higher. These results clearly demonstrate the superiority of the HECC Broadwell with 56-Gbps InfiniBand versus the AWS Broadwell with 25-Gbps network. Table 6 shows the Class C results in tabular form. The results for the Class D benchmarks for AWS in Table 7 show similar characteristics to the Class C results.

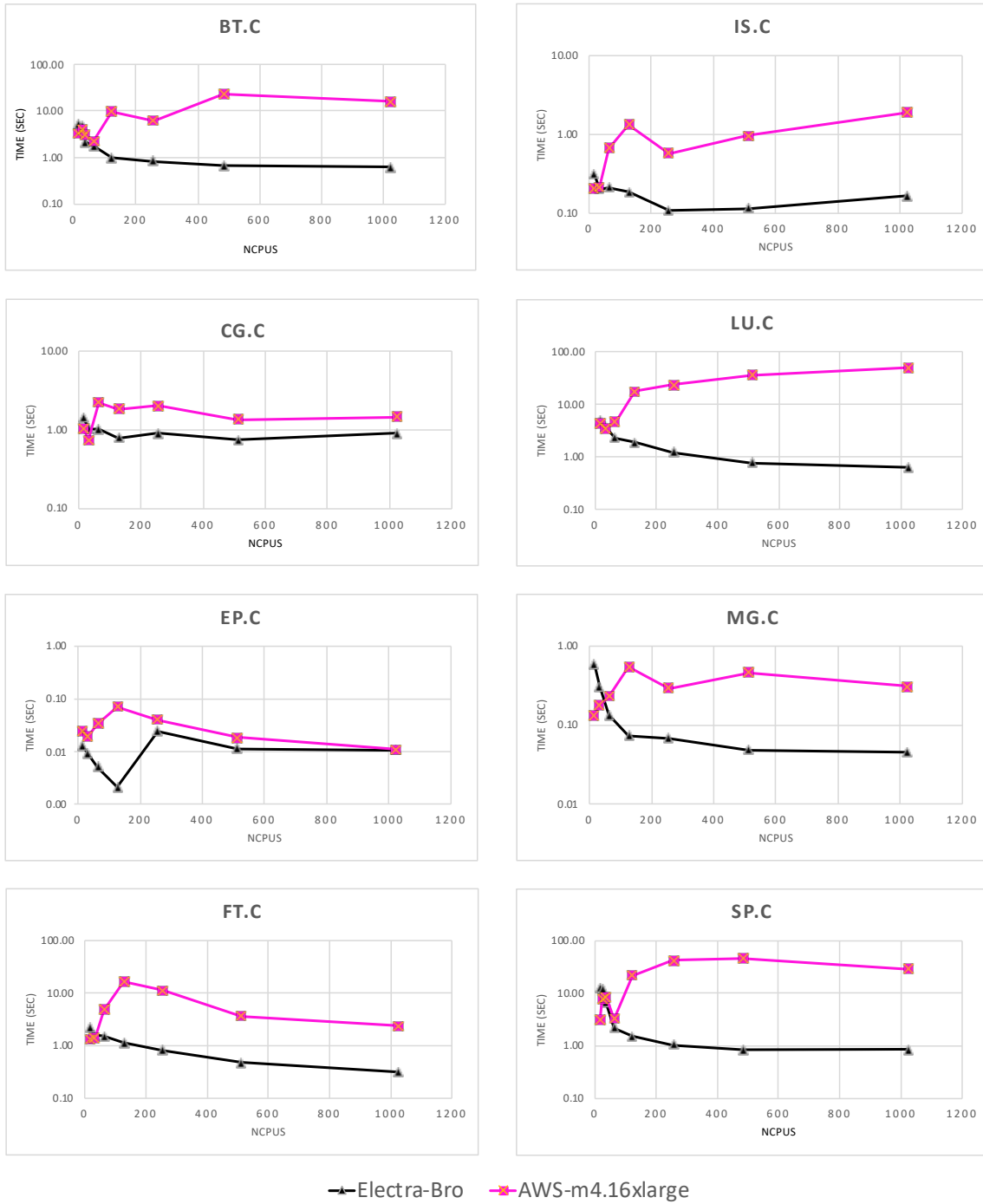


Figure 2: NPB Class C. Time spent in communication.

| Benchmark | NCPUS | # of HECC Broadwell nodes | # of AWS m4.16xlarge (Broadwell) instances | HECC time (sec) | AWS time (sec) |
|-----------|-------|---------------------------|--|-----------------|----------------|
| bt.C | 16 | 1 | 1 | 83.33 | 90.14 |
| bt.C | 25 | 1 | 1 | 52.52 | 61.13 |
| bt.C | 36 | 2 | 2 | 32.81 | 41.51 |
| bt.C | 64 | 3 | 2 | 17.66 | 22.72 |
| bt.C | 121 | 5 | 4 | 8.89 | 19.63 |
| bt.C | 256 | 10 | 8 | 4.87 | 10.86 |
| bt.C | 484 | 18 | 16 | 2.60 | 25.54 |
| bt.C | 1024 | 37 | 32 | 1.58 | 17.14 |
| cg.C | 16 | 1 | 1 | 20.62 | 21.73 |
| cg.C | 32 | 2 | 1 | 7.32 | 9.71 |
| cg.C | 64 | 3 | 2 | 4.37 | 6.87 |
| cg.C | 128 | 5 | 4 | 2.32 | 3.72 |
| cg.C | 256 | 10 | 8 | 1.45 | 2.71 |
| cg.C | 512 | 19 | 16 | 1.05 | 1.74 |
| cg.C | 1024 | 37 | 32 | 1.08 | 1.69 |
| ep.C | 16 | 1 | 1 | 5.54 | 6.11 |
| ep.C | 32 | 2 | 1 | 2.77 | 3.09 |
| ep.C | 64 | 3 | 2 | 1.38 | 1.57 |
| ep.C | 128 | 5 | 4 | 0.69 | 0.84 |
| ep.C | 256 | 10 | 8 | 0.38 | 0.43 |
| ep.C | 512 | 19 | 16 | 0.22 | 0.21 |
| ep.C | 1024 | 37 | 32 | 0.13 | 0.11 |
| ft.C | 16 | 1 | 1 | 17.80 | 16.54 |
| ft.C | 32 | 2 | 1 | 9.45 | 11.29 |
| ft.C | 64 | 3 | 2 | 5.35 | 9.81 |
| ft.C | 128 | 5 | 4 | 3.01 | 19.09 |
| ft.C | 256 | 10 | 8 | 1.79 | 12.28 |
| ft.C | 512 | 19 | 16 | 0.92 | 4.27 |
| ft.C | 1024 | 37 | 32 | 0.86 | 3.06 |
| is.C | 16 | 1 | 1 | 1.34 | 1.22 |
| is.C | 32 | 2 | 1 | 0.72 | 0.77 |
| is.C | 64 | 3 | 2 | 0.47 | 0.96 |
| is.C | 128 | 5 | 4 | 0.31 | 1.48 |
| is.C | 256 | 10 | 8 | 0.17 | 0.65 |
| is.C | 512 | 19 | 16 | 0.14 | 1.02 |
| is.C | 1024 | 37 | 32 | 0.19 | 1.93 |
| lu.C | 16 | 1 | 1 | 50.11 | 46.82 |
| lu.C | 32 | 2 | 1 | 27.07 | 32.66 |
| lu.C | 64 | 3 | 2 | 14.08 | 18.60 |
| lu.C | 128 | 5 | 4 | 8.06 | 24.44 |
| lu.C | 256 | 10 | 8 | 3.74 | 26.19 |
| lu.C | 512 | 19 | 16 | 1.93 | 37.26 |
| lu.C | 1024 | 37 | 32 | 1.20 | 50.81 |
| mg.C | 16 | 1 | 1 | 6.75 | 4.31 |
| mg.C | 32 | 2 | 1 | 3.20 | 4.19 |
| mg.C | 64 | 3 | 2 | 1.49 | 2.08 |
| mg.C | 128 | 5 | 4 | 0.79 | 1.46 |
| mg.C | 256 | 10 | 8 | 0.45 | 0.78 |
| mg.C | 512 | 19 | 16 | 0.22 | 0.67 |
| mg.C | 1024 | 37 | 32 | 0.13 | 0.39 |
| sp.C | 16 | 1 | 1 | 114.10 | 71.34 |
| sp.C | 25 | 1 | 1 | 64.84 | 62.23 |
| sp.C | 36 | 2 | 2 | 38.91 | 48.28 |
| sp.C | 64 | 3 | 2 | 17.92 | 23.38 |
| sp.C | 121 | 5 | 4 | 9.43 | 30.61 |
| sp.C | 256 | 10 | 8 | 4.93 | 47.47 |
| sp.C | 484 | 18 | 16 | 2.81 | 48.36 |
| sp.C | 1024 | 37 | 32 | 2.02 | 31.18 |

Table 6: Scaling Runs of NPB Class C on Broadwell.

On the POD Broadwell system, the story is somewhat different. The Class D results in Table 7 show that the POD Broadwell system with a 100-Gbps network performs quite well. For some benchmarks, such as FT.D and IS.D, POD performs better than HECC. For others, such as EP.D and LU.D, the reverse is true. It should be noted that of the two POD locations available for testing, the MT2 site offers resources with a 100-Gbps Omni-Path interconnect. The NPB Class D performance on POD MT2 Broadwell nodes are closer to performance on

| Benchmark | NCPUS | # of HECC or POD Broadwell nodes | # of AWS m4.16xlarge (Broadwell) instances | HECC time (sec) | HECC full cost | AWS time (sec) | AWS Oregon compute cost | AWS Gov compute cost | POD time (sec) | POD compute cost |
|---|-------|----------------------------------|--|-----------------|----------------|----------------|-------------------------|----------------------|----------------|------------------|
| bt.D | 256 | 10 | 8 | 104.03 | \$0.19 | 176.23 | \$1.25 | \$1.58 | 117.72 | \$0.92 |
| bt.D | 1024 | 37 | 32 | 31.34 | \$0.21 | 151.07 | \$4.30 | \$5.41 | 50.58 | \$1.46 |
| cg.D | 256 | 10 | 8 | 71.12 | \$0.13 | 230.79 | \$1.64 | \$2.07 | 56.96 | \$0.44 |
| cg.D | 512 | 19 | 16 | 44.98 | \$0.15 | 130.84 | \$1.86 | \$2.34 | 33.91 | \$0.50 |
| cg.D | 1024 | 37 | 32 | 45.18 | \$0.30 | 127.95 | \$3.64 | \$4.59 | 33.93 | \$0.98 |
| cg.D | 2048 | 74 | 64 | 30.73 | \$0.41 | 140.17 | \$7.97 | \$10.05 | 25.88 | \$1.49 |
| ep.D | 256 | 10 | 8 | 5.64 | \$0.01 | 6.86 | \$0.05 | \$0.06 | 6.32 | \$0.05 |
| ep.D | 512 | 19 | 16 | 2.82 | \$0.01 | 5.91 | \$0.08 | \$0.11 | 3.25 | \$0.05 |
| ep.D | 1024 | 37 | 32 | 1.43 | \$0.01 | 2.88 | \$0.08 | \$0.10 | 3.16 | \$0.09 |
| ep.D | 2048 | 74 | 64 | 0.73 | \$0.01 | 1.46 | \$0.08 | \$0.10 | 1.11 | \$0.06 |
| ft.D | 256 | 10 | 8 | 58.54 | \$0.11 | 357.7 | \$2.54 | \$3.20 | 38.52 | \$0.30 |
| ft.D | 512 | 19 | 16 | 36.46 | \$0.09 | 194.89 | \$2.77 | \$3.49 | 19.98 | \$0.30 |
| is.D | 256 | 10 | 8 | 6.67 | \$0.01 | 37.02 | \$0.26 | \$0.33 | 3.55 | \$0.03 |
| is.D | 512 | 19 | 16 | 4.22 | \$0.01 | 21.07 | \$0.30 | \$0.38 | 3.54 | \$0.05 |
| lu.D | 256 | 10 | 8 | 68.15 | \$0.12 | 135.26 | \$0.96 | \$1.21 | 75.07 | \$0.58 |
| lu.D | 512 | 19 | 16 | 40.32 | \$0.14 | 170.15 | \$2.42 | \$3.05 | 48.18 | \$0.71 |
| lu.D | 1024 | 37 | 32 | 23.46 | \$0.16 | 174.14 | \$4.95 | \$6.24 | 42.19 | \$1.21 |
| lu.D | 2048 | 74 | 64 | 15.37 | \$0.20 | 153.57 | \$8.74 | \$11.01 | 19.70 | \$1.13 |
| mg.D | 256 | 10 | 8 | 8.89 | \$0.02 | 22.16 | \$0.16 | \$0.20 | 8.93 | \$0.07 |
| mg.D | 512 | 19 | 16 | 4.47 | \$0.02 | 12.6 | \$0.18 | \$0.23 | 4.29 | \$0.06 |
| mg.D | 1024 | 37 | 32 | 2.81 | \$0.02 | 10.21 | \$0.29 | \$0.37 | 4.26 | \$0.12 |
| mg.D | 2048 | 74 | 64 | 1.42 | \$0.02 | 7.1 | \$0.40 | \$0.51 | 2.13 | \$0.12 |
| sp.D | 256 | 10 | 8 | 112.29 | \$0.20 | 283.02 | \$2.01 | \$2.54 | 117.81 | \$0.92 |
| sp.D | 1024 | 37 | 32 | 39.82 | \$0.26 | 272.82 | \$7.76 | \$9.78 | 53.73 | \$1.55 |
| Total Cost | | | | | \$2.81 | | \$54.72 | \$68.94 | | \$13.19 |
| Estimated AWS spot cost (30% of on-demand cost) | | | | | | | \$16.42 | | | |
| Estimated AWS pre-leasing cost (70% of US-Gov cost) | | | | | | | | \$48.26 | | |

Table 7: Scaling Runs of NPB Class D on Broadwell.

HECC’s Broadwell nodes, which have a 4x-FDR (56-Gbps) InfiniBand interconnect (see Table 7). An additional factor is that the HPE-MPT MPI library used on HECC has better scaling than the Intel-MPI on POD.

In addition to the timing comparison, Table 7 also includes the cost of running the benchmarks on Broadwell nodes. On the HECC Broadwell, the sum of the full-blown cost of each benchmark is \$2.81. This is ~4.7x cheaper than the compute-only cost on POD of \$13.19. Of the four AWS costs listed [including the (i) US West (Oregon) on-demand, (ii) US West (Oregon) spot price sampled as 30% of on-demand price, (iii) Gov on-demand, and (iv) Gov pre-leasing sampled as 70% of Gov on-demand], the lowest cost is the compute-only US West (Oregon) spot price of \$16.42, which is still 5.8x that of the HECC full-blown cost of \$2.81. The other three AWS costs are all even more expensive.

Further comparison was performed between the HECC Skylake with a 100-Gbps and the AWS Skylake with a 25-Gbps network. The performance and cost results using selected NPB class D benchmarks are detailed in Table 8. The lowest AWS Skylake total compute-only cost of \$18.29, based on a sample spot price, is about 12x more expensive than the \$1.50 HECC full-blown cost. Examined individually, some of the benchmarks such as CG.D and FT.D perform worse on AWS Skylake than on AWS Broadwell. This indicates that the new AWS Skylake installation may need some tuning or there was something wrong in the study’s runs.

In conclusion, this NPB study among HECC, AWS and POD provides a strong argument that the HECC systems not only provide good scaling performances but they are also more cost effective than existing AWS and POD offerings.

| Benchmark | NCPUS | # of HECC Skylake nodes | # of AWS c5.18xlarge (Skylake) instances | HECC time (sec) | HECC full cost | AWS time (sec) | AWS Oregon compute cost |
|--|-------|-------------------------|--|-----------------|----------------|----------------|-------------------------|
| bt.D | 256 | 7 | 8 | 79.83 | \$0.16 | 146.73 | \$1.00 |
| bt.D | 1024 | 26 | 29 | 20.52 | \$0.15 | 92.08 | \$2.27 |
| cg.D | 256 | 7 | 8 | 34.3 | \$0.07 | 614.2 | \$4.18 |
| cg.D | 512 | 13 | 15 | 17.24 | \$0.06 | 650.58 | \$8.29 |
| cg.D | 1024 | 26 | 29 | 13.51 | \$0.10 | 719.22 | \$17.73 |
| ep.D | 256 | 7 | 8 | 4.82 | \$0.01 | 4.93 | \$0.03 |
| ep.D | 512 | 13 | 15 | 2.44 | \$0.01 | 2.46 | \$0.03 |
| ep.D | 1024 | 26 | 29 | 1.24 | \$0.01 | 1.19 | \$0.03 |
| ft.D | 256 | 7 | 8 | 27.41 | \$0.05 | 526.08 | \$3.58 |
| ft.D | 512 | 13 | 15 | 14.84 | \$0.05 | 349.26 | \$4.45 |
| ft.D | 1024 | 26 | 29 | 8.05 | \$0.06 | 243.06 | \$5.99 |
| is.D | 256 | 7 | 8 | 2.64 | \$0.01 | 71.5 | \$0.49 |
| is.D | 512 | 13 | 15 | 1.45 | \$0.01 | 48.28 | \$0.62 |
| is.D | 1024 | 26 | 29 | 0.84 | \$0.01 | 44.72 | \$1.10 |
| lu.D | 256 | 7 | 8 | 57.58 | \$0.11 | 121.6 | \$0.83 |
| lu.D | 512 | 13 | 15 | 32.37 | \$0.12 | 106.79 | \$1.36 |
| lu.D | 1024 | 26 | 29 | 17.17 | \$0.13 | 99.3 | \$2.45 |
| mg.D | 256 | 7 | 8 | 8.18 | \$0.02 | 39.99 | \$0.27 |
| mg.D | 512 | 13 | 15 | 3.36 | \$0.01 | 15.56 | \$0.20 |
| mg.D | 1024 | 26 | 29 | 1.82 | \$0.01 | 16.44 | \$0.41 |
| sp.D | 256 | 7 | 8 | 101.52 | \$0.20 | 211.48 | \$1.44 |
| sp.D | 1024 | 26 | 29 | 20.32 | \$0.15 | 172.02 | \$4.24 |
| Total Cost | | | | | \$1.50 | | \$60.98 |
| Estimated AWS spot cost (30% of on-demand cost) | | | | | | | \$18.29 |

Table 8: Scaling Runs of NPB Class D on Skylake.

NTR1/SBU1/SBU2 Workload Performance and Cost Comparison

In addition to NPB, a performance and cost comparison was also made for a few full-sized applications, including ATHENA++, ECCO, Enzo, FVCore, NU-WRF, and OpenFOAM. Table 9 shows the results between HECC and AWS Haswell systems.

As shown in that table, the HECC full-blown cost was lower than the AWS Oregon compute-only on-demand cost for each of the applications tested. Even if one were able to get the low 30% compute-only spot price, the HECC full-blown cost is still cheaper. Using the sum, there is a cost ratio of 1.9x (\$141.77 AWS compute-only cost versus \$76.18 HECC full-blown cost).

Finding an AWS full-blown cost on a per-job basis is not possible. An estimate of the extra cost on top of the compute instances cost is done based on the reported total cost for the December 2017 AWS usage. Of the \$1,944 charge reported for December 2017, \$136 was

| Benchmark | Case | NCPUS | # of HECC Haswell nodes | # of AWS c4.8xlarge (Hasewell) instances | HECC time (sec) | HECC full cost | AWS time (sec) | AWS Oregon compute cost | AWS Gov compute cost |
|--|------------|-------|-------------------------|--|-----------------|----------------|----------------|-------------------------|----------------------|
| ATHENA++ | SBU2 | 1024 | 43 | 57 | 2268 | \$14.48 | 2298 | \$57.89 | \$69.68 |
| ATHENA++ | SBU2 | 2048 | 86 | 114 | 1177 | \$15.03 | 1374 | \$69.22 | \$83.32 |
| ECCO | NTR1 | 120 | 5 | 7 | 120 | \$0.09 | 173 | \$0.54 | \$0.64 |
| ECCO | NTR1 | 240 | 10 | 14 | 65 | \$0.10 | 140 | \$0.87 | \$1.04 |
| ENZO | SBU2 | 196 | 9 | 11 | 1827 | \$2.44 | 2266 | \$11.02 | \$13.26 |
| FVCore | SBU1 | 1176 | 49 | 66 | 1061 | \$7.72 | 1104 | \$32.20 | \$38.76 |
| nuWRF | SBU2 | 1700 | 71 | 95 | 529 | \$5.58 | 1302 | \$54.66 | \$65.80 |
| OpenFOAM | Channel395 | 48 | 2 | 3 | 4759 | \$1.41 | 7646 | \$10.14 | \$12.20 |
| OpenFOAM | Channel395 | 144 | 6 | 8 | 12547 | \$11.17 | 20771 | \$73.44 | \$88.39 |
| OpenFOAM | Channel395 | 288 | 12 | 16 | 10194 | \$18.16 | 23013 | \$162.73 | \$195.87 |
| Total Cost | | | | | | \$76.18 | | \$472.71 | \$568.96 |
| Estimated AWS spot cost (30% of on-demand cost) | | | | | | | | \$141.77 | |
| Estimated AWS pre-leasing cost (70% of US-gov cost) | | | | | | | | | \$398.27 |

Table 9: All Application Benchmark Results for HECC and AWS.

spent on disk space usage, \$738 on having the front-end available on a \$4.256 per hour basis, and less than \$2 on data transfer. The rest, \$1,068, was spent on using various compute instances. Therefore, there is ~82% ($1944/1068=1.82$) extra cost on top of the compute instance cost. Similarly, for the January 2018 AWS usage, of the \$1,654 charge, \$137 was spent on disk space usage, \$665 on the front-end, \$12 on data transfer, and \$840 on compute instances. Thus, there was a ~97% ($1654/840=1.97$) extra cost on top of the compute instances cost. There is also the additional OSSC/EMCC overhead, which has not yet been reported and thus not included in this calculation.

For comparison between HECC (with HPE MPT) and POD (with Intel MPI), only Enzo and WRF were tested. The results are summarized in Table 5 of Section 3. The compute-only costs on POD were 5.3x higher than the full-blown HECC costs. Note that the storage cost incurred on POD were minimal during this evaluation period. Given that the cost of POD login node and file transfer will likely be free and there will be volume discount with storage, it is expected that the extra cost on top of compute in a production environment will be smaller with POD than with AWS.

In conclusion, running the MPI workloads on AWS or POD are not cost effective compared to running them on the HECC systems.

Cost Comparison: Preleasing versus On-Premises for 1-Node Jobs

Section 4 raised the possibility that there could be a cost advantage in moving some one-node jobs to the cloud. To compare the annual cost of using in-house resources versus commercial cloud offerings for some of the 1-node jobs, a resource of 144 nodes was assumed.

Cost of using HECC in-house resources

For existing HECC resources, the full-blown cost is based on an SBU rate of \$0.16 and a SBU factor. The following table summarized the annual cost with 144 nodes for 3 processor types:

| Processor Type | SBU factor | Annual cost |
|----------------|------------|-------------|
| Haswell | 3.34 | \$674,114 |
| Broadwell | 4.04 | \$815,395 |
| Skylake | 6.36 | \$1,283,641 |

Cost for pre-leasing 144 AWS compute instances, 1 front-end and 100-TB storage

The c4.8xlarge instances (with 18 cores and 60 GiB of memory per instance) are used in the cost estimate. Other types of instances, such as c5.18xlarge or m4.16xlarge, will cost more. The cost of data transfer is not counted.

The compute-only cost varies among the three regions:

- Cost for leasing instances from the GovCloud(US) region:

1 instance:

No Upfront: $\$883.30 \times 12 \text{ months} = \$10,599.60$

Partial Upfront: $\$5,055 + \$421.21 \times 12 \text{ months} = \$10,109.52$

All Upfront: $\$9,907$

With 144 instances for 1 year All Upfront, the cost is \$1,426,608.

- Cost for leasing instances from the US West (N. California) region:

1 instance:

No Upfront: $\$993.53 \times 12 \text{ months} = \$11,922.36$

Partial Upfront: $\$5,690 + \$474.14 \times 12 \text{ months} = \$11,379.68$

All Upfront: $\$11,152$

With 144 instances for 1 year All Upfront, the cost is \$1,605,888.

- Cost for leasing instances from the US West (Oregon) region:

1 instance:

No Upfront: $\$735.84 \times 12 \text{ months} = \$8,830.08$

Partial Upfront: $\$4,231 + \$352.59 \times 12 \text{ months} = \$8,462.08$

All Upfront: $\$8,293$

With 144 instances for 1 year All Upfront, the cost is \$1,194,192.

Cost for 1 front-end running

In the test environment, an r4.16xlarge on-demand instance (E5-2686v4 @ 2.3 GHz, with 32 cores and 488 GiB of memory and 25-Gbps interconnect) is used as a front-end. Such an instance will cost \$4.256 per hour in the US West (Oregon) region.

To have a front-end running for 1 year will cost $\$4.256 \times 365 \times 24 = \$37,282.56$

Cost for storage space

Charge for storage is based on the amount allocated, not the amount actually used.

- AWS EBS gp2 volumes are used in the test environment. A capacity of 100 TB using gp2 volumes for a production environment will cost:

Sample charge for provisioning 100,000 GB (100 TB) of EBS gp2 volumes

for 1 month $\$0.10 \times 100,000 \text{ GB} = \$10,000/\text{month}$.

for 1 year $\$0.10 \times 100,000 \text{ GB} \times 12 \text{ months} = \$120,000/\text{year}$

- If EFS is used instead of EBS, the cost is:

Sample charge for storing 100 TB of data on EFS:

$\$0.30/\text{GB-month} \times 100,000 \text{ GB} = \$30,000/\text{month}$

$\$30,000/\text{month} \times 12 \text{ months} = \$360,000/\text{year}$

Total Cost

The minimum annual cost of the compute, front-end and storage combined will be $\$1,194,192 + \$37,282 + \$120,000 = \$1,351,474$.

Cost for pre-leasing 144 POD compute instances, 1 login-node and 100 TB storage

If the use of a free login node is adequate for the 1-node job workload, the cost will include just the 144 compute nodes and the storage. The POD Haswell nodes (20 cores and 128 GiB of memory per node) will be used in the cost estimate. Possible discounts (government, volume, long term contract or dedicated resources) for both compute and storage resources are not reflected in the estimate.

- Cost for pre-leasing 144 Haswell nodes for 1 year

- 1 node: $20 \text{ cores/node} \times \$0.09 \text{ core-hour} \times 24 \times 365 = \$15,768$
144 nodes: $\$15,768 \times 144 = \$2,270,592$
- Cost for allocating 100 TB storage for 1 year
 $\$0.10 \text{ GB-month} \times 100,000 \text{ GB} \times 12 = \$120,000$

The total cost of the compute and storage combined will be $\$2,270,592 + \$120,000 = \$2,390,592$.

Appendix IV – Usability Considerations

This section documents the study's experiences, besides performance and cost, in using AWS and POD.

Account Creation

AWS – Accounts for HECC staff who participated in the testing were created by an HECC staff member who has been acting as a support staff for CSSO/EMCC's AWS cloud usage. The SSH public key of each user was copied from a Pleiades front-end node (PFE) to AWS to facilitate direct login from a PFE to an AWS front-end instance (*nasfe01*). In the event that accounts for many users are to be created, the administrators supporting NASA AWS should be able to handle it without user involvement.

POD – Through the POD web portal, one can request an individual account and be responsible for the cost. One can also get an account through an invitation by an existing user. In that case, the usage by the invitee will be charged to the inviter. After an account is created (need an email address and choose a 14 to 64 characters long password), one has to upload an SSH public key for use on MT1 and/or MT2 and choose the storage and login node preferences through the web portal. In the event that there are many users from a site, POD staff can take care of sending out an email invitation to the individual users if an Excel spreadsheet with users' email addresses is provided. POD staff can provide a free training to users on how to use their web portal to create password and upload their SSH public keys.

Front-end

AWS – The *nasfe01* front-end set up for this testing incurs a \$4.256/hr. charge. To reduce cost, there is regular scheduled downtime for *nasfe01*. Restarting it can be done through a web control interface: <https://gpmcepubweb.aws.nasa.gov/> using a username and password.

POD – Each management account can create its own login node. On MT1, a basic and free login node has 1 core and 256 MiB of memory. On MT2, the free login node has 1 core and 2 GiB of memory. Other larger login nodes will cost money to prevent abuse. In the event that many users under a management account need a large login node, POD may be able to offer it for free [22].

Connection to either an AWS front-end or a POD login node from a user's desktop or a Pleiades front-end is through authentication with user's SSH public key. The study's benchmarkers had an issue with SSH from HECC PFE nodes to our MT2 free-login node where the ssh would hang, but no issue to the MT1 free-login node. It was resolved within a few days by "forcing the MTU to 1500 on POD's uplinks to make sure that anything leaving POD's network has a max MTU of 1500". Similarly, POD staff resolved another issue with scp between our PFE to the MT2 login node by adjusting some network settings.

Filesystem

AWS – AWS offers many storage choices – EBS, EFS, S3, Glacier, etc. for various needs. /home and /nobackup using EBS volumes are set up by an HECC staff for this study. Storage cost is based on the size of filesystem allocated for HECC, not the amount actually used. In production, hourly usage for each user will be recorded and that can be used to allocate the overall storage costs if necessary.

POD – POD offers NAS storage on MT1 and Lustre on MT2. Storage is charged \$0.10 per GB per month. Volume discount is available [22]. POD can also provide customized large filesystems if needed.

Data Transfer

AWS – SSH from PFE to *nasfe01* works fine and there is no issue with data transfer. Data transfer from PFE to AWS is free and is currently using CSSO/EMCC's 1-Gbps direct connect. Outbound transfer from AWS is not free. Cost for individual outbound transfer is not tracked by CSSO/EMCC. However, cost of outbound transfer generally is small enough and CSSO/EMCC includes it in the overhead cost. In the event that the data transfer cost is not small, it will need to be written off as HECC overhead or evenly split across all users.

POD – With proper SSH public key, one can scp from PFE to the MT1 and MT2 login nodes. There is no cost for data transfer.

Operating System

AWS – In the testing environment, Amazon Linux was chosen for testing. Other operating systems such as CentOS, Ubuntu, Rhel, SLES, Windows are also supported by AWS but they will cost more.

POD – MT1 and MT2 are fixed with CentOS 6 and CentOS 7, respectively.

Software Stack

AWS – The test environment does not have any software packages pre-installed. Any required software packages for testing, such as Intel compiler, netcdf, hdf4/5, Intel MPI, OpenMPI, Metis and Parmetis, were installed by an HECC staff.

POD – More than 250 commercial and 3rd party software packages have been pre-installed on the MT1 and MT2 clusters. There are small differences in what's available on MT1 and MT2. POD technical support team will help with installing additional software packages upon request.

For commercial software packages (including Intel compilers and MPI), either HECC or the individual users will have to provide licenses for use on either AWS or POD.

Batch Job Management

AWS

- The use of the `spot_manager` script, created by an HECC staff, to (i) check pricing and what resources are available, (ii) request/use instances, (iii) terminate instances, and (iv) check cost for a run, is a temporary solution for staff testing on the AWS cloud. Regular HECC users will have difficulties using the AWS cloud lacking a PBS-like job scheduler. There is no jobid associated with a 'job'.

POD

- The use of the PBS-like Moab scheduler and Torque resource manager provides a familiar environment to submit, monitor and terminate batch jobs with the `qsub`, `qstat`, and `qdel` commands. Each job has a jobid. POD-provided utility `podstatus` shows what resources are not being used and `podstart` provides an estimate of when a submitted job will start running. Below is a summary of available queues. The B30, S30 and IntelKNL queues are for MT2 and the rest are for MT1.

| Queue | Compute Nodes | Cores/Node | RAM/Node |
|-------|------------------------------|------------|----------|
| FREE | Free 5 minute, 24 core jobs | 12 | 48GB |
| M40 | 2.9GHz Intel Westmere | 12 | 48GB |
| H30 | 2.6GHz Intel Sandy Bridge | 16 | 64GB |
| T30 | 2.6GHz Intel Haswell | 20 | 128GB |
| H30G | H30 with two NVIDIA K40 GPUs | 2 | 64GB |
| S30 | 2.4GHz Intel Skylake | 40 | 384GB |
| B30 | 2.4GHz Intel Broadwell | 28 | 256GB |

IntelKNL 1.3GHz Intel Xeon Phi 256 112GB

- Although POD advertised a per-core charge for all processor types except for H30G, our attempt to request fewer cores than maximum per node failed for multi-node jobs. For example, for the Enzo 196 test case, we tried to request 14 cores out of the 28-core Broadwell nodes and got an error. This has been confirmed with POD support as the intended behavior.

```
#PBS -l nodes=14:ppn=14
```

```
qsub error:
```

```
ERROR:
```

```
* ERROR: Multinode jobs in B30 must be ppn=28 (full node)
```

- For 1-node jobs, we verified that requesting fewer cores than the maximum in a node is allowed and charging is based on the number of cores requested.

Ease of Getting the Resources

AWS – It is difficult to get the new Skylake instances without paying the on-demand price.

POD – Compute resources in most queues are frequently 100% taken. The Haswell nodes seem to be more likely to be available. Access to the Skylake nodes was given on March 5, one week before the public release data of March 12, 2018.

Porting Experience

Porting MPI applications from HECC to either AWS or POD may experience difficulties in three different ways: (i) compiling/linking issues with lots of dependencies of packages, (ii) failing to run, and (iii) poor performance for unknown causes. Debugging each of these issues will be non-trivial and puts a significant burden on the HECC support staff or the users. Below are some examples.

AWS

- Porting GEOS-5 (one of the SBU-2) to AWS was difficult due to lots of dependencies. The HPE MPT used at HECC is not available at AWS cloud. Instead, one has to rebuild with Intel-MPI. This will prevent a seamless migration of MPI applications to the cloud.
- Even for MPI applications that overcome the porting issues with Intel-MPI on AWS cloud, there were cases where some applications (such as GEOS-5) would fail with Intel MPI error. Some applications (such as ATHENA++) would run with some core-count cases while failed with other core-count cases. For example, using the AWS c4.8xlarge instances, the 1024-core case of ATHENA++ consistently runs fine while the 512-core and 2048-core cases would sometimes fail with an Intel-MPI error in socksm.c. On the other hand, using AWS c5.18xlarge instances, ATHENA++ runs with 512-core but fails with 1024 and 2048-core counts.
- For some applications that ran properly, their performances were not great. One example is cg.D and ft.D on AWS Skylake. Another example is WRF on AWS Haswell and Broadwell (not attempted on Skylake), which would take a few days to complete instead of 1 – 2 hours in our earlier runs. After experimenting with mpiexec options, we were able to reduce the WRF runtime on AWS Skylake significantly, but it still took more than 2x longer than the run on HECC's Skylake.

POD

- The fact that POD's software stack already includes many familiar commercial and third-party software packages helps to reduce the amount of work needed during a porting process. For example, POD has provided ready-to-use WRF and OpenFOAM modules with OpenMPI. Some issues we encountered are:

- POD-provided OpenFOAM module with OpenMPI does not provide good performance for our test case. Performance analysis with tools will be needed to understand this behavior.
- Failure in building WRF with Intel-MPI ourselves on POD MT2 system but not on MT1. With the help from POD technical staff, this issue was resolved by changing some hard-coded cpp commands.

State-of-the-Art Architecture

AWS – The new Skylake processors were released in early Nov, 2017. Nvidia Tesla GPGPU processors K80, M60 and V100 are available.

POD – POD lags behind AWS in providing state-of-the-art architectures. The Skylake processors were released for public on March 12, 2018. Intel KNL and Nvidia Tesla K40 are available, but POD currently has no time frame on offering newer Nvidia GPGPU processors.

Cloud Bursting Readiness

The AWS and POD testing systems used for this study do not have a mechanism in place yet to allow a seamless migration of jobs from a datacenter to their clouds. There are a few vendors who have solutions for cloud bursting. For example, Altair PBS has cloud-bursting beta testing with Microsoft Azure. Adaptive Computing advertises availability of their Moab/NODUS cloud bursting solution on AWS, Google cloud, AliCloud, Digital Ocean and others [23]. When this technology matures, there will be many issues to be worked out on HECC and the cloud end. In the meantime, a job can be migrated manually if it is set up properly to run under both the HECC and the cloud environment. For example, the NASA NEX group has started using Docker to build “containers” for their jobs that can be run on either HECC or AWS. However, building such containers on a system requires root privileges. Granting root access to general users on HECC systems for building the containers is impossible due to security concerns.

Technical Support

AWS – CSSO/EMCC has a support contract with AWS. We do not know the cost of this contract.

POD – Technical support is available 6 AM to 5 PM PDT. Request for help is by email to pod@penguincomputing.com. There is also an online support portal (https://penguincomputing.my.salesforce.com/secur/login_portal.jsp?orgId=00D300000000y1I&portalId=0605000000D4C9) where one can open and track support cases.

Usage/Cost Tracking System

AWS – With the test environment, after stopping a job, the `spot_manager` script returns the compute-cost associated with the run. An HECC staff also has access to historical data where he can look up the cost based on the date/time and instance type. There is also a monthly cost report where the total charge is separated into compute instances, front-end, storage, etc. Note that the monthly charge does not include the CSSO/EMCC overhead and AWS support contract.

POD – Its web portal provides “My Account Usage” for each user. There are downloadable excel spreadsheets to show usage by job, usage by day, user summary, group summary, and project summary. For users under a managed account, by default, the per-job cost is only available to the person who owns the managed account.

Authorization to Operate (ATO)

AWS – This testing was done on AWS public cloud under the ATO of CSSO/EMCC. EMCC also has resources on AWS GovCloud which in principal allows processing ITAR/EAR99 data. However, CSSO/EMCC project manager does not yet allow anyone to run ITAR/EAR99 on AWS except for tightly controlled situations. If HECC were to use AWS outside of CSSO/EMCC, a separate ATO between HECC Security Lead and NASA HQ would be needed.

POD – This testing was done with a “proof of concept” arrangement without cost. No ATO has been filed by CSSO/EMCC or HECC for using POD for production. POD is SSAE (Statement on Standards for Attestation Engagements) SOC1 and SOC2 compliant.

Sales Channel

AWS – Government agencies have many options for buying AWS cloud services, either direct from AWS or through a range of contract vehicles [24]. NASA’s Solutions for Enterprise-Wide Procurement (SWEP) works with multiple AWS resellers to provide services to federal agencies. For the CSSO/EMCC AWS, the reseller used is Four Points Technology.

POD – Sales are handled by POD’s director of Sales and Federal Accounting Rep.