A Performance and Cost Analysis of the Amazon Elastic Compute Cloud (EC2) Cluster Compute Instance

Michael Fenn (mfenn@psu.edu),
Jason Holmes (jholmes@psu.edu),
Jeffrey Nucciarone (nucci@psu.edu)
Research Computing and Cyberinfrastructure Group, Penn State University.

Introduction:

Amazon recently announced the availability of Elastic Compute Cloud (EC2) Cluster Compute Instances specifically designed for high performance computing (HPC) applications. The instance type, Cluster Compute Quadruple Extra Large, is a 64-bit platform running either Windows or Linux, has 23GB memory, dual quad-core Intel Nehalem processors, 10 Gigabit Ethernet I/O, and 1,690 GB of storage. This instance type is available in Amazon's U.S. – Northern Virginia Region data center. Amazon worked closely with researchers at the Lawrence Berkeley National Laboratory in the development of this instance type. The current charge for this node type is $1.60 per node per hour. An 880 instance running LINPACK achieved 41.82 TFLOPS. With this much cluster compute power available on demand the question arises if cloud computing with using and Amazon EC2 HPC instance type can meet the HPC demands of an individual researcher. This paper seeks to determine the feasibility of utilizing Amazon EC2 clusters for real-world workloads and compare the result with a large University Based Computer Center (UBCC).

Amazon Elastic Compute Cloud Concepts:

The Amazon Elastic Compute Cloud (EC2) is service provided by Amazon.com whereby users can obtain access to virtualized computational resources hosted in one of many Amazon datacenters. Users are given low-level access to a virtual machine (VM) where they can install software, run jobs, etc.

Fundamental EC2 concepts include the *Amazon Machine Image (AMI),* the *EC2 instance* and the *Elastic Block Store (EBS) volume*.

- An AMI is an immutable representation of a set of disks that contain an operating system, user applications, and/or data.
- An EBS volume is a read/write disk that can be created from an AMI and mounted by an instance.
- An EC2 instance is a virtual machine running on Amazon's physical hardware.

Since AMIs are immutable, they are copied into Elastic Block Store (EBS) volumes (one per disk) upon starting an instance. Once created, the EBS volume is independent of the originating AMI and a *snapshot* of the EBS volumes can be created and *bundled* into a new AMI. The independence of the AMI and EBS volume also allows for multiple instances to be started from a single AMI. The EC2 control panel also exposes which *availability zone* into which a particular instance or EBS volume has been placed. Instances and volumes placed in the same zone have higher performance (more bandwidth and less latency) than instances and volumes placed in separate zones.

Amazon's base AMI for High Performance Computing (HPC) is a standard CentOS 5.4 install with only a very minimal set of packages. An HPC user is expected to install the compilers, libraries, and other tools necessary to create a functioning HPC cluster. This creates a substantial barrier to entry for a new user who is not well-versed in the creation and maintenance of an HPC cluster. This can be mitigated somewhat by the ability to share AMIs among EC2 users. A possible opportunity exists whereby a knowledge user or organization can create a set of fully-featured AMIs and then share those with less knowledgeable users.

Configuration:

Our initial goal was to determine the minimum necessary configuration in order run a code representative of a typical application on an existing RCC cluster. We began with the EC2 Cluster Compute Quadruple Extra Large instance for HPC, starting with a single 8 core instance to minimize startup complexities and to learn what a typical user would encounter when beginning to use this service. This instance comes minimally configured and requires the user to apply all necessary updates, patches, tools, and software.

Our first startup day was spent simply obtaining and configuring an instance and then to stop and wait until a billing record was generated so that we examine typical 'startup' expenses.To that end, the base CentOS 5.4 image was updated and various necessary tools and compilers installed so that we could run a typical HPC workload. This included GNU C and Fortran compilers and the OpenMPI library. The workload we wished to run required the PGI compilers and tools be installed. As these are licensed products and not open source we initially installed a demo license for the duration of the experiment.

Unfortunately each start-up of the EC2 instance changed the virtual machine's identification that the trial license used as part of its authentication process. This created an unacceptable situation so we then purchased the Amazon Elastic IP service which allows us to define a fixed IP address associated with the image. This allowed us to open a fixed port on the RCC license manager firewall and use one of our existing licenses for the PGI compiler suite.

Our next steps were to configure the instance with all the required prerequisite libraries required by our workload code. This experiment required the use of the NETCDF library so this was also downloaded and built. We next downloaded and built our workload code, then pausing to wait until the next billing cycle to estimate what a typical user would experience getting an instance ready just to run their code. At this time we took a snapshot of the node, creating an AMI for future use, helping to amortize these start-up costs. An EBS volume was attached to the instance and mounted as work partition because of the increased performance of EBS over the root filesystem.

Due to time restrictions, a multi-node run has not yet been attempted, but starting multiple fully-configured should not be an issue since this is a native capability of Amazon's AMI infrastructure. A master node will also be needed in order to provide a shared filesystem for a multi-node run.

We selected the RCC operated Cyberstar cluster as our performance reference. The EC2 instance is very close to that of Cyberstar: node memory is approximately the same (23 GB on EC2 vs. 24 GB on Cyberstar) and the processor speeds are only two bins apart (2.93 on EC2 vs. 2.67 GHz on Cyberstar). The Cyberstar cluster is operated by the RCC group and is funded and supported by the Major Research Instrumentation Program from the National Science Foundation through Award Number #0821527 and several Penn State units including the Office of the Sr. Vice President for Research, the Institute for CyberScience, the Huck Institutes of the Life Sciences, the Materials Research Institute, the College of Engineering, the Eberly College of Science, the College of Earth and Mineral Sciences, and Information Technology Services.

WRF Model Description:

The model chosen for this study is the Weather Research and Forecasting (WRF) system's Advanced Research WRF (ARW) version 2.2.1 (Skamarock et al. 2005). Meteorological models often are significant contributors to errors in atmospheric transport and dispersion (AT&D) predictions. Wind errors can be especially large in the nocturnal stable boundary layer (SBL). Because turbulence tends to be so weak in the shallow nocturnal SBL, compared to deep convective boundary layers, these cases are much more likely to exhibit poor dispersion characteristics, thus maintaining high concentrations of airborne contaminants for many hours. The example research production run used in this benchmark study came from actual research conducted at Penn State using RCC systems. The research continued recent DTRA-sponsored numerical research at PSU investigating SBL predictability at very fine mesoscale resolutions.

To study the evolution of SBL flows, ARW is configured with four nested domains, each having a one-way interface with the next smaller grid. The finest domain covers ~67 X 67 km, has a horizontal resolution of 444 m and is centered over the Nittany Valley of central PA. This region is dominated by narrow quasi-parallel ridges oriented southwest-to-northeast, which flank broad valleys, with the Allegheny Mountains located in the northwest part of the domain. The 1.333-km domain covers ~256 X 224 km, encompassing almost the entire Allegheny Mt. region, but it resolves the narrow ridge-and-valley topography of Central PA with lesser fidelity. Figure 1 is a representative image of the highest resolution grid.

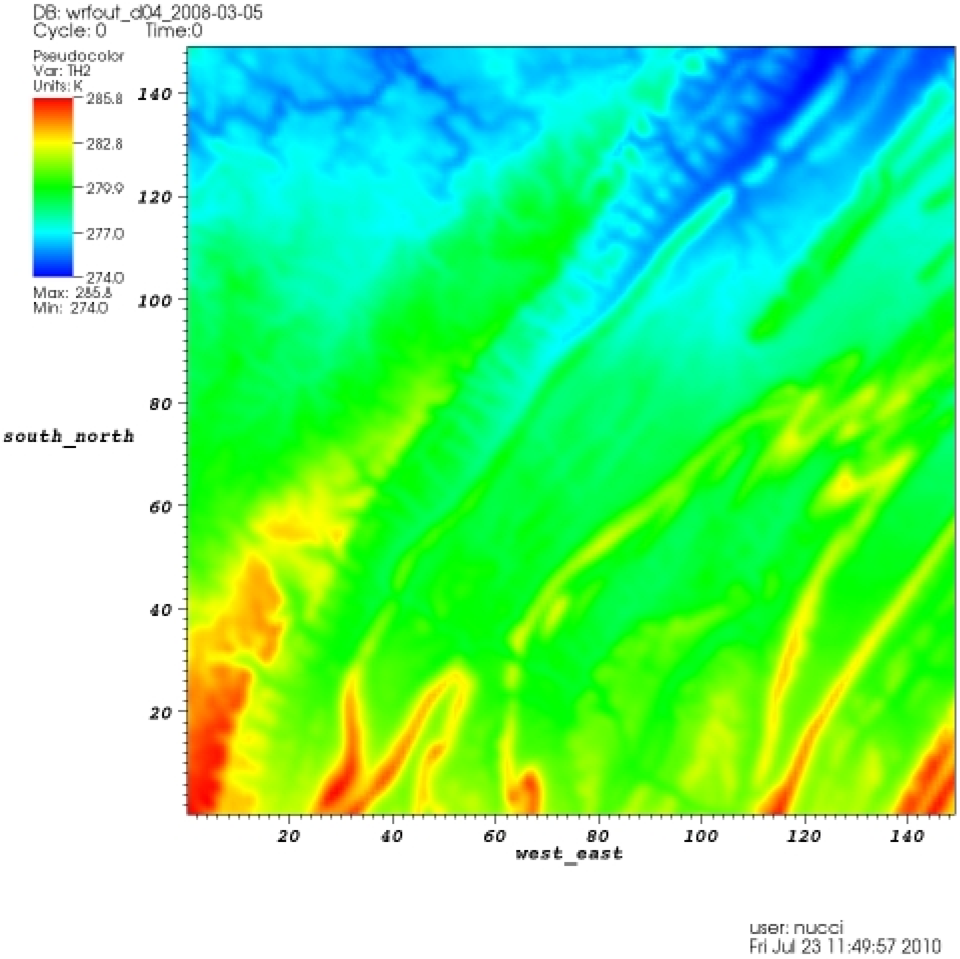

Figure 1: WRF output for March 5, 2008 benchmark model run, centered over the Nittany Valley in PA, using a 444m grid resolution. Field displayed is potential temperature at the 2-m level above the surface

This code is representative of parallel applications running on the RCC clusters. We have run extensive benchmarks from 8 - 96 cores using various hardware and runtime configurations. This provides a reasonable baseline performance against which we will measure EC2 performance. Figure 2 is representative of WRF performance for 2.67 GHz Intel Nehalem based nodes when using 2 cores per 8 core node, from 8 to 96 cores. The EC instance we used for this study has 2.93 GHz Intel Nehalem processors so barring extraordinary virtualization overhead we would reasonably expect EC2 performance to be close.

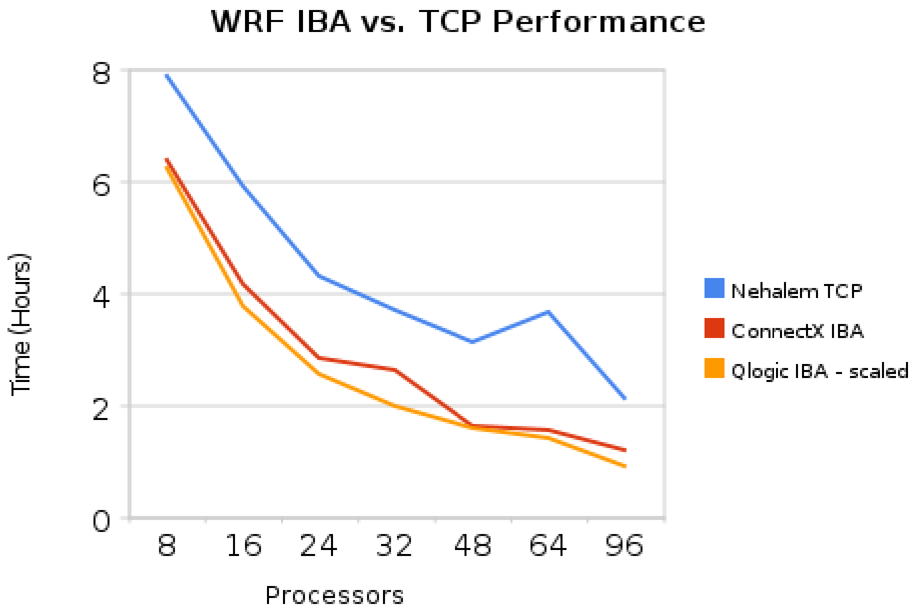

Figure 2: WRF performance using the Intel Nehalem processor at 2.67 GHz using 2 cores per 8 core node.

Results:

The benchmark described above was run on both Cyberstar and EC2 Cluster Compute Quadruple Extra Large (cc1.4xlarge) nodes. The Cyberstar nodes have two quad-core Intel Xeon X5550 processors running at 2.67GHz with 24GB of RAM. The EC2 nodes are VMs running on a physical system with two quad-core Intel Xeon X5570 processors running at 2.80GHz with 23GB of RAM available to the VM.

An initial run on the EC2 instance had a run time of 684 minutes, 52% slower than a corresponding node a the Cyberstar duster. The EC2 node has hyper-threading enabled, which caused multiple processes to be mapped to the same physical core, thus lowering performance. Most HPC systems, including Cyberstar, are run with hyper-threading disabled to prevent these types of problems. An additional run with an explicit processor / core mapping supplied to the MPI run command eliminated this problem and resulted in a more reasonable run time as presented in Table 1.

Table 1: WRF Performance

Benchmark	Cyberstar	cc1.4xlarge	Cyberstar Advantage
1 node, 8 processes per node (minutes)	448	584	**30.3%**

Despite the EC2 instance being two processor speed bins faster than Cyberstar (2.93 vs. 2.67 GHz) virtualization overhead that is present on the EC2 node resulted in a 30.3% performance degradation.

Wait time to create a single 8 core instance was very low, usually less than a minute for a resource to be obtained in the requested zone. Instances are requested in the same zone as the EBS volume in order to minimize i/o performance issues. Boot time was usually several minutes. Amazon charges for the EC2 node instance only when the instance is running and is not based on whether the CPU is active or not, so it is important to monitor and disable the instance once the application code is complete to avoid being charged for an idle instance. Snapshots of machine images took upwards of 40 minutes but was only needed when changes needed to be persisted into a new AMI. By contrast a 30 day average queue wait time on the Cyberstar cluster for an equivalent node was 24 minutes. Even when factoring in this additional wait time Cyberstar still has a 25% performance advantage, 472 vs. 584 minutes.

Costs:

Amazon EC2 utilizes a “pay as you go” model wherein a user’s usage is measured on a variety of metrics such as instance-hours, size of provisioned EBS volumes, I/O requests to EBS volumes, requests to save/load snapshots, Elastic (i.e. static) IP usage, data transfer into/ out of EC2, etc. Each of these metrics are assigned a cost and a billing report is generated summarizing a user’s costs.

Table 2: Usage over the period 7/20/10-7/22/10

	7/20/10	7/21/10	7/22/10	7/23/10
Instance-hours	5	6	13	12
EBS GB-month	1.909	4.322	4.774	4.775
I/O requests to/ from EBS	183,407	420,765	648,458	817,721

Table 3: Costs over the period 7/20/10-7/22/10

	7/20/10	7/21/10	7/22/10	7/23/10
$1.60 per Instance-hour	$8.00	$9.60	$20.80	$19.20
$0.10 per EBS GB-month	$0.19	$0.43	$0.48	$0.48
$0.10 per million I/O requests to/ from EBS	$0.02	$0.04	$0.07	$0.08
Total	**$8.22**	**$10.07**	**$21.35**	**$19.76**

Initial setup costs include the costs shown in the tables above for 7/21 and 7/22 plus other miscellaneous costs for a total of $18.44. The charge of $21.35 associated with the 7/22/2010 bill reflects the WRF run that had performance issues due to the hyper-threading problems mentioned above. The charge of $19.76 on 7/23 is associated with the representative single node run of WRF from Table 1. When miscellaneous costs such as elastic IPs are included, the total day's bill for a single instance 8 core job run was $19.76. The EC2 instance was kept alive for approximately 1 hour after the WRF job completed so if we were to only look at associated costs for the single job run we could subtract 1 hour's instance or $1.60 from the $19.76 for a **total charge of $18.16 for a single instance 8 core WRF run.**

The expenses of 7/22 reflect that of a single WRF run where issues with hyper-threading caused a severe degradation of performance. Once we realized our mistake we were able to correct it and obtain the more representative performance run of 7/24. Configuration errors and failed job runs incur charges just as production runs do, so users need to be mindful to avoid unnecessary expenses due to configuration or job setup errors.

Future Work:

Work to date represents use of only a single node running parallel on all 8 cores. Future work will use multiple multi-core instances of WRF with MPI to also communicate across nodes. This will allow us to benchmark the performance of the EC2 node interconnect network and its effect on MPI application scaling. Since these EC2 HPC instances do not provide a shared file system we need to investigate approaches to establish a shared file system using EBS.

Scaling studies will be performed with these multiple instances and added to the baseline performance chart as seen in Figure 2 above. WRF, as many other parallel codes, does not scale linearly as more cores are added. This reduction of application run-time via increasing the number of participating cores comes at a greater financial cost. A cost-benefit analysis of the extra speedup will need to be performed. We will research other parallel codes in addition to WRF to cover a broader spectrum of cloud applicability to a broader HPC application base.

Summary and Conclusions:

Amazon EC2 provides a service that allows users to obtain root-level access to virtual machines hosted on Amazon's infrastructure. The user is then free to install any software and run any process that can be run on a physical machine. These virtual machine instances are provided within a fee-for-use payment structure. Additional services such as elastic (static) IPs are useful to support the installation of licensed software, but come at an additional cost.

An EC2 instance that has roughly equivalent hardware to a single Cyberstar node costs $1.60 per hour. A single run of the Weather Research and Forecasting (WRF) system's Advanced Research WRF (ARW) model (version 2.2.1) costs $18.16 when not amortizing fixed setup costs. These fixed costs are not included because they will likely decrease as administrators and users become more familiar with best practices for efficiently configuring EC2 instances.

Some concerns with EC2's current compute cluster instance include the inability to disable hyper-threading at the BIOS level, forcing users to rely on manual process mappings as well as the relatively limited amount of memory available on the instance (23GB). More work is also necessary to develop a true cluster of instances with the ability to run MPI jobs on several instances.

References relating to the modeling aspects from the DTRA SBL project, for which these simulations were performed:

Gaudet, B.J., N.L. Seaman, D.R. Stauffer, S. Richardson, L. Mahrt, and J.C. Wyngaard, 2008: Verification of WRF-predicted mesogamma-scale spectra in the SBL using a high-frequency filter decomposition. 9th WRF Users' Workshop, Boulder, 23-27 Jun., 4 pp.

Seaman, N.L., B. Gaudet, A. Deng, S. Richardson, D.R. Stauffer, J.C. Wyngaard, and L. Mahrt, 2008: Evaluation of meander-like wind variance in high-resolution WRF model simulations of the stable nocturnal boundary layer. 10th AMS Conf. on Atmos. Chem., New Orleans, LA, 21-24 Jan., 9 pp.

Skamarock, W. C., J. B. Klemp, J. Dudhia, D. O. Gill, D. M. Barker, W. Wang and J. G. Powers, 2005: A Description of the Advanced Research WRF Version 2. NCAR Technical Note.

Stauffer, D.R., B.J. Gaudet, N.L. Seaman, J.C. Wyngaard, L. Mahrt, and S. Richardson, 2009: Sub-kilometer numerical predictions in the nocturnal stable boundary layer. 23rd Conf. on Weather Analysis and Forecasting / 19th Conf. on Numerical Weather Prediction, Omaha, NE, 1-5 Jun., 8 pp.

Young, G. S., B. J. Gaudet, N. L. Seaman, and D.R. Stauffer, 2009: Interaction of a mountain lee wave with a basin cold pool. 13th AMS Conf. on Meso. Proc., Salt Lake City, UT, 17-20 Aug., 6 pp.

About RCC:

Research Computing and Cyberinfrastructure (RCC), a unit of Information Technology Services (ITS), provides systems and services that are used extensively in research, teaching, and service missions at the Pennsylvania State University.

About the Authors:

Michael Fenn is a systems administrator in the RCC group and is part of the team that administers the group's Linux and Windows HPC clusters. He has extensive experience in leveraging virtualized resources in a grid computing environment.

Jason Holmes is the lead systems administrator in the RCC group. He leads the team that administers the group's HPC resources, including the clusters, storage, networking, and software infrastructure.

Jeffrey Nucciarone is the lead research programmer int he RCC group. He has extensive experience in the development, porting, tuning, running, and characterization of parallel applications.