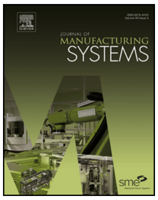
Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys



Democratizing digital design and manufacturing using high performance cloud computing: Performance evaluation and benchmarking

Dazhong Wu[a],*, Xi Liu[b], Steve Hebert[c], Wolfgang Gentzsch[d], Janis Terpenny[a]

[a] *Department of Industrial and Manufacturing Engineering, Pennsylvania State University, University Park, PA 16802, United States*
[b] *The G.W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology Atlanta, GA 30332, United States*
[c] *Nimbix, Dallas, TX 75201, United States*
[d] *The UberCloud, Los Altos, CA 94024, United States*

ARTICLE INFO

Article history:
Received 6 May 2016
Received in revised form
14 September 2016
Accepted 15 September 2016
Available online 26 October 2016

Keywords:
Digital design and manufacturing
Cloud computing
High performance computing
Performance evaluation
Benchmark
Finite element analysis

ABSTRACT

Cloud computing is an innovative computing paradigm that can bridge the gap between increasing computing demands in computationally intensive tasks for digital design and manufacturing applications and limited resources, scalability, flexibility, and agility in traditional computing paradigms. In light of the benefits of cloud computing, cloud-based high performance computing (HPC) has the potential to enable users to not only accelerate computationally expensive tasks, but also to reduce costs by utilizing on-demand, ubiquitous, seamless, and user-friendly access to remote engineering application packages as well as remote HPC resources. However, due to uncertainty about computing performance on the cloud, many manufacturers find it challenging to justify and adopt Cloud-Based Design and Manufacturing (CBDM). Therefore, the objective of this research is to evaluate the performance of solving a large-scale engineering problem using finite element analysis on several public HPC clouds as well as introduce a new workflow for CBDM. A set of experiments is conducted to compare the performance of the public HPC clouds with that of a standard workstation and a dedicated in-house supercomputer. The performance metrics include elapsed time, speedup, scalability, and stability. Experimental results have shown that the Azure Cloud with 32 cores and the Nimbix Cloud with 16 nodes speed up the finite element analysis over a workstation with 8 cores by more than seven-fold and eight-fold. A dedicated in-house supercomputer speeds up the finite element analysis over cloud computing by approximately two-fold because of better I/O performance and larger memory. In addition, considerable variations of elapsed time for solving the finite element model with multiple nodes in the cloud were observed due to resource sharing in cloud computing.

1. Introduction

High performance computing (HPC) refers to the use of supercomputers and parallel processing techniques for solving large-scale and complex engineering and scientific research problems. Dedicated in-house supercomputers have been playing an important role in a wide range of computational and data intensive fields such as biosciences, molecular-scale physics, weather forecasting, media and entertainment, financial modeling, oil and gas exploration, medical research, and product design. According to the 44th TOP500 list of supercomputers, Tianhe-2, a supercomputer developed by China's National University of Defense Technology, is ranked as No. 1 system with a performance of 33.86 petaflops/s on the LINPACK benchmark [1]. However, very few individuals and organizations have access to dedicated in-house supercomputers due to extremely high initial and maintenance costs. For example, the initial cost of Titan built by Cray at Oak Ridge National Laboratory was $60 million. In addition, it costs $6 million per year to operate and maintain Titan.

With rapid changes in technology, computing that utilizes HPC clouds with virtualization technology can deliver high performance computing power that is similar to bare-metal servers while enabling flexible pricing models such as pay-per-use and subscription [2–6]. According to NIST [7], cloud computing is "a model for

* Corresponding author.
E-mail addresses: dwx279@psu.edu (D. Wu), xi.liu@gatech.edu (X. Liu), steve.hebert@nimbix.net (S. Hebert), wolfgang.gentzsch@theubercloud.com (W. Gentzsch), jpt5311@psu.edu (J. Terpenny).

http://dx.doi.org/10.1016/j.jmsy.2016.09.005
0278-6125/© 2016 The Society of Manufacturing Engineers. Published by Elsevier Ltd. All rights reserved.

enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction." Cloud computing is an innovative computing paradigm that can potentially bridge the gap between increasing computing demands and limited resources, scalability, flexibility, and agility in-house computing. Some of the key characteristics of cloud computing, including virtualization, multi-tenancy, scalability or elasticity, and pay-per-use pricing model, have transformed the traditional information and communication technology (ICT) business model from capital expenditure (CapEx) to operating expenditure (OpEx) [8]. For example, the economics of cloud computing and cloud-based services against conventional in-house application deployment have been investigated using benchmarks [9,10]. More importantly, HPC clouds allow users to have on-demand, ubiquitous, and instant access to big data, computing resources, and application software with no upfront cost. Therefore, HPC clouds has the potential to democratize computer-aided engineering by allowing infrastructures, platforms, hardware, and software to be more accessible to more users.

While cloud computing is already widely accepted in the ICT industry, the computer-aided design (CAD), engineering analysis (CAE) and manufacturing (CAM) communities have recently drawn attention to cloud computing. While research pertaining to cloud-based CAD/CAE/CAM is in its infancy, several well-known independent software vendors (ISVs) in the field of CAE have started offering cloud-based services and more flexible licensing models over the past few years [11,12]. Given this trend, modern product design will increasingly rely on large-scale simulations and engineering analysis such as structural and thermal analysis. ANSYS itself [13], and in collaboration with cloud computing companies such as Nimbix, UberCloud, and Gompute, has created cloud-based HPC environments for running ANSYS simulations and post-processing. For example, Nimbix [14] enables users to perform 3D visualizations with ANSYS in the cloud, and has demonstrated that cloud-based rendering allows for compelling, near-photorealistic visualizations without running into memory limits on the graphics server.

More specifically, the benefits of applying cloud-based HPC in CAD/CAE/CAM applications are as follows:

- *Anytime, anywhere access*. Cloud-based HPC enables users to acquire access to any CAD/CAE/CAM (e.g., ANSYS, Abaqus, and AutoCAD) software packages and state-of-the-art HPC computing hardware (e.g., CPU cores, GPUs, memory, and high-speed interconnects) via a web portal and/or application program interfaces (APIs) anytime, anywhere.
- *Cost efficiency*. Cloud-based HPC allows users to solve complex science and engineering problems using cloud-based CAD/CAE/CAM simulations that typically require high bandwidth, low latency networking, numerous CPU cores, and large memory size. In particular, cloud-based HPC enables users to not only improve computing performance as dedicated on premise HPC clusters, but also to reduce costs by utilizing on-demand computing resources and the pay-per-use pricing model without large capital investments.
- *High flexibility*. Cloud-based HPC can scale up and down on demand and transform dedicated HPC clusters into flexible HPC clouds that can be shared and adapted for rapidly changing customer requirements through private, hybrid, and public clouds.
- *High throughput*. Cloud-based HPC can significantly increase the utilization of computing resources as opposed to dedicated on premise HPC by allowing globally dispersed engineering teams to perform complex engineering analysis and simulations concurrently and collaboratively.

Although both academia and industry have shown increasing interest in exploring HPC cloud in CAD/CAE/CAM applications, little work has been reported on evaluating the performance of running CAD/CAE/CAM applications in public HPC clouds against that of traditional dedicated supercomputers and standard workstations. In particular, an important research question to answer is as follows: *How does the performance of public HPC clouds compare with that of a conventional workstation and an in-house supercomputer for large-scale finite element analysis?* As such, the objective of this research is to report on the experiments that were performed via a quantitative and comparative case study for running a large-scale finite element analysis (FEA) simulation model on public HPC clouds. The significance of the research is found in experimental results that provide researchers and practitioners a deeper understanding of the implementation of cloud-based digital design and manufacturing as well as public HPC cloud benchmarks. Specifically, this investigation should be useful to those interested in adopting and implementing cloud-based design, engineering analysis, and manufacturing in public HPC clouds. Primary aspects of this set of experiments that were utilized in answering the aforementioned question include:

- Evaluation of the performance of several public HPC clouds;
- Comparison of the performance of the public HPC clouds with that of a standard workstation;
- Comparison of the performance of the public HPC clouds with that of a dedicated in-house supercomputer.

The remainder of the paper is organized as follows: Section 2 presents a brief overview of cloud-based HPC and CAE on the cloud. Section 3 presents the experimental setup, including the hardware and software specifications of several cloud-based HPC clusters. Section 4 presents the background of a large-scale FEA application example. In the case study, the thermo-mechanical behavior of a 3D stacked die microelectronic package integrated with through silicon vias is analyzed. Section 5 presents the performance evaluation of cloud-based HPC for the FEA application. Section 6 provides conclusions that include a discussion of research contribution and future work.

2. Related work

2.1. CAE applications on the cloud

Providing needed background and context of high performance cloud computing for CAE, this section provides a brief overview of CAE applications on the cloud. CBDM refers to a cloud-based service-oriented product development model in which service consumers are able to accelerate digital design, engineering analysis, and manufacturing simulation in the high performance computing (HPC) cloud as well as reduce upfront costs of building data centers, purchasing software licenses, and acquiring access to massive amount of design- and manufacturing-related data [15–21]. In the Infrastructure-as-a-Service (IaaS) model, cloud service providers offer on-demand access to computing resources such as virtual or bare-metal machines and cloud storage. Examples of IaaS providers include Rackspace, Amazon, Google, and Dropbox. In the Platform-as-a-Service (PaaS) model, cloud service providers deliver computing platforms such as social collaboration platforms, programming and execution environments for cloud computing. Examples of PaaS providers include Google, Microsoft, Amazon, Salesforce, and Nimbix. In the Hardware-as-a-Service (HaaS) model, cloud service providers and consumers

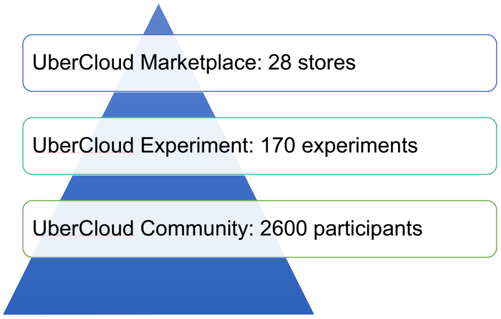

UberCloud Marketplace: 28 stores
UberCloud Experiment: 170 experiments
UberCloud Community: 2600 participants

Fig. 1. UberCloud experiments.

are allowed to rent and lease manufacturing equipment such as milling machines and 3D printers without permanently purchasing and owning them. Examples of HaaS providers include Shapeways, Cubify, Quickparts, MakeXYZ, 3D Hubs, and MFG.com. In the Software-as-a-Service (SaaS) model, cloud service consumers are enabled to run computationally intensive application software such as AutoCAD remotely without installing and running the software on their local computers. Examples of SaaS providers include ANSYS, Autodesk, Dassault Systemes, Sabalcore, Nimbix, and UberCloud.

To date, FEA and computational fluid dynamics (CFD) use cases have been developed or are being developed in HPC cloud environments. As shown in Fig. 1, UberCloud has conducted over one hundred FEA and CFD experiments in various HPC cloud environments over the past two years [22]. However, to the best of our knowledge, no study so far has been conducted to systematically benchmark and evaluate the performance of public HPC clouds using medium- and large-scale use cases for FEA. Without the fundamental understanding of how public HPC clouds can be applied to CAE and how the performance of HPC clouds can be optimized, it will be very challenging for end users, software providers, and cloud service providers to adopt and implement CAE in HPC cloud environments.

2.2. High performance cloud computing

Since the 1990s, the Department of Defense (DoD) High Performance Computing Modernization Program (HPCMP) [23] and the Department of Energy (DoE) Innovative and Novel Computational Impact on Theory and Experiment (INCITE) program [24] have been driving the use of advanced computational environments to solve the most demanding problems in science and engineering. However, in light of the benefits of cloud computing, cloud computing services have been increasingly utilized for HPC applications without building a local (in-house) supercomputer and running on legacy UNIX systems. Although cloud-based HPC enables users to solve complex problems with improved scalability at more affordable prices through various pricing models (e.g., pay-per-use and subscription), HPC workloads are often not suitable for standard clouds, especially those that use virtualization technology. The reason is that HPC applications often require high bandwidth and low latency networking [25]. In this section, the related works on HPC in the cloud are presented.

In order to address the performance and management overhead associated with virtual machines, Huang et al. [26] proposed a framework for HPC applications. This framework was demonstrated using a case study in which a prototype of a virtualized Infiniband cluster was developed. Performance evaluation results showed that HPC applications could achieve almost the same performance as those running in a native and non-virtualized environment. Moreover, this work also demonstrated that other costs incurred by virtualization (e.g., memory consumption ad virtual machine management) could also be reduced by optimizing resource allocation.

In order to perform a quantitative and comparative case study for running HPC applications in public clouds, He et al. [27] conducted a few experiments on three public cloud platforms, including Amazon EC2, GoGrid, and IBM public clouds. Results showed that virtualization adds little performance overhead. In addition, results also demonstrated one of the major problems pertaining to current public clouds for running HPC applications is caused by poor network capabilities. For example, the IBM cloud platform could be equipped with a better network that has higher bandwidth.

Expóito et al. [28] analyzed the performance bottlenecks in HPC applications on the Amazon EC2 Cluster Compute platform using selected micro-benchmarks and representative kernels. Specifically, the communication performance was evaluated on shared memory and a virtualized 10 Gigabit Ethernet network and assessed the scalability of representative HPC codes using parallel CPU cores. Results showed that the scalability of HPC applications in the cloud relies on the network fabric and efficient input/output virtualization support.

Jackson et al. [29] also evaluated the performance of HPC applications in a cloud environment to understand the tradeoffs in migrating into the cloud by comparing HPC clusters to the Amazon EC2 platform. Based on these results, the Amazon EC2 was six times slower than a typical mid-range Linux cluster, twenty times slower than a modern HPC cluster. In addition, the results showed that there was a strong negative correlation between the percentage of time an application spends on communications and its overall performance.

Iosup et al. [30] investigated the performance of cloud computing services for multi-task scientific computing on four commercial cloud platforms, including Amazon EC2, GoGrid, ElasticHosts, and Mosso. The selected performance metrics included wait time, response time, and bounded slowdown (i.e., the ratio between the job response time in the real and exclusively-used environment). Results showed that the performances of the four tested cloud platforms were worse than those of selected scientific computing infrastructures.

Because virtualization may induce significant performance penalties, Ostermann et al. [31] evaluated the usefulness of the current cloud computing services for scientific computing on the Amazon EC2 platform using micro-benchmarks and kernels. Results indicated that the performance and reliability of Amazon EC2 was insufficient for some large scientific computing applications. The reason is that Amazon's EC2 instances run on 10 Gigabit Ethernet networks which result in relatively low-speed interconnections. Most recently, Amazon has launched C3 and C4 instances which can provide enhanced HPC performance such as higher packet per second and lower network latency.

Hazelhurst [32] evaluated the viability of Amazon EC2 for highly scalable HPC applications. A large-scale scientific application example on bioinformatics was performed to explore the computational performance of Amazon EC2 clusters. The impact of shared memory, virtual cores, and network speed on the performance of the Amazon EC2 clusters has been investigated. Results indicated that the relative performance of the Amazon EC2 platform depends on the nature of an application. In addition, it was concluded that cloud computing was a feasible and cost-effective alternative but perhaps will not replace dedicated clusters and supercomputers.

Previous work has focused on the HPC benchmarking and performance evaluation for Amazon EC2 clusters using one of the standard benchmarking tools LINPACK. The LINPACK benchmarks are a measure of a system's floating point computing power (i.e., millions of floating point operations per second (MFLOP/s)) [33].

The LINPACK benchmarks are used to benchmark and rank supercomputers for the 500 most powerful computer systems in the world (i.e., TOP 500 list). The limitations of the previous work are as follows:

1. Although the LINPACK benchmarks measure how fast a computer solves a system of linear equations, the benchmarks can only approximate how fast a computer will perform when solving real engineering and scientific problems. In addition, the LINPACK benchmarks only test the resolution of dense linear systems, which are not representative of all the operations performed in engineering and scientific computing [34]. Therefore, performance evaluation and benchmarks for real-world engineering applications are needed;
2. From the virtualization perspective, Amazon EC2 utilizes Xen hypervisors to allow multiple computer operating systems to execute on the same computer hardware concurrently [3]. However, as an alternative to hypervisor-based virtualization, container-based virtualization has the potential to provide a near-native system performance by virtualizing the user space only instead of both kernel and user spaces of an operating system [35]. In addition, from the I/O perspective, Amazon EC2 instances with 10 GB network connectivity only support up to 4000 Mbps of throughput as opposed to 56 Gbps supported by FDR Infiniband interconnects. Therefore, performance evaluation and benchmarks for a single machine with multi-core or many-core and shared memory parallel as well as multiple machines with distributed memory parallel on more hypervisor-based and container-based public HPC clouds are needed.

3. Experiment

This section presents the details of the experiments conducted. The benchmarks are based on the ANSYS Mechanical which is a comprehensive FEA tool that can be applied for linear and nonlinear structure and thermal analysis. The experiments were conducted on three types of computing systems, including a standard workstation, three public HPC clouds (i.e., Nimbix Multi-node Cloud, NephoScale Single-node Cloud, and Microsoft Single-node Azure Cloud), and a dedicated in-house supercomputer (i.e., CyEnce at Iowa State University). The focus of the experiments was to evaluate the performance of the public HPC clouds as well as to compare the performance of the public clouds with that of the workstation and the supercomputer. Such a comparison can provide benchmarks and help early adopters of HPC cloud understand the benefits of cloud-based HPC.

3.1. Experimental setup

Table 1 lists the hardware and software specifications of the workstation, public HPC clouds, and the dedicated in-house supercomputer, respectively.

Nimbix is a container-based HPC cloud service provider delivering cloud infrastructure and platform as a pay-per-use service [14,36]. Since cloud computing has come into existence in the late 2000s, there is an increasing need to develop a platform that enables developers to quickly build and integrate an application software and its dependencies in a distributed and virtualized cloud environment so that the application software can run on any Linux and Windows servers. Currently, cloud computing makes extensive use of hypervisor-based virtual machines (VMs) that enable the software implementation of a physical computer that executes programs like a physical machine. However, the hypervisor-based virtualization technology virtualizes not only an application and the necessary binaries and libraries but also an entire guest operating system, as illustrated in Fig. 2.

Table 1
Hardware and software specifications.

Workstation	
Processor model	Intel(R) Xeon(R) CPU E5530@ 2.40 GHz
Number of physical cores	8
Memory	24 GB
Parallel processing method	Shared memory parallel (SMP)
Single machine on the Microsoft Azure Cloud (G5 instance)	
Processor model	Intel (R) Xeon (R) CPU E5-2698B v3@ 2.00 GHz
Number of virtual cores	32
Memory	448 GB
Parallel processing method	Shared memory parallel (SMP)
Single machine on the Nephoscale Cloud	
Processor model	Intel(R) Xeon(R) CPU E5-2690 v2 @ 3.00 GHz
Number of physical cores	20
Memory	256 GB
Parallel processing method	Shared memory parallel (SMP)
Multiple machines on the Nimbix Cloud	
Processor model	Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60 GHz
Number of virtual cores per node	16
Memory per node	32 GB
Interconnect	56 Gbps FDR infiniband
File system	Network file system (NFS)
Parallel processing method	Distributed memory parallel (DMP)
Supercomputer at Iowa State University	
Processor model	Intel(R) Xeon(R) CPU E5-2650 @ 2.00 GHz
Number of physical cores per node	16
Memory per node	128 GB
Interconnect	40 Gbps QDR infiniband
File system	Lustre parallel file system
Parallel processing method	Distributed memory parallel (DMP)

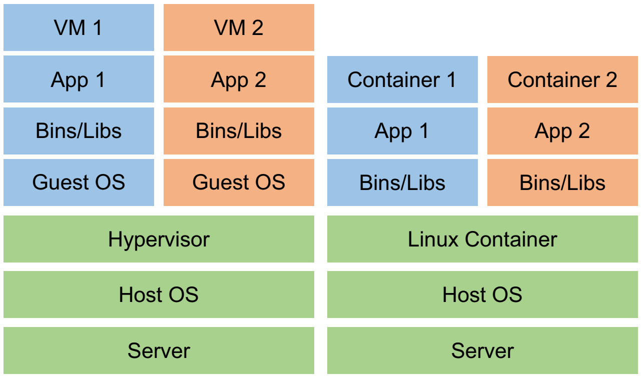


Fig. 2. Hypervisor-based versus container-based virtualization [37].

The advantage of hypervisor-based virtualization is that it allows one to run an application on the entire guest operating system. However, the disadvantage is that system performance may degrade due to additional storage, memory, and I/O overhead incurred by virtualizing the entire operating system. Container-based virtualization, or operating system virtualization as it is sometimes known, is an approach to virtualization in which the virtualization layer of cloud computing systems runs as an application within the operating system. In container-based virtualization, the kernel of the operating system runs on the hardware node with several isolated guest virtual machines installed atop. The isolated guests are called containers as shown in Fig. 2. Container-based virtualization has the potential to provide a lightweight virtualization layer, which promises a near-native system performance. Therefore, container-based virtualization cannot only simplify the access and deployment of application software but also reduce overhead as well as provide better performance. The potential benefits of container-based cloud computing environments include better software portability, better workload visibility, lower overhead, and faster provisioning.

Nimbix implements a cloud software stack based on containerization technology rather than relying on hypervisor-based virtualization technology. Therefore, the container-based virtualization technology in the Nimbix Cloud platform enables end users to consume HPC applications such as ANSYS via a web portal and/or APIs. Because the Nimbix Cloud platform does not need virtualization hypervisors to manage virtual machines, the Nimbix Cloud has the potential to deliver near-native performance with significantly less overhead than hypervisor-based virtualization.

Azure is Microsoft's cloud computing platform providing a collection of integrated services such as data analytics, computing, database, and storage. Azure virtual machines enable users to deploy a Windows Server or Linux image on the cloud. As opposed to container-based virtualization, Azure utilizes Windows Server 2008 and Microsoft Azure Hypervisor to provide virtualization of services. A virtual machine on the Azure Cloud can be scaled up from 1 core with 0.75 GB of RAM and 20 GB of disk size up to 32 cores with 448 GB of RAM and 6144 GB of disk size [38].

Nephoscale is a HPC cloud platform providing both public and private cloud computing services. Virtual servers on the Nephoscale Cloud platform utilize the kernel-based virtual machine hypervisor, Intel E5 Processors, DDR3 RAM, Solid State Drive (SSD) storage, and 10 Gbps networking devices. In addition to virtual servers, Nephoscale also enables users to solve large-scale engineering and scientific problems on bare-metal dedicated servers which generally outperform virtual servers with the same RAM and CPU cores by 25% or more depending on applications, workload, and server parameters [39].

3.2. *Workflow*

This section presents the workflows for running CAE applications such as finite element simulations on a dedicated in-house supercomputer and a public HPC cloud.

As illustrated in Fig. 3, a typical workflow for running finite element simulations on dedicated in-house clusters consists of the following steps:

(1) Stage an input file from the end user onto the cluster's parallel file system called scratch space;
(2) Create a portable batch system (PBS) script for running a job;
(3) Submit a job using the PBS script and the qsub command to the portable batch system that allocates computational tasks (e.g., the number of nodes and processors);
(4) The job will either start running immediately on the cluster or wait in the queue for its turn, while input data waits on the scratch space. Because HPC clusters are generally fully utilized, it is very common for a job to spend hours or even days in the queue;
(5) Once the job is finished, transfer the result file to a local machine;
(6) View the FEA result using a FEA viewer from the local machine.

In comparison to the aforementioned workflow, a typical workflow for running finite element simulations on public HPC clouds consists of the following steps:

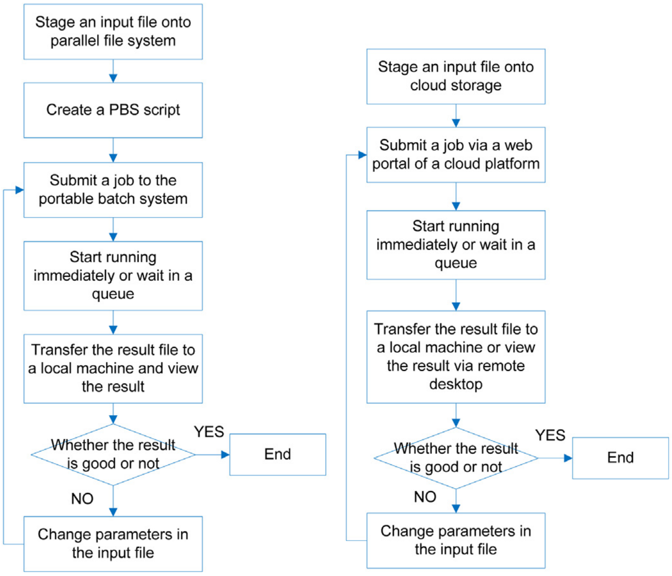

Stage an input file onto parallel file system
Create a PBS script
Submit a job to the portable batch system
Start running immediately or wait in a queue
Transfer the result file to a local machine and view the result
Whether the result is good or not
YES
End
NO
Change parameters in the input file
Stage an input file onto cloud storage
Submit a job via a web portal of a cloud platform
Start running immediately or wait in a queue
Transfer the result file to a local machine or view the result via remote desktop
Whether the result is good or not
YES
End
NO
Change parameters in the input file

Fig. 3. Workflow on the supercomputer and HPC clouds.

(1) Stage the input data from the end user onto cloud storage such as Amazon Cloud Storage and Dropbox. The input data can be transferred to and from the cloud storage via secure file transfer protocol (SFTP) or a folder on a desktop using web distributed authoring and versioning (WebDAV) over HTTPS;
(2) Submit a job to a web portal of a HPC cloud by configuring the computing parameters such as the number of nodes or submit a job using the interactive remote desktop mode;
(3) Similar to an in-house supercomputer, the job will either start running immediately on the cloud or wait in the queue for its turn. However, because of the virtualization technology, HPC clouds generally have higher service availability than in-house dedicated supercomputers;
(4) If the job is submitted via a web portal, transfer the result file to a local machine and view the result from the local machine. If the job is submitted via the remote desktop mode, view the result via remote desktop without transferring the result file to a local machine.

The aforementioned workflows will be implemented for solving a large-scale FEA problem.

4. Example application

The example application is the thermo-mechanical warpage analysis of a 3D stacked die microelectronic package integrated with through silicon vias (TSVs). Over the last decade, digital information processing devices for HPC systems require an increasing level of computing power while using less power and space. 3D integrated logic devices with stacked memory using through-silicon vias have the potential to meet this demand because the shorter and highly parallel connection between logic and high-capacity memory can avoid the von Neumann bottleneck, reduce power consumption, and realize the highest device density. However, the challenges pertaining to 3D packaging with TSVs lie in yield, assembly, test, and reliability issues [40–42]. In particular, the 3D stacked die package warpage problem is one of the key challenges for 3D package assembly. Understanding the package warpage behavior is crucial to achieve high package stack yield because the different warpage directions of top and bottom package will impact the yield of package stacking [43–45].

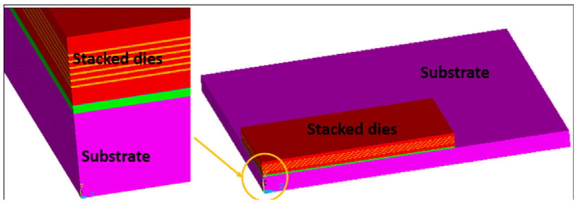


Fig. 4. Schematic view of the 3D microelectronic package.

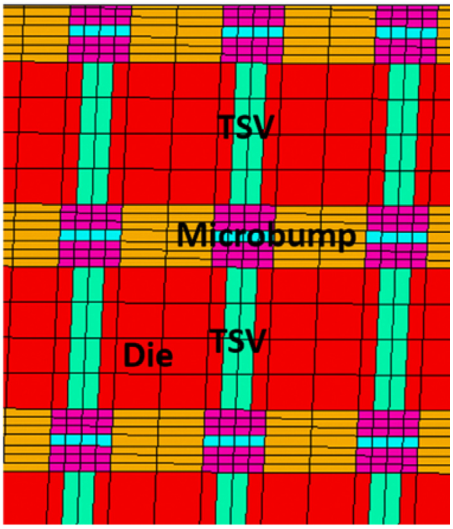


Fig. 5. Cross-section view of the mesh near the stacked-die region.

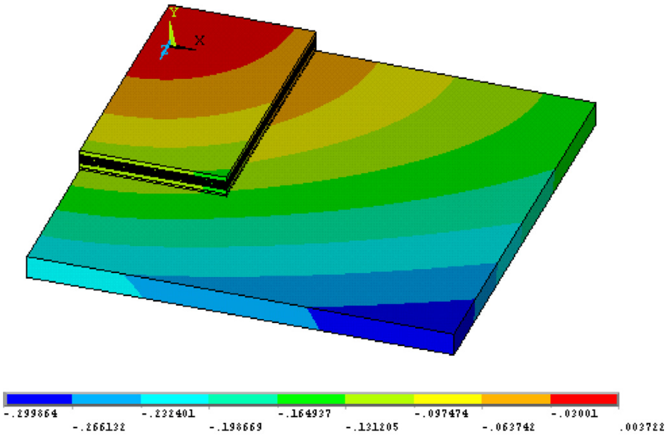


Fig. 6. Warpage of the 3D microelectronic package at room temperature.

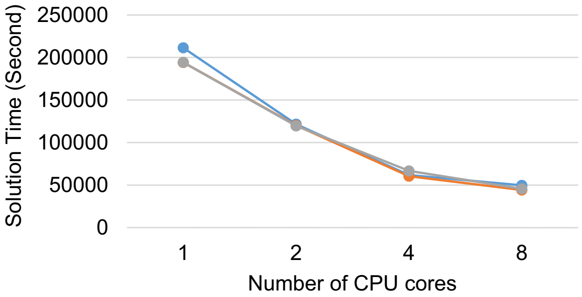


Fig. 7. Solution scalability and performance stability for the workstation.

In the example application, to address the aforementioned issue, a finite element model is created with detailed package features, as shown in Fig. 4, to investigate the warpage behaviors of stacked die 3D packages, which is not well understood due to various reasons. One of the reasons is that 3D stacked dies interconnected with TSVs are still in the development stage. Thus, a very limited amount of prototype samples are available for investigating the warpage problem. The other reason is that the numerical simulation of 3D packages is extremely computationally expensive. For example, in the 3D packages, the in-plane dimensions are at the millimeter scale. However, the out-of-plane dimensions, TSVs, and micro-bumps are at the micrometer scale, which results in a significantly increased finite-element mesh density to meet the element aspect ratio requirements. In addition, there are generally hundreds or thousands of TSVs/micro-bumps between each stacked die.

As shown in Fig. 5, when a coarse mesh has been used to model the TSVs/micro-bumps region, there are still 8,075,538 degrees of freedom (DOF) in the finite element model. Consequently, it takes a standard workstation 15–60 h to solve only one step, cooling down the entire package from the underfilling temperature (165 °C) to the room temperature (26 °C). To further simplify the finite element model, static structural analysis with steady state thermal loading was applied to simulate the cooling process. Fig. 6 presents the dome warpage shape after cooling down, which is due to the larger coefficient of thermal expansion of organic substrate than that of silicon dies. Large package warpage may pose serious package interconnect (TSVs, micro-bumps, and solder bumps) reliability challenges in addition to package assembly yield issues.

Some of the detailed specifications about the finite element model are summarized as follows:

- 8,075,538 degrees of freedom.
- SOLID185 elements.
- Linear elastic analysis.
- Distributed sparse solver.

5. Experimental results

5.1. Workstation performance

In the first experiment, the finite element model was executed on the 8-core high performance workstation using 1, 2, 4, 6, and 8 cores for three times. Table 2 lists the results on the core count, memory usage, solver computational rate, I/O rate, and elapsed time. The average elapsed time using 8 CPU cores has achieved 4.3× speed-up than that of the single CPU core.

Optimally, the speed-up from parallelization would be linear. However, according to the Amdahl's law [46,47], the speed-up of a program from parallelization is limited by how much of the program can be parallelized. In practice, most of parallel computing results have a near-linear speed-up for small number of processors, which flattens out into a constant value for large numbers of processors rather than achieve optimal speed-ups, as shown in Fig. 7. For example, in the experiment, relatively good solution scalability can be achieved using less than 4 CPU cores. However, the finite element simulation does not scale using 6 and 8 CPU cores. Adding more CPU cores will not significantly reduce elapsed time, and in some cases, result in increased elapsed time, as shown in Fig. 7. In addition, variability for elapsed time for running the finite element model on the workstation is very small. In other words, the experimental results on the workstation are repeatable.

5.2. HPC cloud performance

In the following experiments, the finite element model was executed on three public HPC clouds, including two clouds with a single

Table 2
Elapsed time for the workstation.

CPU core #	Memory (MB)	Solver computational rate (Mflops)	I/O rate (MB/s)	Elapsed time (s)
1	30,167	8231.5	131.2	211,319
1	30,167	9438.9	134.9	194,132
1	30,167	9438.9	134.9	194,132
2	30,068	16,550.5	143.6	121,553
2	30,068	16,649.6	142.7	120,147
2	30,068	16,452.9	145.1	119,586
4	30,069	36,507.4	127.6	61,752
4	30,069	37,572.3	142.5	60,391
4	30,069	25,045.9	142.6	66,671
8	30,069	53,704.6	139.7	49,625
8	30,069	57,483.3	143.9	44,156
8	30,069	55,137.2	138.9	45,572

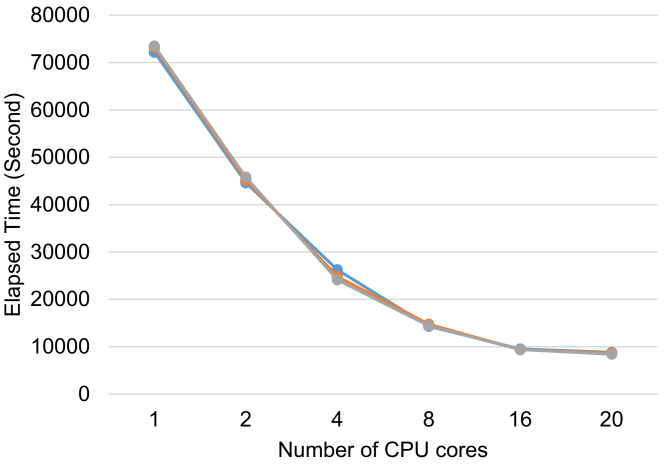


Fig. 8. Solution scalability and performance stability for the single machine on the Nephoscale Cloud.

machine and one cloud with multiple machines. In order to benchmark these HPC clouds and evaluate performance stability, three replicates are generated for each experiment.

5.2.1. *Single machine instance performance*

The objective of the following two experiments is to evaluate the performance of single cloud server instances without interprocess communication across multiple nodes. The finite element model was executed using 4, 8, 16, and 20 CPU cores on a single machine on the Nephoscale Cloud. Fig. 8 shows the solution scalability and performance stability. Relatively good solution scalability is achieved using less than 16 CPU cores. Similar to the workstation, variability for elapsed time for running the finite element model on the single machine of the Nephoscale Cloud is very small. Similar to the workstation, the experimental results on the single machine on the Nephoscale Cloud are repeatable. Fig. 9 shows the relative speedup between the single machine on the Nephoscale Cloud and the workstation using 8 CPU cores. The largest speedup is 5.39× using 20 CPU cores.

We also execute the finite element model using 4, 8, 16, and 32 CPU cores on the other single machine on the Microsoft Azure Cloud. Fig. 10 shows the solution scalability and performance stability. Similar to the single machine instance on the Nephoscale Cloud, relatively good solution scalability is achieved using less than 16 CPU cores. Variability for elapsed time for running the finite element model on the single machine on the Microsoft Azure Cloud is also very small. Therefore, the experimental results on the single machine on the Microsoft Azure Cloud are also repeatable. Fig. 11 shows the relative speedup between the single machine on the

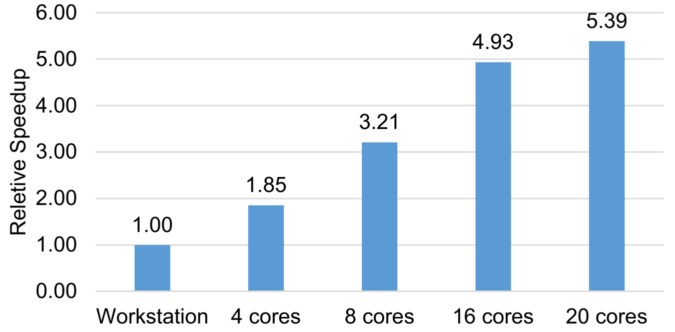


Fig. 9. Relative speedup between the single machine on the Nephoscale Cloud and the workstation.

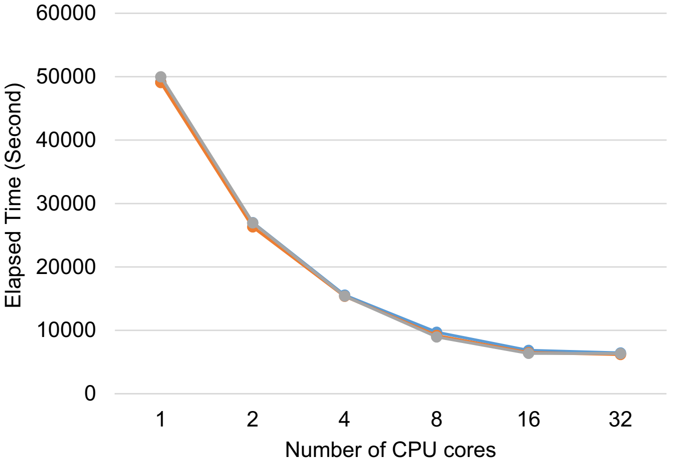


Fig. 10. Solution scalability and performance stability for the single machine on the Azure Cloud.

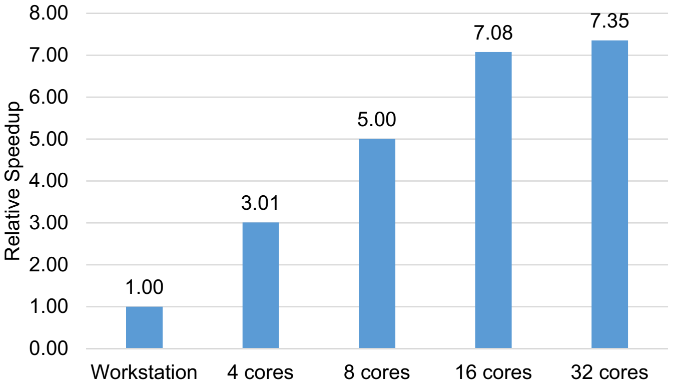
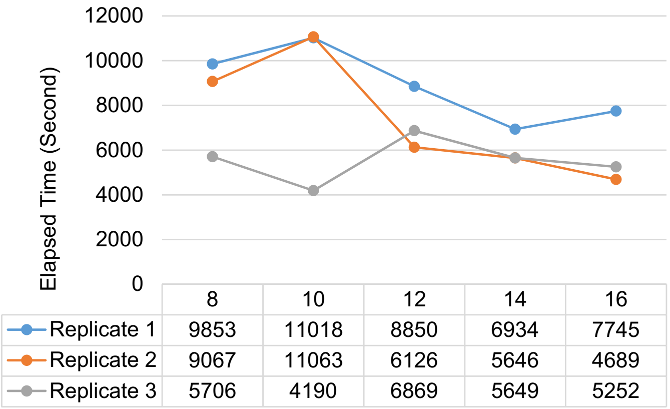


Fig. 11. Relative speedup between the single machine on the Azure Cloud and the workstation.

Table 3
Elapsed time for the Nimbix Cloud.

Node #	CPU core #	Memory (MB)	Solver computational rate (Mflops)	I/O rate (MB/s)	Elapsed time (s)
8	128	270,847	439,807.6	551.2	9853
8	128	270,847	567,303.3	603.4	9067
8	128	274,096	683,412.6	636.9	5706
10	160	329,111	368,512.3	541.7	11,018
10	160	327,716	329,803.6	475.1	11,063
10	160	329,111	916,573.4	1102.4	4190
12	192	381,439	555,754.6	840.9	8850
12	192	381,439	941,153.8	942.5	6126
12	192	381,439	739,670.4	1222.4	6869
14	224	430,585	735,222.3	840.9	6934
14	224	436,738	768,837.5	948.4	5646
14	224	429,825	988,689.6	1456.4	5649
16	256	485,795	592,235.4	873.9	7745
16	256	504,824	717,520.8	1843.0	4689
16	256	479,581	700,708.9	4863.0	5252



	8	10	12	14	16
Replicate 1	9853	11018	8850	6934	7745
Replicate 2	9067	11063	6126	5646	4689
Replicate 3	5706	4190	6869	5649	5252

Fig. 12. Solution scalability and performance stability for the multiple machines on the Nimbix Cloud.

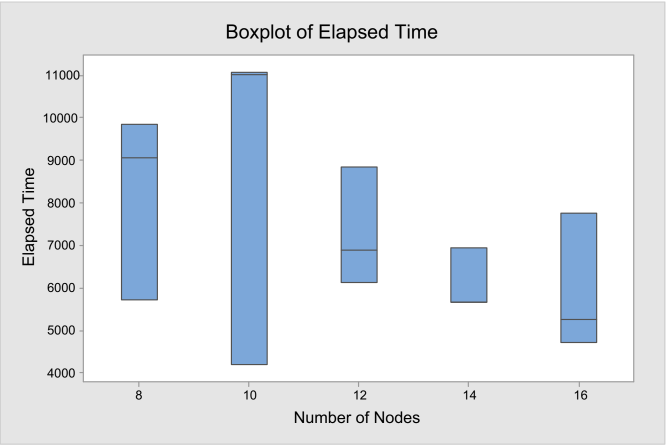


Fig. 13. Boxplot for elapsed time.

Microsoft Azure Cloud and the workstation using 8 CPU cores. The largest speedup is 7.35× using 32 CPU cores.

5.2.2. *Multiple machine instance performance*

In the fourth experiment, we execute the finite element model on the Nimbix Cloud with multiple nodes. The objective of the experiment is to evaluate the performance of the Nimbix Cloud with inter-process communication across multiple nodes. Table 3 lists the memory usage, solver computational rate, I/O rate, and elapsed time for running the finite element model on the Nimbix Cloud using 8, 10, 12, 14, and 16 nodes. Fig. 12 shows the solution scalability and performance stability. As opposed to the workstation and single machine instances on the Nephoscale and Microsoft Clouds, the finite element simulation almost does not scale using 8, 10, 12, 14, and 16 nodes. More importantly, variability for elapsed time for running the finite element model on the Nimbix Cloud is very large. Fig. 13 shows a boxplot which indicates the mean and degree of dispersion of the elapsed times for the Nimbix Cloud (i.e., the minimum and maximum elapsed time in each replicate). These large variations in elapsed times for running the finite element simulation on the Nimbix Cloud result from the fact that multiple users or instances share the same physical machines or compete for the same computing resources (e.g., memory and I/O bandwidth) simultaneously on the cloud. Fig. 14 shows the relative speedup between the Nimbix Cloud and the workstation using 8 CPU cores. The largest speedup is 8.37× using 16 nodes.

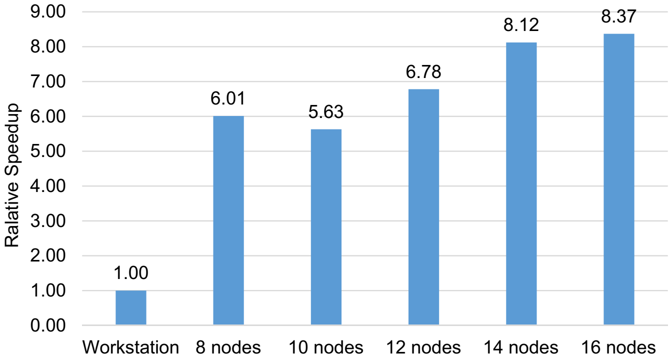


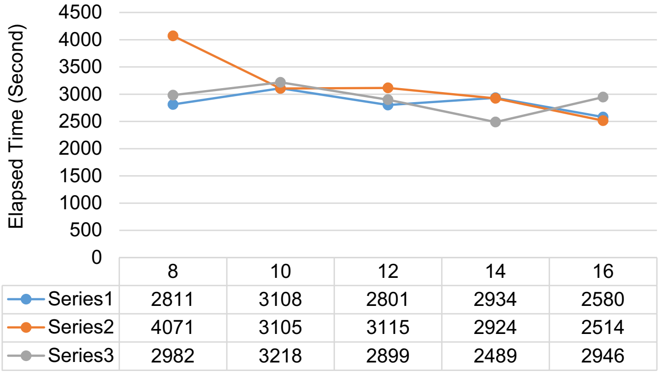
Fig. 14. Relative speedup between the Nimbix Cloud and the workstation.

5.3. *Dedicated in-house supercomputer performance*

In the fifth experiment, we execute the finite element model on a dedicated in-house supercomputer, CyEnce, built by Iowa State University. The objective of the experiment is to evaluate the performance of the dedicated supercomputer against that of the workstation and the HPC clouds. Table 4 lists the memory usage, solver computational rate, I/O rate, and elapsed time for running the finite element model on the CyEnce supercomputer using 8, 10, 12, 14, and 16 nodes. Fig. 15 shows the solution scalability and performance stability. Similar to the multiple machines on the Nimbix Cloud, the finite element simulation does not scale using 8, 10, 12, 14, and 16 nodes. However, as opposed to the Nimbix Cloud, vari-

Table 4
Elapsed time for the CyEnce supercomputer.

Node #	CPU core #	Memory (MB)	Solver computational rate (Mflops)	I/O rate (MB/s)	Elapsed time (s)
8	128	330,675	747,772.8	41,355.6	2811
8	128	306,614	764,756.4	41,479.1	4071
8	128	335,819	846,864.5	41,197.2	2982
10	160	361,189	960,935.3	48,143.2	3108
10	160	405,707	897,665.1	45,224.4	3105
10	160	383,273	972,394	51,346.4	3218
12	192	404,863	1,037,305	49,931.6	2801
12	192	448,757	1,120,895	50,601.7	3115
12	192	434,235	1,075,439	49,388.1	2899
14	224	477,620	1,053,513	46,727	2934
14	224	473,944	1,123,758	53,453.1	2924
14	224	496,306	1,158,918	48,369.4	2489
16	256	561,353	1,153,494	54,077.3	2580
16	256	555,877	1,207,179	51,219.1	2514
16	256	559,037	895,557.2	41,779.1	2946



	8	10	12	14	16
Series1	2811	3108	2801	2934	2580
Series2	4071	3105	3115	2924	2514
Series3	2982	3218	2899	2489	2946

Fig. 15. Solution scalability and performance stability for the supercomputer.

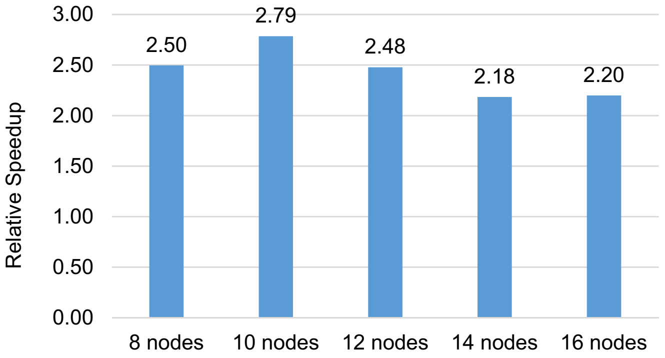


Fig. 17. Relative speedup between the supercomputer and Nimbix Cloud.

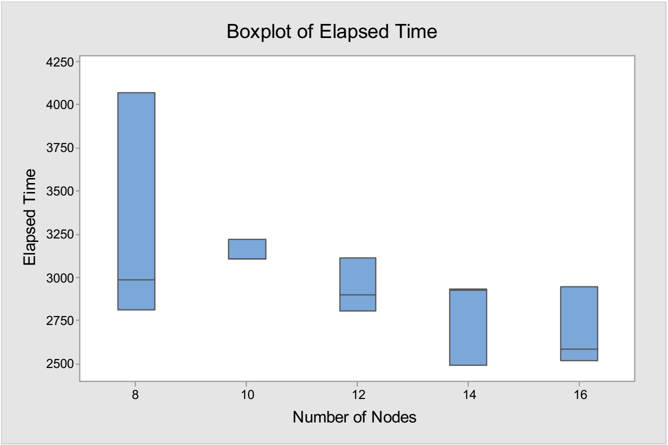


Fig. 16. Boxplot for elapsed time.

ability for elapsed time for running the finite element model on the dedicated supercomputer is very small, as shown in Fig. 16. Fig. 17 shows the relative speedup between the supercomputer and the Nimbix Cloud. The relative speedup is 2–3 times using 8, 10, 12, 14, and 16 nodes. In this particular case, the dedicated supercomputer outperforms the Nimbix Cloud for the following two reasons:

1. Each node on the dedicated in-house supercomputer has 128 GB of memory which is significantly larger than 32 GB of memory on each node of the Nimbix Cloud.
2. The Lustre file system on the dedicated in-house supercomputer is a parallel distributed file system used for large-scale cluster computing. In general, the Lustre file system can achieve better aggregate I/O throughput than that of the NFS.

6. Conclusion and future work

In this paper, a new workflow was introduced to utilize high performance cloud computing resources for accelerating compute-intensive tasks in digital design and manufacturing. A set of experiments was conducted to evaluate the performance of several public HPC clouds using a large-scale finite element model with more than eight million degrees of freedom. The elapsed time, speedup, scalability, and stability were used to measure the performance of the cloud computing services. The experimental results have shown that the Azure Cloud with 32 cores and the Nimbix Cloud with 16 nodes speed up the finite element analysis over a workstation with 8 cores by more than seven-fold and eight-fold. The dedicated in-house supercomputer speeds up the finite element analysis over cloud computing by approximately two-fold because of better I/O performance and larger memory. Moreover, considerable variations of elapsed time for solving the finite element model using multiple nodes in the cloud were observed. These variations in elapsed times resulted from the fact that multiple users or instances shared the same physical machines simultaneously.

We believe that public HPC clouds will not replace on-premise supercomputers in the near future, although public HPC clouds provide compelling performance. This is because in-house supercomputers provide high-bandwidth, low latency interconnects and high-speed distributed file systems that are required by large-scale and latency sensitive applications such as real-time process monitoring, while cloud computing might have limitations in both interconnects and distributed file systems. However, HPC clouds offer unique advantages, including remote access to large volumes of historical and streaming data, scalable computing capacity, and flexible pricing models [9], although performance stability might be one of the major concerns with democratizing digital design and manufacturing using high performance cloud computing. In response to the initial research question, our experimental results have shown that the performance of the HPC clouds are reasonably sufficient for solving the large-scale finite element analysis problem.

In the future, it will be worthwhile to benchmark and evaluate the performance of public, private, and hybrid HPC clouds using more FEA and CFD software packages such as MSC Nastran and ANSYS Fluent packages. In addition, it is also very important to design the experiments using the design of experiments (DOE) technique to identify the relationship between factors affecting the performance of the HPC clouds. A well-designed experiment can help find cause-and-effect relationships as well as provide answers to the following more in-depth research questions: (1) what are the key controllable input factors in conducting FEA and CFD simulations on public HPC clouds? (2) What are the main and interaction effects in conducting these simulations on the public HPC clouds? (3) At what settings would the FEA and CFD simulations deliver optimal performances (i.e., minimum elapsed time and linear scalability)? (4) What hardware and software settings would generate less variation in the simulation output? Moreover, a multi-objective optimization problem needs to be formulated for studying trade-offs in the performance and costs associated with solving engineering and scientific problems on the cloud. From a cybersecurity perspective, as manufacturers are increasingly being targeted by not only hackers and cyber criminals but also competing companies and nations engaged in corporate espionage, much work remains to be conducted to address concerns for data confidentiality, integrity, and availability. Although the aforementioned challenges may exist when implementing HPC clouds, we believe HPC clouds with improved workflow, user experience, and cybersecurity will enable users ranging from individuals, SMEs, and large enterprises to have affordable, secure, on-demand, user-friendly, and instant remote access to infrastructures, platforms, hardware, and software for solve large-scale and complex engineering problems in the field of digital design and manufacturing.

Disclaimer

Certain commercial equipment, instruments, suppliers, and software are identified in this paper to foster understanding. Such identification does not imply recommendation or endorsement by the authors, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

Acknowledgements

The research reported in this paper is partially supported by NSF under grant number IIP-1238335. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. We would also like to thank ANSYS, Inc., Microsoft, and Nephoscale, Inc. for providing the ANSYS HPC Packs license, access to Azure and Nephoscale Cloud resources, respectively.

References

[1] TOP500, http://www.top500.org/lists/2014/11/; 2015.

[2] Vecchiola C, Pandey S, Buyya R. High-performance cloud computing: a view of scientific applications. Proceedings of the 2009 10th International Symposium on Pervasive Systems, Algorithms, and Networks (ISPAN), IEEE 2009:4–16.

[3] Mauch V, Kunze M, Hillenbrand M. High performance cloud computing. Future Gener Comput Syst 2013;29(6):1408–16.

[4] Xu X. From cloud computing to cloud manufacturing. Robot Comput Integr Manuf 2012;28(1):75–86.

[5] Wang L. Machine availability monitoring and machining process planning towards cloud manufacturing. CIRP J Manuf Sci Technol 2013;6(4):263–73.

[6] Chen T. Strengthening the competitiveness and sustainability of a semiconductor manufacturer with cloud manufacturing. Sustainability 2014;6(1):251–66.

[7] Liu F, Tong J, Mao J, Bohn R, Messina J, Badger L, Leaf D. NIST cloud computing reference architecture, vol. 500. NIST Special Publication; 2011. p. 292.

[8] Wang XV, Wang L. WRCloud: a novel WEEE remanufacturing cloud system. Procedia CIRP 2015;29:786–91.

[9] Wu D, Terpenny J, Gentzsch W. Economic benefit analysis of cloud-based design, engineering analysis, and manufacturing. J Manuf Sci Eng 2015;137(4):040903.

[10] Marston S, Li Z, Bandyopadhyay S, Zhang J, Ghalsasi A. Cloud computing—the business perspective. Decis Support Syst 2011;51(1):176–89.

[11] Gentzsch W, Yenier B. The UberCloud HPC experiment: compendium of case studies, https://www.theubercloud.com/ubercloud-compendium-2013/; 2013.

[12] Gentzsch W, Yenier B. The UberCloud HPC experiment: compendium of case studies, https://www.theubercloud.com/ubercloud-compendium-2014/; 2014.

[13] ANSYS. ANSYS enterprise cloud, http://www.ansys.com/Products/Workflow+Technology/Cloud+&+IT+Solutions/ANSYS+Enterprise+Cloud; 2015.

[14] Nimbix, http://www.nimbix.net/jarvice/; 2015.

[15] Wu D, Rosen DW, Schaefer D. Cloud-based design and manufacturing: status and promise, cloud-based design and manufacturing (CBDM). Springer; 2014. p. 1–24.

[16] Wu D, Thames JL, Rosen DW, Schaefer D. Towards a cloud-based design and manufacturing paradigm: looking backward, looking forward. In: ASME 2012 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. 2012. p. 315–28.

[17] Wu DZ, Thames JL, Rosen DW, Schaefer D. Enhancing the product realization process with cloud-Based design and manufacturing systems. J Comput Inf Sci Eng 2013;13(4).

[18] Wu D, Rosen DW, Wang L, Schaefer D. Cloud-based design and manufacturing: a new paradigm in digital manufacturing and design innovation. Comput Aided Des 2015;59:1–14.

[19] Wu D, Greer MJ, Rosen DW, Schaefer D. Cloud manufacturing: strategic vision and state-of-the-art. J Manuf Syst 2013;32(4):564–79.

[20] Wu D, Schaefer D, Rosen DW. Cloud-based design and manufacturing systems: a social network analysis. In: Proceedings of the 19th International Conference on Engineering Design (ICED13). 2013.

[21] Wu D, Rosen DW, Wang L, Schaefer D. Cloud-based manufacturing: old wine in new bottles? Proceedings of the 47th CIRP Conference on Manufacturing Systems. 2014.

[22] Gentzsch W, Yenier B. The UberCloud experiment: tech. comp. in the cloud-2nd compendium of case studies. Tech. Rep., Tabor Communications, Inc.; 2014.

[23] DoD, http://140.32.246.40/; 2015.

[24] DoE, http://www.doeleadershipcomputing.org/incite-program/; 2015.

[25] Hwang K, Bai X, Shi Y, Li M, Chen W, Wu Y. Cloud performance modeling with benchmark evaluation of elastic scaling strategies. IEEE Trans Parallel Distr Syst 2016;27(1):130–43.

[26] Huang W, Liu J, Abali B, Panda DK. A case for high performance computing with virtual machines. Proceedings of the 20th Annual International Conference on Supercomputing, ACM 2006:125–34.

[27] He Q, Zhou S, Kobler B, Duffy D, McGlynn T. Case study for running HPC applications in public clouds. Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing, ACM 2010:395–401.

[28] Expósito RR, Taboada GL, Ramos S, TouriñO J, Doallo R. Performance analysis of HPC applications in the cloud. Future Gener Comput Syst 2013;29(1):218–29.

[29] Jackson KR, Ramakrishnan L, Muriki K, Canon S, Cholia S, Shalf J, Wasserman HJ, Wright NJ. Performance analysis of high performance computing applications on the amazon web services cloud. Proceedings of the 2010 IEEE Second International Conference on Cloud Computing Technology and Science (CloudCom), IEEE 2010:159–68.

[30] Iosup A, Ostermann S, Yigitbasi MN, Prodan R, Fahringer T, Epema DH. Performance analysis of cloud computing services for many-tasks scientific computing. IEEE Trans Parallel Distrib Syst 2011;22(6):931–45.

[31] Ostermann S, Iosup A, Yigitbasi N, Prodan R, Fahringer T, Epema D. A performance analysis of EC2 cloud computing services for scientific computing. Springer: Cloud Computing; 2010. p. 115–31.

[32] Hazelhurst S. Scientific computing using virtual high-performance computing: a case study using the Amazon elastic computing cloud. Proceedings of the 2008 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries: Riding the Wave of Technology, ACM 2008:94–103.

[33] Dongarra JJ, Bunch JR, Moler CB, Stewart GW. LINPACK users' guide. Siam; 1979.

[34] Luszczek P, Dongarra JJ, Koester D, Rabenseifner R, Lucas B, Kepner J, McCalpin J, Bailey D, Takahashi D. Introduction to the HPC challenge benchmark suite. Lawrence Berkeley National Laboratory; 2005.

[35] Xavier MG, Neves MV, Rossi FD, Ferreto TC, Lange T, De Rose CA. Performance evaluation of container-based virtualization for high performance computing

environments. Proceedings of the 2013 21st Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), IEEE 2013:233–40.

[36] Reiter L. Containers in the cloud, http://www.nimbix.net/blog/2014/04/29/containers-in-the-cloud-2/; 2014.

[37] Docker, What is Docker? https://www.docker.com/whatisdocker/.

[38] Azure, Microsoft Azure, http://azure.microsoft.com/en-us/pricing/details/virtual-machines/#Linux;
2015.

[39] Nephoscale, http://nephoscale.com/dedicated-servers/; 2015.

[40] Liu X, Chen Q, Dixit P, Chatterjee R, Tummala RR, Sitaraman SK. Failure mechanisms and optimum design for electroplated copper through-silicon vias (TSV). Proceedings—Electronic Components and Technology Conference, 2009. ECTC 2009. 59th, IEEE 2009:624–9.

[41] Liu X, Chen Q, Sundaram V, Tummala RR, Sitaraman SK. Failure analysis of through-silicon vias in free-standing wafer under thermal-shock test. Microelectron Reliab 2013;53(1):70–8.

[42] Liu X, Simmons-Matthews M, Wachtler KP, Sitaraman SK. Reliable design of TSV in free-standing wafers and 3D integrated packages. In: Proceedings ASME International Mechanical Engineering Congress and Exposition. 2011. p. 903–10.

[43] Liu X, Li M, Mullen D, Cline J, Sitaraman SK. Design and assembly of a double-sided 3D package with a controller and a DRAM stack. Proceedings Electronic Components and Technology Conference (ECTC), 2012 IEEE 62nd, IEEE 2012:1205–12.

[44] Liu X, Li M, Mullen D, Cline J, Sitaraman S. Experimental and simulation study of double-sided flip-chip assembly with a stiffener ring. IEEE Trans Device Mater Reliab 2014;14(1):512–22.

[45] Liu X, Chen Q, Sundaram V, Simmons-Matthews M, Wachtler KP, Tummala RR, et al. Reliability assessment of through-silicon vias in multi-die stack packages. IEEE Trans Device Mater Reliab 2012;12(2):263–71.

[46] Gustafson JL. Reevaluating Amdahl's law. Commun ACM 1988;31(5):532–3.

[47] Hill MD, Marty MR. Amdahl's law in the multicore era. IEEE Comput 2008;41(7):33–8.