

Utilize cloud computing to support dust storm forecasting

Qunying Huang^{a*}, Chaowei Yang^a, Karl Benedict^b, Songqing Chen^c,
Abdelmounaam Rezgui^a and Jibo Xie^d

^aJoint Center for Intelligent Spatial Computing, Geography and Geoinformation Science, George Mason University, Fairfax, VA, USA; ^bEarth Data Analysis Center (EDAC), University of New Mexico, Albuquerque, NM, USA; ^cDepartment of Computer Science, George Mason University, Fairfax, VA, USA; ^dCenter for Earth Observation and Digital Earth, Beijing, China

(Received 24 December 2011; final version received 11 November 2012)

The simulations and potential forecasting of dust storms are of significant interest to public health and environment sciences. Dust storms have interannual variabilities and are typical disruptive events. The computing platform for a dust storm forecasting operational system should support a disruptive fashion by scaling up to enable high-resolution forecasting and massive public access when dust storms come and scaling down when no dust storm events occur to save energy and costs. With the capability of providing a large, elastic, and virtualized pool of computational resources, cloud computing becomes a new and advantageous computing paradigm to resolve scientific problems traditionally requiring a large-scale and high-performance cluster. This paper examines the viability for cloud computing to support dust storm forecasting. Through a holistic study by systematically comparing cloud computing using Amazon EC2 to traditional high performance computing (HPC) cluster, we find that cloud computing is emerging as a credible solution for (1) supporting dust storm forecasting in spinning off a large group of computing resources in a few minutes to satisfy the disruptive computing requirements of dust storm forecasting, (2) performing high-resolution dust storm forecasting when required, (3) supporting concurrent computing requirements, (4) supporting real dust storm event forecasting for a large geographic domain by using recent dust storm event in Phoenix, 05 July 2011 as example, and (5) reducing cost by maintaining low computing support when there is no dust storm events while invoking a large amount of computing resource to perform high-resolution forecasting and responding to large amount of concurrent public accesses.

Keywords: spatial cloud computing; CyberGIS; cloud GIS; loosely coupled nested model; Amazon EC2

1. Introduction

Dust storms are one of the most dangerous natural hazards and rank only after extreme heat/cold and floods for causing the most deaths and injuries (Shoemaker and Davis 2008). An increasing number of dust storm events have caused an increase in the reports of the loss of lives and property damages since 1995 (Figure 1). Deaths and injuries are usually caused by car accidents, because of the reduced visibility during dust storm events. A dust storm forecasting system would be able to alert the

*Corresponding author. Email: qhuang1@gmu.edu

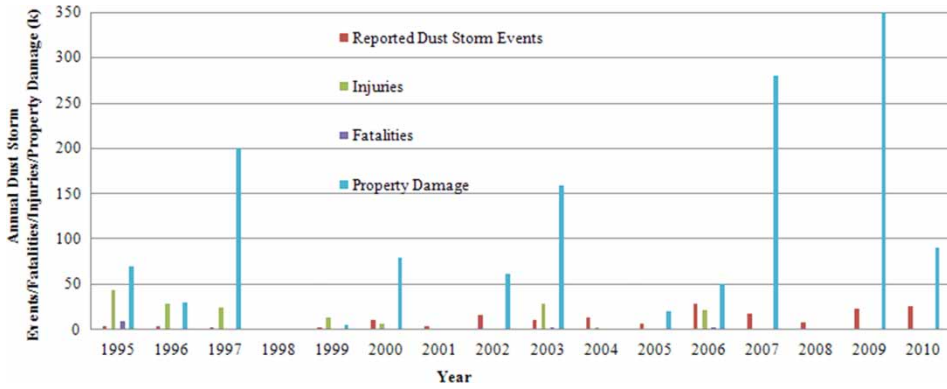


Figure 1. Yearly frequency of reported dust storm events, fatalities, injuries, and property damages in Arizona from 1995 to 2010 (the unit of property damages is k dollar) (Data source: NOAA).

public of an approaching dust storm and can help people make safer plans before the potential event. That being said a dust storm forecasting system would be very helpful to enable the public to better prepare for such deadly events and to help reduce potential property damages.

Dust storm events pose significant annual variability, for example dust storms usually occur during May and June and from September to November in southwest United States (Brazel and Nickling 1986). Typically, most dust storms last from several hours to 1 day. Figure 2 shows the reported yearly frequency and percentage of the total time of dust storm in the United States (NOAA 2011). It is estimated that the total time of dust storms was generally less than 90 hours and only taking less than 1% of 1 year, if assumed that each dust storm lasts an average of 2 hours. Therefore, a forecasting system for such events would require different computing and access requirements during different times of a year and even different hours within a day. During dust storm events, massive concurrent user accesses to high-resolution dust storm forecasting are expected. The computing platform supporting such an emergency response system should be able to scale up enough computing

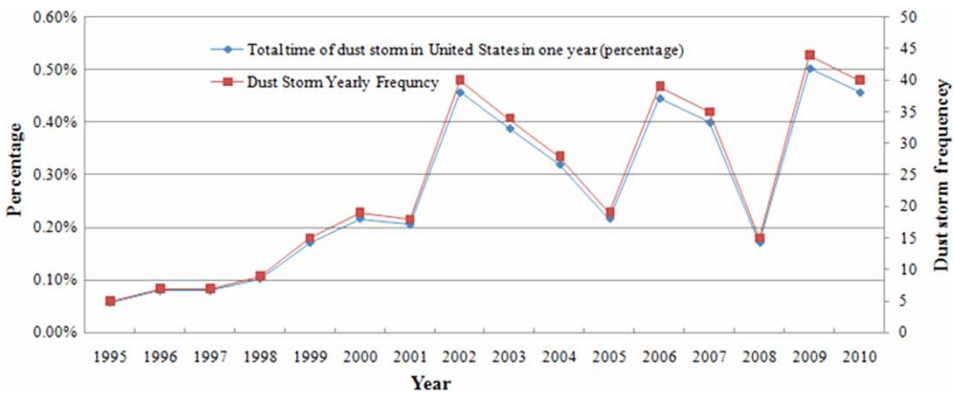


Figure 2. Reported yearly frequency and percentage of the total time of dust storm events in the United States (Data source: NOAA).

resources to conduct relevant science, application, and education for the public to make smarter decision for saving more lives and reducing assets loss. After the emergency response, the access to such information will be down and computing resources should be leveraged for other science, application, and education purposes. Cloud computing is such a computing platform that its computing capabilities can be rapidly and elastically provisioned, in some cases automatically, to scale up and rapidly released to scale down.

For different sizes of dust storm forecasting (sizes are determined by the spatial domain, resolution, and temporal scope), different amounts of computing resources are required (Yang, Wu, et al. 2011). For example, 10 KM spatial resolution requires one CPU core, while 4 KM requires at least 16 CPU cores to complete the computing within 2 hours (Figure 3, Yang, Wu, et al. 2011). Therefore, different amounts of computing resources should be allocated for different problem sizes. This also causes disruptive computing requirements that can be best handled by elastic and on demand cloud computing platforms.

The emergency response systems, such as dust storm forecasting, require elastic large-scale computing pool. On the other hand, with the development of cloud computing technologies, for example virtualization, and the advancement of cloud infrastructure, cloud computing platforms are ready to support data and computing-intensive applications. For example, Amazon EC2 offers cluster instances with 10 Gbps network connection. Each cluster instance has minimum 23Gbytes memory, a clock speed of 2.93 GHz, and two quad-core processors. Such hardware and network configurations are far better than grid computing environments with heterogeneous computing resources and Wide Area Network (WAN) connection, and even better than most private homogenous high performance computing (HPC) cluster configurations. This makes cloud computing a new advantageous computing paradigm to resolve scientific problems traditionally, requiring a special high-performance cluster (Rehr et al. 2010). The traditional cluster environment requires typically several months and usually hundreds of hours of labor to procure, install, and configure new infrastructure. Traditional processes will involve multiple personnel with different technical backgrounds to ensure the success of the configuration. Oftentimes, greater efforts and costs are required to maintain the

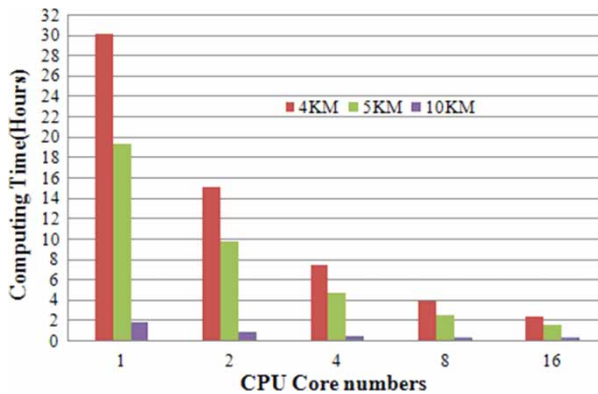


Figure 3. The computing time and required CPU numbers for different resolution forecasting (adopted from Yang, Wu, et al., 2011).

infrastructure while the utilization of the infrastructure is only 10–15% of its full capacity (Marston et al. 2011). Therefore, cloud computing would have a great impact on parallel scientific HPC applications in the near future by facilitating the deployment process while increasing the rate of utilization of computing resources (Yang, Goodchild, et al. 2011).

This paper tests the feasibility of using cloud computing to support dust storm forecasting based on Amazon EC2 cloud computing platform from capacity and cost aspects. Within Amazon EC2, computing resources can be leased on demand, which can satisfy the computational requirements of dust storm forecasting at different times. The paper will introduce how to deploy Earth science models, using NMM-dust as an example, on Amazon EC2 to provide guidance for utilizing cloud computing to support computing and data-intensive Earth sciences and applications. The feasibility of Amazon EC2 to support the parallel execution of a large amount of small regions with scalable cloud resources will be tested through systematic experiments and comparisons between Amazon EC2 and traditional HPC cluster in the aspects of deployment time, supporting high-resolution forecasting, concurrent computing, and loosely coupled nesting approach, and cost savings. This paper presents a complete study to answer the question of ‘could cloud computing enable the computability of dust storm forecasting, a typical parallel HPC application.’

Section 2 reviews the research efforts for utilizing cloud computing to support geoscience applications. Amazon EC2, a popular platform to host applications, and the loosely coupled model nested approach will also be described in Section 2. The configuration of a cloud HPC environment through Amazon EC2 cluster instances (<http://aws.amazon.com/ec2/hpc-applications/>) and deployment of the dust storm model to such a virtualized cluster are introduced in Section 3. Section 4 reports experiments and comparisons between traditional HPC cluster and EC2 that test the feasibility of Amazon EC2 to satisfy the computational requirements of the dust storm forecasting. Section 5 introduces a case study using the 5 July 2011 Phoenix dust storm as an example. Section 6 concludes and discusses future research directions and issues for cloud computing to support Geosciences.

2. Related work

2.1. Cloud computing for geoscience

Several academic research programs have investigated and developed technologies and infrastructures for cloud computing, such as Nimbus (Nimbus 2011), OpenNebula (OpenNebula 2010), and Virtual Workspaces (Keahey et al. 2005). Nimbus is an open-source toolkit that allows a user to turn their clusters into an Infrastructure-as-a-Service (IaaS) cloud. OpenNebula is an open-source toolkit used to easily build private, public, and hybrid cloud.

Cloud computing is also now a key strategy adopted by many commercial IT vendors, ISPs, and telecom service providers. Through these providers, many cloud services are available over the Internet. For example, the most popular cloud service provider, Amazon, offers several types of cloud services from IaaS to Platform as a Service (PaaS) (Armbrust et al. 2009). RESERVOIR is an IBM and European Union joint research initiative for cloud computing that will enable massive scale

deployment and management of complex IT services across different administrative domains, IT platforms, and geographic regions (Reservoir 2010). Verizon's CaaS allows customers to pay for data-center resources such as storage and application dynamically based on the amount of resources consumed (CaaS 2010). Google has published several research papers from 2003 to 2006, outlining their PaaS cloud computing. The platform, called Google App Engine (GAE), was released to the public as a service in 2008 (GAE 2008). Increasingly, more and more cloud computing providers cooperated with each other to construct more powerful cloud services. For example, NASA and RACKSPACE are joined by leaders from across technology industries, such as CITRIX, DELL, NTT DATA, and RIGHTSCALE, to work on the OpenStack project (OpenStack 2010) to create open-source code, badly needed standards, creating common ground for the benefit of both cloud providers and cloud customers. enStratus provides PaaS to enable users to access and manage different IaaS and PaaS (enStratus 2011)

A very important feature of cloud computing is the abstraction of the implementation, meaning that the client is unaware of application deployment details (e.g. where is the hardware located and how is it configured to run the application). This enables cloud computing to provide smart/broader discovery, enhanced access to data and services, on-the-fly integrations of applications, and transparent platforms for model running so that geoscience scientists can focus on research without considering the underlying mechanism to implement the time-consuming computational task (Evangelinou and Hill 2008). This is an objective that geoscience scientists have dreamed about (Nebula 2010; Walker 2008; Wang and Liu, 2009; Yang et al. 2010).

Many studies have been conducted to explore the feasibility of utilizing cloud computing to support geoscience applications and learn how to best adapt to this new computing paradigm. Since cloud computing can leverage scalable computing resources for the dynamic traffic demands, more and more web applications have been migrated to the cloud platform. Huang et al. (2010) utilized cloud computing for geosciences and show that EC2 cloud computing platform could provide geospatial applications with good (a) scalability and (b) reliability, and (c) reduce duplicated efforts among geosciences communities. Evangelinou and Hill (2008) found that cloud computing could provide a potential solution to support responsive on-demand, small-sized HPC applications.

2.2. Amazon EC2

As a central part of Amazon's cloud services (AWS 2011), Amazon EC2 allows users to deploy scalable resources on demand. Amazon EC2 is a typical IaaS cloud service. Based on Xen and other virtualization technologies (Barham et al. 2003), EC2 enables users to create a virtual machine (VM) from an Amazon Machine Image (AMI), which Amazon calls an 'instance.' AMI is a bootable VM root image with various OS and any software desired to create a VM. Presently, EC2 offers a number of different instance types to meet computing needs with each instance providing a predictable amount of dedicated compute capacity (CPU power, memory, and storage). Amazon classifies these EC2 instances into six categories, including standard instances, Micro Instances, High-Memory instances, High-CPU Linux instances, Cluster Compute Instances, and Cluster GPU Instances. Different

categories are suitable for different types of applications. In this paper, cluster instances will be utilized to construct the cloud cluster environment.

2.3. Loosely coupled nested models

Current constraints on computing power and the scalability of parallel system preclude an immediate solution to perform the high-resolution dust storm forecasting over a large area (Huang et al. 2013); more strategies should be explored to support an operational system capable of performing large-area and high-resolution forecasting (Kuligowski and Barros 1999, Yang et al. 2010). One solution is to nest a finer-scale grid or succession of grids within a model to enhance the resolution over specific areas of interest while moderating the required computational cost (Anthes 1983, Kuligowski and Barros 1999). Such an approach is called nested models, or more commonly named as high-resolution limited-area models. However, usually the tightly coupled nesting approach is used (Neilson et al. 2005). Within this approach, a model is able to run with multiple resolutions. For example, Figure 4a shows that a model can run in two scales. Parent domain is low resolution while the two nest domains run with higher resolution.

However, achieving those two tightly coupled approaches requires extensive modification of the models to enable the models to be able to be executed with different resolutions (Michalakes et al. 1998). In addition, this approach requires us to know where the high-resolution area is before running the model (Constantinescu, Sandu, and Carmichael 2008). For a real-time dust storm forecasting system, users are not aware of where the dust storm will occur, and therefore traditional tightly coupled nested models are not suitable.

Huang et al. (2013) proposed a loosely coupled nested solution to overcome this issue while enabling some sub-regions with high resolution over a large domain. In the loosely coupled approach, two dust storm simulation models instead of one model will be used. A coarse dust storm model (ETA-8bin) is able to simulate large area but with low resolution, and a fine model NMM-Dust is able to produce high resolution results but with smaller area (Figure 4b). The ETA-8bin is first executed with low spatial resolution (50 km) to identify hotspots with high dust concentration over a large area. Another fine model, NMM-dust model, will run with high resolution (3 km) for those hotspots domains on different groups of computing resources with each group performing one hotspot forecasting. In this way, the hotspots with much smaller regions can be completed with much less computing resources.

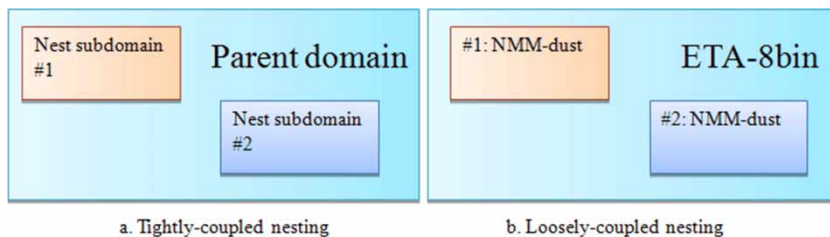


Figure 4. Nested modeling approaches.

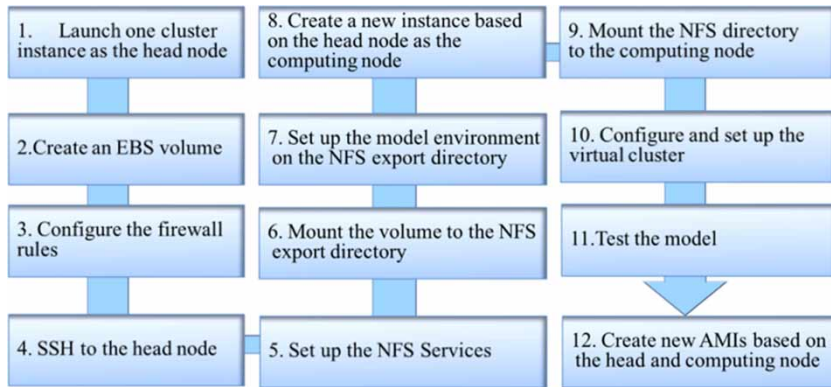


Figure 5. The process for deploying dust storm models onto Amazon EC2 (revised from Huang et al., 2010)

Loosely coupled nested models require a large computing pool to run various hotspots identified by coarse model concurrently to achieve the best performance. This will cause computing spike requirements that can be best handled by elastic and on-demand computing platform. Section 5 will test, as an example, if the cloud computing can handle the computing for multiple sub-regions with the Phoniex, 5 July 2011 dust storm event.

3. Deploying parallel model onto Amazon EC2

Figure 5 illustrates deploying dust storm models onto Amazon EC2 high-performance platform. Amazon provides users cluster instance from a special Elastic Block Storage (EBS)-backed AMI using Hardware Virtual Machine (HVM) virtualization. (1) One cluster instance can be launched based on the dust storm forecasting requirements as the head node. The AMI could be an existing public AMI, or created by users from scratch.

After the instance is loaded with the AMI and booted up, users are able to interact with the head node instance. (2) Users can create an EBS volume to attach the head node instance for keeping the dust storm model and required package suits to run the model, such as, MPICH2 software to schedule the dust storm simulation. Such a separate volume has two benefits: (a) one is to restore the dust storm simulation system from the volume in case the current head node instance crashes, and (b) another is that volume could be any size from 1 GB to 1 TB in size. Because the dust storm simulation is a data-intensive application, hundreds or thousands of datasets could be produced. Therefore, such an EBS volume would be perfect to resolve the storage capability problems.

(3) The next step is to setup the firewall for the group of instances, and the system administrator should fully control permissions to the instances and the instances should be able to communicate with one another over a range of ports while performing the simulation to exchange the data. This can be setup through Amazon's command line tools or AWS Management console. (4) Therefore, users can login to the instance to get full root access through remote access method SSH (Secure Shell)

after authorizing the network access by opening the port 22 to enable the SSH access performed on the previous step.

(5) After logging in, users can explore and play with the system however they like: setup the required environment to host a web application or run a model. In this case, the Network File System (NFS) service, should be setup to enable all other computing instances to share the package suits. With NFS only the head node should setup the software environments and configure the model while other computing nodes can share this environment without installing and configuring. After (6) mounting the EBS volume to the NFS export directory, (7) the package suits, including Fortran and MPICH2 in this case, to enable the model to run can be installed, and NMM-dust dust storm model can be deployed on the NFS export directory.

(8) After successfully setting up the head node environment, users can create a new AMI based on the head node and launching a VM as computing node from this new AMI. (9) The computing nodes can share the NMM-dust model environment by mounting the NFS export directory to the local directory like using a local storage. (10) As long as the firewall between the head node and computing node is properly setup, and the middleware MPICH2 is properly configured, the cluster environment with only one head and computing nodes should be able to communicate with each other and (11) the model NMM-dust should be able to run successfully.

(12) Finally, two new AMIs can be created based on the running head and computing nodes. In this way, the system can be restored in a very fast fashion if the head node or computing node is crashed by launching an Amazon EC2 cluster instance from those new AMIs. In addition, geoscience communities can build their own dust storm forecasting system from the two AMIs. This would greatly reduce the duplication of efforts among organizations as other organizations can run the model based on the work without putting forth great efforts to compiling, installing, and configuring the NMM-dust on their own, which is a very complex process, taking several days and even several weeks for a new user. Traditionally, various researchers and scientists have used the available geoscience data to build a large number of complex algorithms, models, and applications tailored to their specific studies (Votava et al. 2002). The goals achieved by these algorithms, models, and applications are not always completely different, despite the differences in research objectives. If we build the functions or applications as cloud services to share with others, duplications can be eliminated. Hence, cloud computing would greatly facilitate the geosciences in reusing and sharing functionalities and knowledge across geoscience communities.

Amazon EC2 offers a highly reliable environment for the dust storm models because the service runs within Amazon's proven network infrastructure and data centers. The Amazon EC2 Service Level Agreement (SLA) guarantees 99.95% availability for all Amazon EC2 regions, including US Standard, EU (Ireland), US west (Northern California), and Asia Pacific (Singapore). An additional EBS volume to keep dust storm models and running environment has been configured for added reliability. Since the EBS data volume can be attached to another instance, the dust storm simulation system is therefore protected from instance termination or failure. Therefore, Amazon EC2 improves the reliability of geoscience applications.

Table 1. The configuration of EC2 and our HPC cluster computing platform.

Items	EC2 cluster compute instances	HPC cluster
Nodes	Theoretically unlimited	28
CPU	2.93 GHz	2.33 GHz
Numbers of processors	2	2
Numbers of cores/processors	4	4
Support hyper-threading	Yes (resulting in 16 CPU cores in total)	No
Total memory	23 GB	16 GB
Network bandwidth	10 Gbps	1 Gbps
Virtualization	Hardware-based virtualization	N/A

4. Experiments and results

4.1. Computing environment

Table 1 shows the configuration of EC2 and our HPC cluster. HPC cluster has 28 computing nodes including the head node, and all nodes are connected through local area networks (LANs with 1Gbps). Each node has 16 Gbytes memory and two quad-core processors (8 physical cores) with a clock frequency of 2.33 GHz, a peak performance of 7.6 Gflops/core and a sustained performance of 1 Gflop/core.

EC2 cluster compute instances provide proportionally high CPU with increased network performance and are well suited for HPC applications and other demanding network-bound applications. Cluster Compute Quadruple Large instance has 23 GB memory, 8 CPU cores and supports hyperthreading resulting in 16 CPU cores in total, 1690 GB of local instance storage, 64-bit platform, 10 Gigabit Ethernet. Unique to Cluster compute instances is the ability to group them into clusters of instances with the high speed network connection for use with HPC applications. This is particularly valuable for those applications that rely on protocols like Message Passing Interface (MPI) for tightly coupled inter-node communication (<http://aws.amazon.com/ec2/hpc-applications/>).

4.2. Elastic resources for disruptive requirements

Table 2 shows the average time spent on deploying and operating dust storm model on Amazon EC2 and local server. The EC2 only requires several hours to several days for the first time model deployment depending on if customizing the model based on a public image or hardening the image from scratch. If users chose to use the public available AMI rather than harden the image from the scratch, users are not required to configure and install the basic software package and required dependent package suits, such as the scheduling software MPICH2. Also, the configuration time for the deployment environment could be reduced significantly from possibly several days to several hours. Taking into account the time for purchasing the servers and configuring the hardware and software, the traditional HPC cluster could require several weeks to setup the deployment environments for the first time.

After the first time deployment, EC2 users can launch many VMs including models using the customized image. And it usually takes approximately one minute to start, stop, and resume those Amazon EC2 instances which perform the same as local servers (Table 2). The deployment and launching time results indicate that

Table 2. Average time spent on deploying dust storm model on Amazon EC2 and local time.

Items	Options for cloud environment	Amazon	Local	Description
Hardware configuration		None	~4 weeks	Time for purchasing the servers
Setting up basic software configuration	Use a public AMI with OS installed	None	~1 weeks	OS, firewall, and cluster environment, such as networking configuration and cooling
Deploying dust storm model	Harden image from scratch	~1 week		
	Use a public AMI with most required dependent software package installed	~2 hours	~1 days	Setting up the model for local server
	Harden image from scratch	~1 days		
Instance startup time		45 s	120 s	Start up the virtual machine or local machine
Instance stop time		57 s	60 s	Stop the virtual machine or local machine
Instance resume time		45 s	N/A	Resume the paused virtual machine; local server generally does not support this option
Total time needed for the first time deployment	Customize a public AMI	~2 Hours	~ 5 weeks	Only require time to customize the model based on the public image under EC2
	Harden image from scratch	~1 week		

cloud computing can help with disruptively computing requirements for dust storm forecasting, ready and run in a few minutes.

4.3. High resolution forecasting and computing intensive support

In this experiment, different amounts of Amazon EC2 and HPC computing nodes are used to test the computing time for a regional domain and short-term but high-resolution forecasting. Figure 6 shows the computing time and computing time difference between EC2 and local HPC for 3 KM resolution, 4.8×4.8 degree domain size, and 3-hour forecasting with 1, 4, and 8 Amazon EC2 and HPC computing nodes with different process numbers. It is observed that the Amazon EC2 node performs better than the HPC node as it has lower CPU speed (2.33 GHz vs 2.93 GHz), and Amazon EC2 node supports hyper-threading resulting in 16 CPU cores per node. When more nodes are utilized, HPC cluster nodes perform better. This is because HPC has relatively low communication overhead, as it is not based on virtualization technology. However, the time differences are usually less than 0.05 hours (3 minutes) when process numbers are more than 96, which are typical for

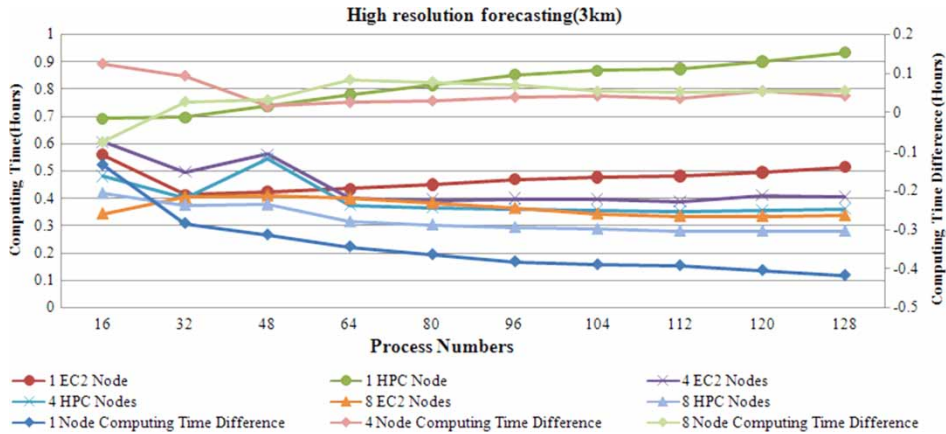


Figure 6. Computing time and difference for 3 KM resolution, 4.8×4.8 degree domain size, and 3-hour forecasting with 1, 4, and 8 Amazon EC2 and our HPC computing nodes with different process numbers.

large-scale computing with hundreds of CPU cores utilized. Therefore, this result indicates that cloud computing is able to perform high resolution with reasonable computing or elastic amount of computing resources.

4.4. Concurrent computing support

Generally, if users request to perform a regional dust storm forecasting, the system would first check the storage if the forecasting results are available. If not, the system would revoke the forecasting process. Although it could reduce the computing requests by responding with the previous simulation results, there will be concurrent forecasting requests for different areas without previous simulation results.

In this experiment, local cluster and cloud computing platform are tested and compared with 15 computing tasks with different domain sizes for 24-hour forecasting. In the cloud platform, 15 tasks are submitted to 15 Amazon EC2 instances with each Amazon instance serving as a computing platform to run one task simultaneously. At the same time, those 15 forecasting tasks are submitted to the HPC cluster at the same time. Figure 7 shows the computing time for those 15

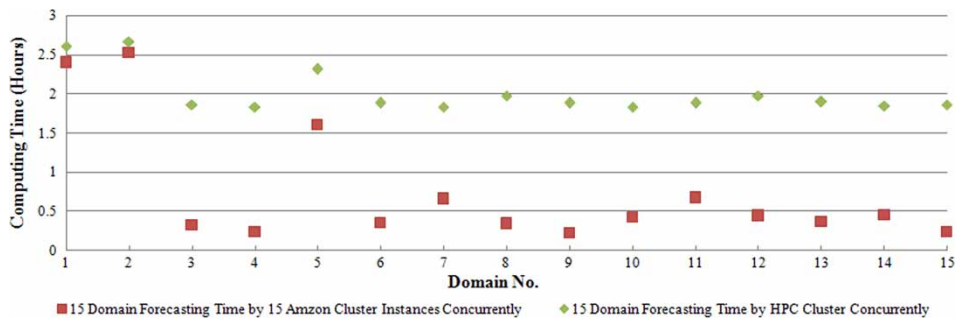


Figure 7. NMM-dust execution time for 15 forecasting tasks on Amazon EC2 and HPC cluster.

forecasting tasks by both Amazon EC2 and HPC cluster. It is observed that Amazon instances can complete most of those tasks with less time with the dedicated computing platforms for each task. The results indicate that cloud computing has great potentials to resolve the concurrent intensity of the computing demanding applications.

5. Case study

5.1. Phoenix dust storm event

The state of Arizona is home to some of the most spectacular displays of blowing dust in the United States. Dangerous dust storms impact the state several times per year and cause major vehicle accidents, property damage, injuries, and the loss of lives (Shoemaker and Davis 2008). The largest of these dust storms is called a ‘haboob,’ which is most common in the central deserts of Arizona during its summer season, with the frequency of occurrence peaking in late July and early August. The city of Phoenix, AZ, experiences on average about three haboobs per year during the months of June–September (Shoemaker and Davis 2008). Figure 8 shows a classic haboob that moved through the Phoenix area on 5 July 2011. The dust storm starts around 5 July 2011, 7:30 PM, Mountain Daylight Time (MDT). Using the UTC (Coordinated Universal Time) time, the dust storm event would start on 6 July 2011 at 1:30 AM UTC.



Figure 8. Pictures of a haboob that hit Phoenix, Arizona, on 5 July 2011 (courtesy of Meteorology News 2011).

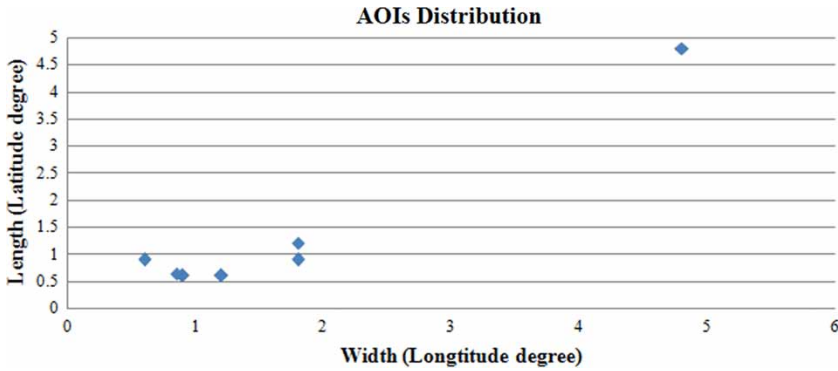


Figure 9. AOIs distribution identified by the ETA-8bin on 5 July 2011.

5.2. Performance test

Elastic Amazon EC2 cluster instances are utilized to enable the loosely coupled nesting approach to its full extent. During the loosely coupled nesting test, seven AOIs are identified by the coarse model ETA-8bin. Figure 9 shows the width and length distribution of those seven AOIs. It is observed that they almost distributed within 2×2 degree while only one sub-region has the area of 4.8×4.8 degree. Therefore, seven cluster instances are launched and each instance is responsible for handling the simulation for one AOI.

Figure 10 shows the execution time for each instance by seven Amazon clusters, with each instance simulating one AOI region for 24-hour forecasting. The results reveal that most AOIs can be successfully completed within 1 hour for 24-hour forecasting. However, the largest region (AOI No.1) requires about 3.2 hours for finishing the computation. Therefore, 2, 4, and 8 cluster instances are used to test the computability of this sub-region. Figure 11 shows the performance of EC2 when utilizing 1, 2, 4, and 8 instances for 24-hour forecasting over the first sub-region (4.8×4.8 degree). It is observed that eight EC2 nodes can reduce the computing to around 2.6 hours. These results show that cloud computing is able to complete the computation of large area forecasting within a relatively small time period with the loosely coupled nesting approach (2.6 hours for this case).

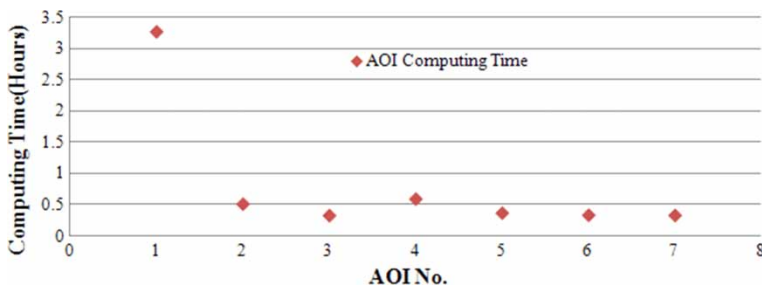


Figure 10. Seven Amazon cluster instances are launched to run seven AOIs in parallel with each instance simulating one AOI region for 24-hour forecasting.

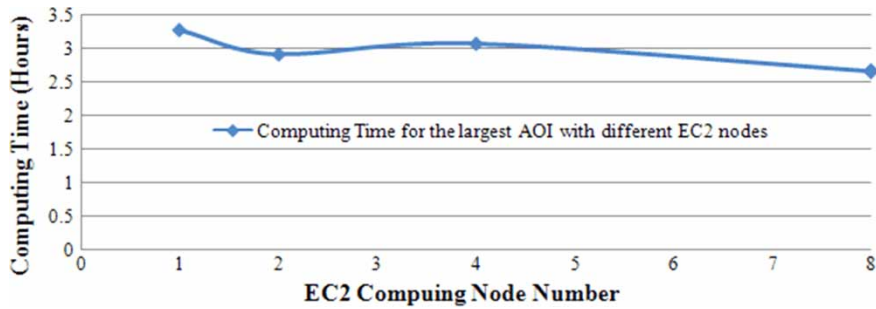


Figure 11. Computing times for the largest AOI with different computing nodes for 24-hour forecasting.

5.3. Economy

For a dust storm operational forecasting system, we would usually reserve an EC2 instance for a long-term forecasting with a much lower price, and it costs only \$0.16 per hour for general user access and low-resolution forecasting (Table 3). When the dust storm events occur, a large group of instances would be leveraged to respond to concurrent users and high-resolution forecasting. Table 3 shows the cost comparison between AWS cloud services and the HPC cluster. It is observed that the hourly cost of the cluster is much higher than that of EC2 cloud platform when there is no dust storm event occurring and only one reserved instance is needed. Under this situation, the cluster's hourly costs are more than 896 times than those of EC2 cloud platform.

Typically, there are around 45 dust storm events occurring in the United States in total per year (NOAA 2011). The yearly cost of local cluster is around 12.7 times than that of the EC2 cloud platform if 28 EC2 instances (with 400 CPU cores) are leveraged to handle the high resolution and concurrent computing requirements for a duration of 48 hours. With much less costs, the cloud platform can provision the same computing power as a local cluster within a shorter time period in responding to the dust storm events. These cost comparison results indicate that it is economic to use cloud computing by maintaining low access and resolution forecasting while invoking a large scale of computing power to perform high-resolution forecasting for large public access when needed.

6. Conclusion

Dust storm events pose an inter-annual variability and are typical disruptive events. The computing pool to host an operational dust storm forecasting system should also support the disruptiveness by scaling up to respond to potential high resolution and concurrent computing requests when dust storm events happen and by scaling down when no dust storm events occur to save the energy and costs. This paper discusses and performs a systematic study to find if cloud computing platforms, using Amazon EC2 as an example, are capable of supporting such disruptive event forecasting.

We found the following: (1) cloud computing is ready to support dust storm forecasting by being capable of leveraging a large group of computing power in a few

Table 3. Cost comparison between AWS services and local HPC cluster.

Time	Items	Types	Amazon EC2	HPC cluster	Assumption
Fixed	On-time cost		None	~\$200,000	Spends around \$200,000 to purchase 28 computing nodes in 2007
Hourly	Network cost		\$0.16/hour for data transferring	None (included in the maintenance fee)	Use 1T data transfer out per hour
	Storage		\$0.13/hour	None (included in the first purchase)	Use 1T Amazon EBS storage for the EC2 cloud platform
	Computing power	Not reserved	\$1.3/hour for one instance	~\$138.8 per hour	HPC can be used for 5 years, and then each month's cost would be 138.8 ($200000/(5 \times 12 \times 24) = 138.8$)
		Reserved	\$0.16/hour for one instance		Reserve an EC2 quadruple extra large cluster instance that costs \$1450 per year for handling low access and low-resolution forecasting
	Maintenance cost(cooling, system, sys admin maintenance, room, etc.)		None	\$1.41/per hour	\$1 for cooling, network, and room fee per hour, and \$40/hour for paying a part-time sys admin to check and maintain the HPC for 4 hours per day, and then the maintenance costs for HPC would be: $1 + 40 \times 4/24 = 1.41$ per hour
Yearly	Total	Not reserved	\$85,860	\$102,353.3	There are around 45 dust events each year, and 28 EC2 nodes are launched to respond to the public for 48 hours for each dust storm forecasting
			$((0.16 + 0.13 + 1.3) \times 45 \times 28 \times 24) = 96163.2$	$((138.8 + 1.41) \times 365 \times 24 = 1228240)$	
		Reserved	\$1450		
	Total		97613.2		

Note: The cloud computing cost is based on the EC2 quadruple extra large cluster instance with 8 CPU cores of EC2 to run the dust storm model. If negotiating with the cloud provider, discount would be obtained to launch a large group of instance for scientific computing purpose.

minutes to satisfy the disruptive computing requirements of dust storm forecasting (Section 4.2). (2) On comparison of performances of EC2 instances and our HPC cluster using different computing instances, the cloud computing platforms also show a great potential to perform high-resolution dust storm forecasting (Section 4.3). (3) On comparison of performance of concurrent computing, the EC2 demonstrates that cloud platform is capable of handle massive concurrent computing requirements. (4) The research results of real case study of dust storm event in Phoenix 05 July 2011 demonstrates a great potential to support real dust storm forecasting for a large geographic domain (Section 4.4). (5) In addition, it is economic to utilize cloud computing at reduced cost by maintaining low access and resolution while invoking a large scale of computing power to perform high-resolution forecasting for large public access when needed (Section 5.1).

There is an urgent need to investigate how geoscience can leverage cloud computing to improve the performance and enable the computability of scientific problems, and hide the complexity of the computing infrastructure so that scientists can focus on scientific problems. However, geoscience applications have special requirements that cannot be automatically supported by generic cloud computing platforms, because most geospatial algorithms and applications are not designed to leverage multiple CPUs and be delivered through the Internet as a service. Most importantly, both the geoscience and the cloud computing environments are spatiotemporal intensive. Geoscience phenomena are complex processes and geoscience applications often take a variety of data as input with a long and complex workflow. SCC (Spatial Cloud Computing) can be researched with spatiotemporal constraints and drivers considered to construct the next generation cloud computing platform to better support complex geoscience applications, such as dust storm simulation (Yang, Goodchild, et al. 2011).

However, it would be a critical challenge to build a SCC platform to deliver geoscience applications to the end users as a transparent service to support massive numbers of users. For example, the middleware used to schedule computing tasks on a cloud computing platform is mostly not developed for geoscience applications and does not take the spatiotemporal principles and patterns into consideration. Such middleware should be reengineered to support spatiotemporal processing. Also, spatiotemporal patterns of phenomena, data, services, models, and computing resources must be utilized to optimize the performance of geospatial processing and applications (Yang, Goodchild, et al. 2011). Such an effort would also help to construct a better geospatial cyber infrastructure (Yang et al. 2010) and a spatial cloud computing platform (Yang, Goodchild, et al. 2011). In addition, the success of spatial cloud computing depends on many factors, such as the outreach of spatial cloud computing to geospatial scientists and engineers who can employ the spatiotemporal principles to best design, construct, and deploy the cloud solutions for their scientific studies, applications and operations.

Acknowledgements

Research reported is supported by NSF (CSR-1117300 and IIP-1160979), NASA (NNX07AD99G), and Microsoft Research.

References

- Anthes, R. 1983. "Regional Models of the Atmosphere in Middle Latitudes." *Monthly Weather Review* 111: 1306–1335.
- Armbrust, M., A. Fox, and R. Griffith, et al. 2009. *Above the Clouds: A Berkeley View of Cloud Computing*. Tech. Rep. UCB/EECS-2009-28. Berkeley, CA: EECS Department, University of California.
- AWS. 2011. "Amazon Web Service" [online]. Accessed June 10. <http://aws.amazon.com>
- Barham, P., B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield. 2003. "Xen and The Art of Virtualization." In *Proceedings of the Nineteenth ACM Symposium on Operating Systems Principles* (Bolton Landing, NY, USA, October 19–22, 2003). SOSP '03, 164–177. New York: ACM. Accessed June 10. doi:<http://doi.acm.org/10.1145/945445.945462>
- Brazel, A. J., and W. G. Nickling. 1986. "The Relationship of Weather Types to Dust Storm Generation in Arizona (1965–1980)." *Journal of Climatology* 6 (3): 255–275.
- CaaS. 2010. "Computing as a Service" [online]. Accessed June 10. <http://www.verizonbusiness.com/products/itsolutions/caas/>.
- Constantinescu, M., A. Sandu, and G. R. Carmichael. 2008. "Modeling Atmospheric Chemistry and Transport with Dynamic Adaptive Resolution." *Computational Geosciences* 12 (2): 133–151.
- enstratus. 2011. "Enstratus Company" [online]. Accessed November 10. www.enstratus.com
- Evangelinos, C., and C. Hill. 2008. "Cloud Computing for Parallel Scientific HPC Applications: Feasibility of Running Coupled Atmosphere-Ocean Climate Models on Amazon's EC2." *Ratio* 2 (2.40): 2–34.
- GAE. 2008. "Google App Engine" [online]. Accessed June 10. <http://appengine.google.com>
- Huang, Q., C. Yang, H. Wu, W. Li, J. Xie, and C. Yang. 2010. "GeoInformation Computing Platforms." In *Advanced GeoInformation Science*, edited by C. Yang, D. Wong, Q. Miao, and R. Yang, 79–127. Boca Ration, FL: CRC Press, Taylor & Francis Group.
- Huang, Q., C. Yang, K. Benedict, A. Rezgui, J. Xie, J. Xia, and S. Chen. 2013. "Using Adaptively Coupled Models and High-performance Computing for Enabling the Computability of Dust Storm Forecasting." *International Journal of Geographic Information Science*. doi:10.1080/13658816.2012.715650
- Keahey, K., I. Foster, T. Freeman, and X. Zhang. 2005. "Virtual Workspaces-Achieving Quality of Service and Quality of Life in the Grid." *Scientific Programming* 13 (4): 265–275.
- Kuligowski, R. J., and A. P. Barros. 1999. "High-Resolution Short-Term Quantitative Precipitation Forecasting in Mountainous Regions using a Nested Model." *Journal of Geophysical Research* 104 (D24): 31, 553–31, 564. doi:10.1029/1999JD900938.
- Marston, S., Z. Li, S. Bandyopadhyay, J. Zhang, and A. Ghalsasi. 2011. "Cloud Computing – The Business Perspective." *Decision Support Systems* 51: 176–189.
- Meteorology News. 2011. "Meteorology News" [online]. Accessed December 10. <http://www.meteorologynews.com/2011/07/05/dust-storm-shrouds-phoenix-in-zero-visibility/>
- Michalakes, J., J. Dudhia, D. Gill, J. Klemm, and W. Skamarock. 1998. "Design of a Next-Generation Regional Weather Research and Forecast Model." In *Towards Teracomputing*, edited by W. Zwiefhofer and N. Kreitz, 117–124. River Edge, NJ: World Scientific.
- Nebula. 2010. "NASA Cloud Computing Platform" [online]. Accessed March 10. http://www.nasa.gov/offices/ocio/ittalk/06-2010_cloud_computing.html
- Neilson, R. P., L. F. Pitelka, A. M. Solomon, R. Nathan, G. F. Midgley, J. M. V. Fragoso, H. Lischke, and K. Thompson. 2005. "Forecasting Regional to Global Plant Migration in Response to Climate Change." *American Institute of Biological Sciences* 55 (9): 749–759.
- Nimbus. 2011. "The Nimbus Cloud" [online]. Accessed June 10. <http://www.nimbusproject.org/>
- NOAA. 2011. "Dust Storm Database" [online]. Accessed October 30, 2011. <http://www4.ncdc.noaa.gov/cgi-win/wwwcgi.dll?wwevent~storms>.
- OpenNebula. 2010. "OpenNebula Project" [online]. Accessed June 10, 2010. <http://www.opennebula.org>
- OpenStack. 2010. "OpenStack Project" [online]. Accessed June 10. <http://openstack.org/>
- Rehr, J. J., F. D. Vila, J. P. Gardner, L. Svec, and M. Prange. 2010. "Scientific Computing in the Cloud." *Computing in Science & Engineering* 12 (3): 34–43.

- Reservoir. 2010. "Reservoir Project" [online]. Accessed June 10. <http://www.reservoir-fp7.eu/>
- Shoemaker, C., and J. Davis. 2008. "Hazardous Weather Climatology for Arizona" [online]. Noaa Technical Memorandum. Accessed June 10. <http://www.wrh.noaa.gov/wrh/techMemos/TM-282.pdf>
- Votava, P., R. Nemani, C. Bowker, A. Michaelis, and J. Coughlan. 2002. "Distributed Application Framework for Earth Science Data Processing." In *Proceedings of IEEE International of Geosciences and Remote Sensing Symposium (IGARSS'02)* 2, 24–28, Toronto. Washington, DC: IEEE Computer Society.
- Walker, E. 2008. "Benchmarking Amazon EC2 for High-Performance Scientific Computing." *Login* 33 (5): 18–23.
- Wang, S., and Y. Liu. 2009. "TeraGrid GIScience Gateway: Bridging Cyberinfrastructure and GIScience." *International Journal of Geographical Information Science* 23 (5): 631–656.
- Yang, C., M. Goodchild, Q. Huang, D. Nebert, R. Raskin, Y. Xu, M. Bambacus, and D. Fay. 2011. "Spatial Cloud Computing: How Could Geospatial Sciences Use and Help to Shape Cloud Computing." *International Journal on Digital Earth* 4 (4): 305–329.
- Yang, C., R. Raskin, M. F. Goodchild, and M. Gahegan. 2010. "Geospatial Cyberinfrastructure: Past, Present and Future." *Computers, Environment, and Urban Systems* 34 (4): 264–277.
- Yang, C., H. Wu, Q. Huang, Z. Li, and J. Li. 2011. "Using Spatial Principles to Optimize Distributed Computing for Enabling the Physical Science Discoveries." *Proceedings of National Academy of Sciences of USA* 108 (14): 5498–5503.