International Journal of
Applied Mathematics,
Electronics and Computers

ISSN:2147-8228 www.atscience.org/IJAMEC

Advanced Technology and Science

Original Research Paper

Energy-Aware Cloud Computing Simulators: A State of the Art Survey

Ammara Sajjad, Aleena Aqdus Khan, Muhammad Aleem*

Accepted : 12/06/2018 **DOI 10.18100/ijamec.2018241182**

Abstract: Cloud computing is an emerging technology that offers pay-per-use IT services around the world. The daily increase in Cloud infrastructure and modern IT demands (such as web applications, scientific, and business) results in large-scale data centers that lead to additional electricity consumption. The high energy consumption causes a high operating cost and involves a large amount of carbon dioxide emissions that are harmful to the atmosphere. Many academic, governmental, and industrial studies show that energy consumed by computers and different communication units in a data center plays a significant role in the increase of data center operating costs. It is difficult to calculate the performance, security problems, and energy consumption in a true Cloud computing infrastructure. Therefore, in recent years, several simulation tools have been introduced to help researchers for the analysis of a Cloud computing environment. Cloud simulators need to perform simulation tests to reduce the various complexities of Cloud computing environments. Some Cloud simulators have been specially created to test the performance of Cloud computing environments. This study presents the comparative analysis of the state-of-the-art energy aware Cloud simulators. For comparison, several characteristics, functions, and models are employed for most famous energy aware Cloud simulators such as Green Cloud, CloudSim, iCanCloud, CloudAnalyst, NetworkCloudSim, and CloudReports.

Keywords: *Cloud Computing, Cloud simulators, Energy-aware Cloud;*

1. Introduction

Cloud computing is a technology that utilizes computing resources and Internet to provide scalable, on-demand, pay-as-you-go computing and storage services [1]. Cloud computing uses a large number of heterogeneously distributed resources to offer many services with unique requirements for *Quality of Service* (QoS) [2]. Some of the most prominent Cloud computing platforms are Microsoft Azure [18], Aneka, Amazon EC2 [17], Google App Engine [19], and GoGrid.

Generally, Clouds are classified as *public*, *private*, and *hybrid* Clouds [3]. In public Cloud model, a Cloud service provider offers Cloud services over the Internet. Public Cloud services are sold on demand, usually per minute or per hour basis. Customers only pay for the CPU, memory, and bandwidth they consume. Private Clouds and their infrastructure are owned and accessible by specific organizations [4]. By using a set of physical resources private Cloud provides computing power and other services within a specific organization or to the fixed small community of users. Moreover, users can only access the Cloud resources and services from a single organization, which provides more control and privacy. A hybrid Cloud is a combination of private and public Cloud [5]. Moreover, due to the availability issues of a single Cloud infrastructure, many efforts have been made towards multi-Clouds [6], which emphasize on the federation of different Clouds infrastructures [7, 8].

In addition, the most common services provided by the Cloud are classified as *Software as a Service* (SaaS), *Platform as a Service* (PaaS), and *Infrastructure as a Service* (IaaS) [9]. In SaaS, users do not need to install software on their local machines [10]. PaaS provides a platform for customers who can develop their propriety applications on top of this platform [11]. IaaS provides an infrastructure for different purposes, such as storage systems, computing, and IT resources [12, 13]. Users of IaaS can adopt the available resources according to the required platform and application of their choice.

Cloud computing [14] focus on evolving data centers as a dynamic, scalable, and flexible platform for on-demand service and application delivery. This is done by revealing data-center capabilities as a network of virtual services for example user interface, database, hardware, and application logic so that the users can access and deploy applications over the Internet [15]. IT companies with new application service ideas are no need to make large expenditures on the hardware and software infrastructure. By using the Cloud as a platform for application hosting, IT companies are relieved of configuring basic hardware and software infrastructures. Therefore, for application service companies focus more on innovating and creating business values [16].

Some of the basic and developing Cloud-based application services in real time are content delivery, social networks, data processing, and web hosting. These applications have a different composition, configuration, and implementation requirements. In real Cloud computing environments such as (Microsoft Azure [18], Google App Engine [19], and Amazon EC2 [17]) and the evaluation of the provisioning policies for different applications under the temporary condition is very difficult. The use of real infrastructures, for benchmarking application performance (cost benefits, performance analysis) in variable conditions (workload, availability) is often limited by the inflexibility of the infrastructure. Therefore, in real infrastructures, an extremely difficult task is the reproduction of the reliable results. Furthermore, in extensive Cloud infrastructure reconfiguration of many experiments is a tedious and time-consuming task. The

Department of Computer Science, Capital University of Science and Technology, Islamabad, 44000, Pakistan
** Corresponding Author: Email: aleem@cust.edu.pk*

existing conditions in Cloud result in these limitations that are not under the control of application service providers. Therefore, benchmarking experiments cannot be performed in a repeatable, reliable, and scalable manner in a real Cloud environment. Therefore, many research efforts have been focused on developing Cloud simulators which are useful for benchmarking, analysis, and study the Cloud-related policies and techniques. In this work, we provide a comparative survey of the state-of-the-art in Cloud computing simulation. This study will aid researcher to select and employ the appropriate Cloud simulation framework for the experimentation and analysis of the proposed Cloud heuristics. The rest of this paper is organized as follows. In Section 2, a general description of Cloud simulation, an overview of Cloud simulators, a general and a policy based comparison of Cloud simulators is presented. Section 3 includes brief conclusive remarks about the Cloud simulators.

2. Cloud Simulation

Simulation is a technique to create a process model of a real system. Simulation is designed for system evaluation and test heuristics or strategies [21]. In a real Cloud computing environment, all implementation and evaluation operations are time-consuming, not repeatable, and expensive. Moreover, the performance problems and security issues are often difficult to analyze. Therefore, it is difficult to synthesize and analyze diversified aspects of the underlying heuristic in a real Cloud environment [22, 23]. So a more effective alternative is the use of simulation tools [20]. A wide variety of Cloud simulation tools are available for modeling and simulating extensive Cloud computing environments [24]. In this study, we summarize energy aware Cloud computing simulators that are used to evaluate the performance and security aspects of Cloud computing environments. Some of these simulators are CloudSim [20], Green Cloud [35], iCanCloud [29], CloudReports [25], CloudAnalyst [39], and NetworkCloudSim [38]. Table I provides a comparison of Cloud simulators based on common simulator features.

Table 1. **Cloud Simulators: a generalized comparison.**

Compare Factors	CloudSim	Green Cloud	iCanCloud	CloudAnalyst	CloudReports	Network CloudSim
GUI Support	Limited	Limited	Yes	Yes	Yes	-
Availability	Open Source	Open Source	Open Source	Open Source	Open Source	Open Source
Scalability	Yes	-	Yes	-	-	-
Programming Language	Java	C++, otcal	C++	Java	Java	Java
Platform	SimJava	Ns2	INET, OMNET++	CloudSim	CloudSim	CloudSim
Energy-Aware	Yes	Yes	-	Yes	Yes	-
Cloud Security	No	No	No	No	No	No
Task Scheduling	Yes	Yes	-	Yes	Yes	-
Resource management	Yes	Yes	-	Yes	Yes	-
Migration Support	Auto Migr	Auto Migr	-	Auto Migr	Auto Migr	-
Topologies in data centers	Yes	-	-	Yes	No	Yes
Simulation Model	Event Oriented	Packet Level	-	Event Oriented	Event Oriented	-
Accuracy	High	-	-	High	High	-
Services	IaaS,Paas,Saas	IaaS	IaaS	IaaS, SaaS	IaaS, SaaS	IaaS
OS	Linux, Windows, MAC	Linux	Linux	Linux, Windows, MAC	-	Linux
Power Consumption Support	Limited	Full	Wip	Limited	Limited	-
TCP/IP Support	Yes	Yes	Yes	No	-	-
Graphical Output	Partial	Partial	Yes	Yes	-	-
Supports Parallel Experiments	No	No	Wip	No	-	No
Communication Model	Limited	Full	Full	Limited	-	-
Energy Model	Yes	Yes	No	Yes	-	-

Simulators should provide an easy-to-use Graphical User Interface that supports visual modeling [26]. *Availability* refers to a user's ability to access information or resources from a specific location and in the desired and correct format. *Scalability* indicates the enhanced operating speed of the simulator (with respect to the simulation time) and the amount of memory space that the simulator utilizes with the increased workload. Most of the Cloud simulators use *programming languages* such as Java and C ++ for the implementation of simulation programs. Moreover, as most of these simulators are themselves implemented using these programming languages (i.e., C++, Java, etc.) and support extensions to the simulator frameworks [26]. The *platform* features mentioned in Table I represent the implementation base of a simulator. *Energy efficiency* has become a major concern for Cloud computing platforms and data centers and can be considered at different levels such as energy-aware scheduling, resource management, etc. *Cloud security* refers to a broad set of policies used for the protection of data, applications, and incorporated Cloud computing infrastructure. *Task scheduling* determines the shared processing power between tasks and the available resources. In Cloud computing, *resource management* means the efficient usage of various geographically distributed resources [36]. *Migration support* (mentioned in Table I) refers to the migrating possibilities of VMs from one computing resource to another [31]. A simulator may support multiple topologies and also allows the addition of new ones. Tree-based topologies are commonly used in Cloud data-center design [30].

Moreover, Table I also provides several other parameters such as *simulation model* that represents simulation granularity at an event or at packet levels. In an event-based simulation, communication takes place via message passing between different system entities. Packet-level simulation relates to the ad-hoc network supported network simulators such as NS-2 [37]. *Accuracy* in simulation defines the degree to which the results of an experiment are semantically correct. The Cloud *services* parameter defined in Table I is commonly categorized as *Software as a Service* (SaaS), *Platform as a Service* (PaaS), and *Infrastructure as a Service* (IaaS) [9]. Cloud simulators are available for many operating system platforms as mentioned in Table I. The quantification of network-related parameters requires the support of communication and networking protocols (such as TCP/IP) and several simulators provide such support (as shown in Table I) [26]. The *graphical output* provided by the simulators is the only format results in a quick and easy understanding of the produced information as compared to the command line based [27]. The parameter *Parallel experiment support* (mentioned in Table I) means that a particular simulator supports concurrent execution of the simulation on multi-processor machines [29]. *Communication model* is a very important requirement for a Cloud simulator to model and profile communication among data-centers and Cloud applications [28]. A Cloud simulator providing an *energy consumption model* facilitates to compare the energy consumption of different simulated algorithms and scheduling strategies. With the help of *cost model*, researchers can easily evaluate the cost-effectiveness of the new policies [28].

CloudSim [20] is a new, open-source, widespread, and flexible simulation framework [43] written in Java (based on *SimJava* framework [33]). It enables the modeling and simulation of Cloud computing infrastructures and emerging application services [20]. CloudSim is an *event-based* simulator which means that messages are sequenced as events in time. CloudSim provides limited support for the GUI and provides high simulation scalability. Moreover, it provides a limited energy based profiling for the executing simulations [29]. It performs task scheduling by using different scheduling techniques such as dynamic or static scheduling [50]. CloudSim supports auto-migration of VMs. In CloudSim *BRITE* format [40] is used to store the topology description that contains a number of network nodes. BRITE is a universal parameterized topology generation tool which can be used to study various properties of generated topologies such as power laws and average path length. *CloudSim* [20] provides highly accurate simulation results and supports modeling of IaaS, PaaS, and SaaS-based resource management policies. CloudSim is available on several platforms such as Linux, Windows, and MAC operating systems. To model communication in Cloud services, it supports TCP/IP based profiling [42]. CloudSim supports different energy consumption models such as *Dynamic Voltage Frequency Scheduling* (DVFS).

Green Cloud [35] is an energy-aware and packet-level simulator. Green Cloud is based on NS2 [37] simulation framework and focuses on simulating the communication between Cloud services at packet level [41]. In the same way as NS2, GreenCloud is written in C++ and OTcl [28], and this is a disadvantage of this simulator as two different languages must be used to implement one single experiment. GreenCloud is an open-source simulator that supports limited GUI [42]. This simulator can be used to model and simulate Cloud services at IaaS level. Green Cloud provides profiling of energy efficiency which is helpful in simulating green resource allocation and scheduling policies. Green Cloud is available for Linux platform and supports modeling the auto-migration of VMs (transferring a VM across host machines in a Cloud). Green Cloud gives a wide range of power consumption profiling support; however, it does not support parallel simulations. To model communication, Green Cloud provides complete TCP/IP implementation [47] with partial support of GUI. The energy model provided by the green Cloud offers to model energy consumption for several communication devices in data-center such as a router, switches, gateways, and communication links [29]. Green Cloud fully supports modeling of power consumption which is the distinguished attribute of this simulator [47].

iCanCloud [29] is an open-source simulation platform that is designed to model and simulate Cloud computing systems [29]. Simulation of large storage networks is the core function of iCanCloud. iCanCloud provides an easy-to-use GUI which helps to manage, configure, and launch Cloud-based heuristic experiments. iCanCloud is one of the few simulators that provides a comprehensive GUI support [47]. A simulation in iCanCloud can be programmed using C++ [29]. iCanCloud provides scalable simulation at the expense of more consumed memory as compared to CloudSim. iCanCloud has been developed on the top of the INET and OMNeT++ [48] simulators. It is used to model IaaS level Cloud services and is available for Linux platform. Although iCanCloud has been designed to perform parallel simulations so that one experiment can be executed by spreading over several machines but this feature is not completely implemented yet. iCanCloud claims parallel experimentation and power consumption model as *Work in Progress* (WiP*)* [47]. For modeling of communication, iCanCloud supports TCP/IP based profiling of the execution entities.

CloudAnalyst [39] is an event-oriented graphical Cloud simulator and is build on top of the CloudSim simulation framework [20]. CloudAnalyst supports visual modeling of Cloud-based resource allocation, scheduling policies, and large-scale Internet applications [49]. It is an open source simulator that is implemented in Java language. CloudAnalyst can generate graphical results of simulation in the form of tables and graphs, which is useful for efficiently summarizing a large amount of data collected during simulation [39]. CloudAnalyst is compatible with Linux, Windows, and MAC operating systems. It has no TCP/IP implementation available and provides a limited support for power consumption. Additionally, auto VM migration support is available in Cloud Analyst to model dynamic and fault-tolerance related Cloud services. There is a partial support of communication-related performance profiling in CloudAnalyst.

CloudReports [25] uses the CloudSim toolkit as underlying simulation framework [20]. Using the GUI support of CloudReports, researchers can easily model complex Cloud simulations. CloudReports is an open-source simulator [25] that provides the simulation at core Cloud service levels such as IaaS and SaaS. CloudReports supports limited power consumption modeling and is based on event-oriented simulation mechanism. Notable features of CloudReports include high accuracy and adaptive simulations based on auto-VM migration support. It provides Java Reflection API [25] to create extensions that are loaded as plugins.

NetworkCloudSim [38] is a Java-based extension of CloudSim to model network flow within Cloud data-centers. It utilizes bandwidth sharing and latencies to facilitate scalable simulations [38]. It is an open source simulator and is available for Linux platform. At IaaS level several classes such as Switch, Network Packet, and Host Packet are used to model a network topology in NetworkCloudSim [38]. This simulator provides support for TCP/IP based communication profiling.

Next, we compare these six Cloud simulators considering the features related to modeling of Cloud policies and heuristics (as shown in Table II).

Table 2. Cloud simulators: policies based comparison

Simulators	**License**	**Traffic-aware**	**Energy-aware**	**Static Scheduling**	**Dynamic Scheduling**	**VM Migration Policy**		**VM Scheduling Policy**		**VM Allocation Policy**		**VM Selection Policy**			**Power Saving Strategy**	**Transparent Migration**	**SLA Support**	**Power Consumption**	
						Live Migration	**Cold Migration**	**Specification-Based**	**Space Shared, Time Shared**	**LAA**	**Default**	**Heuristic Search**	**MMT, RC**	**MU, MC**				**Static Power Dissipation**	**Dynamic Power Dissipation**
Green Cloud	GPLA	-	Yes	Yes	Yes	Yes	-	Yes	-	Yes	FCFS	Yes	-	-	DVFS	Yes	Yes	Yes	Yes
CloudSim	Apache Version 2.0 License	Yes	Yes	Yes	Yes	Yes	-	-	Yes	Yes	FCFS	-	Yes	Yes	-	Yes	Yes	Yes	Yes
Cloud Analyst	-	Yes	-	-	Yes	-	-	-	Time Shared	-	Round Robin	-	-	-	-	-	Yes	-	-
Cloud Reports	-	-	Yes	-	Yes	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Network CloudSim	-	-	-	-	-	-	-	-	Space Shared	-	FCFS	-	-	-	-	-	-	-	-
iCanCloud	GPL version 3	-	-	-	-	-	-	-	Time Shared	-	max min, min-min	-	-	-	-	-	-	-	-

A Cloud scheduling algorithm can be classified into *static* or *dynamic*. In static scheduling, task priorities and allocations are pre-determined before execution of the simulation while in dynamic scheduling tasks priorities are determined at runtime. *VM migration* is used to migrate a VM between host machines within a data-center. VM migration is useful for implementing dynamic or adaptive policies and to support fault tolerance. Generally, there are two types of migrations: [34], 1) regular or *cold migration* and 2) *live migration*. In live VM migration, a VM switches from one host machine to another while continuing its execution resulting in no obvious effect on end-user performance [35]. The cold or regular migration involves halting execution of a VM before migration to the destination host in a Cloud data-center. The scheduling policies at VM level (allocation of processes to VMs) are comprises of two sub-policies: 1) *space-shared* and 2) *time-shared* [20]. *VM allocation* is the process of creating VMs on hosts. In this allocation, hosts and VMs should match the important characteristics (memory), configurations (software environment), and requirements of the SaaS provider [33]. The *VM selection* problem defines that which VM should be migrated to a new host machine with respect to power and other Cloud policy considerations [33]. The *Service Level Agreement* (SLA) is a contract negotiated between a customer and a service provider which defines all the services that must be provided [32]. Power consumption has two types, static and dynamic power dissipation. The main cause of static power dissipation is current leakages and the dynamic power dissipation is related to the workload processing based consumption during CPU utilization [35].

CloudSim is released under Apache version 2.0 that is an energy and traffic aware simulator [20, 45]. CloudSim supports programming and simulation of both static and dynamic Cloud scheduling policies [44]. CloudSim is capable of managing scheduled processes based on both the space and time-shared VM scheduling heuristics. A space-shared policy assigns specific CPU cores to specific VMs and a time-shared policy dynamically distributes the capacity of a processor among VMs [20]. CloudSim provides VM's live migration. In CloudSim default VM allocation policy is *First Come First Serve* (FCFS) [40] and it provides four VM Migration policies (i) *Minimum Migration Time* (MMT) policy, (ii) *Random Choice* (RC) policy, (iii) *Maximum Correlation* (MC) policy, and (iv) *Minimum Utilization* (MU) policy [33]. DVFS policy can be utilized in CloudSim for the simulation of green scheduling heuristics [46]. CloudSim provides an implementation of SLA based policies and scheduling heuristics.

Green Cloud [35] is released under *General Public License Agreement* (GPLA). Green Cloud is a traffic and energy-aware simulator. It supports both static and dynamic scheduling [44] heuristics with live VM migrations. Green Cloud implements specifications based on VM scheduling policies. It provides *Load-Aware Allocation* (LLA) VM allocation policy [35], by default VM allocation policy in Green Cloud is FCFS. Green Cloud performs a heuristic search for VM selection. Power saving mode in Green Cloud is based on DVFS model. Green Cloud also provides an implementation of SLA agreed on Cloud policies. In Green Cloud, both static and dynamic power consumption policies can be accommodated for the executing simulation.

iCanCloud [29] is released under *General Public License* (GPL). It supports time shared VM scheduling policy. iCanCloud implements two default resource provisioning and scheduling algorithms (i.e., Min-Min and Max-Min) [29].

In *CloudAnalyst* [39], the default VM allocation policy uses a round robin algorithm. By using this algorithm all requests are assigned to the available VMs in a round robin manner, regardless of the current load on each virtual machine [39]. SLAs are met at a minimal cost for consumers in CloudAnalyst. A time-shared policy is used to resource scheduling for virtual machines in CloudAnalyst. Resource provisioning is a configuration for extensive applications in which resources are dynamically assigned by geographic location based on workload [39].

CloudReports [25] is an energy-aware simulator. This simulator provides resource utilization models, VM allocation policies, VM schedulers, which can be further extendable for the customized user needs.

NetworkCloudSim [38] implements time-shared VM scheduling policy. Default VM allocation policy in this simulator is based on FCFS [38].

3. Conclusions

To ensure Green Cloud computing environment many efforts have been conducted to reduce the worldwide energy crisis to some extent. Researchers have used Cloud computing simulators for testing and evaluating these techniques. In this paper, we compare Cloud computing simulators named as Green Cloud, CloudSim, iCanCloud, CloudAnalyst, CloudReports, and NetworkCloudSim on the basis of different parameters. We have found that different simulators have distinct characteristics, features, and developed to meet specific requirements of the researchers. For energy aware simulation *Green Cloud* is the most suitable simulator. *Green Cloud* has many advantages over other Cloud simulators where energy consumption is the key concern. *Green Cloud* is based on the popular NS2 network simulator, which permits it to keenly observe packet-level details. *Green Cloud* accurately calculates the energy consumption because energy consumption is computed on packet levels whereas the *CloudSim* does not provide a comprehensive energy model for packet level simulation. *CloudSim* can be more beneficial when a precise analysis of complex resource allocation policies and scheduling heuristics is required. *iCanCloud* claims parallel experimentation feature in progress and yet not provided by any other simulator. Moreover, *iCanCloud* provides full GUI based interface support. *CloudAnalyst* supports the evaluation of social network tools employing the geographically distributed data-centers and users. *CloudAnalyst* also provides results in full graphical mode based on several types of visualizations (such as Graphs, charts etc.) *CloudReports* is one of the unique simulators to simulate and evaluate the cost of implementation, the degree of resource utilization, and power consumption of the simulated algorithm. Finally, we observe that most of the Cloud computing simulators lack features related to analyzing security issues in Cloud.

References

[1] N. Kaur, T. S. Aulakh, and R. S. Ceema, “Comparison of workflow scheduling algorithms in cloud Computing,” *IJACSA*, vol. 2, no. 10, pp. 81-86, 2011.

[2] Z. Wu, X. Liu, Z. Ni, D. Yuan, and Y. Yang, “A market-oriented hierarchical scheduling strategy in cloud workflow systems,” *J Supercomput*, vol. 63, no. 1, pp. 256- 293, Jan. 2013.

[3] J. Huang, “The Workflow Task Scheduling Algorithm Based on the GA Model in the Cloud Computing Environment,” *JSW*, vol. 9, no. 4, pp. 873-880, April. 2014.

[4] Huth, and J. Cebula, “The basics of cloud Computing,” *US-CERT*, 2011.

[5] D. S. Marcon, L. F. Bittencourt, R. Dantas, M. C. Neves, E. R. Madeira, S. Fernandes, and N. L. da Fonseca. (2013, December).Workflow specification and scheduling with security constraints in hybrid clouds (LatinCloud),2013 2nd IEEE Latin American Conference.

[6] M. A. AlZain, E. Pardede, B. Soh, and J. A. Thom. (2012, January). Cloud computing security: from single to multi-clouds (HICSS), 2012 45th IEEE Hawaii International Conference.

[7] M. Jensen, J. Schwenk J. M. Bohli, N. Gruschka, and L. L. Iacono. (2011, July). Security prospects through cloud computing by adopting multiple clouds (CLOUD), IEEE International Conference Cloud Computing.

[8] R. Buyya, R. Ranjan, and R. N. Calheiros. (2010, May). Intercloud: Utility-oriented federation of cloud computing environments for scaling of application services, 2010 International Conference.

[9] J. Wang, P. Korambath, I. Altintas, J. Davis, and D. Crawl, “Workflow as a service in the cloud: architecture and scheduling algorithms,” *Procedia Comput Sci*, vol. *29*, pp. 546-556, 2014.

[10] L. Wu, S. K. Garg, and R. Buyya. (2011, May). Sla-based resource allocation for software as a service provider (saas) in cloud computing environments *(CCGrid)*, 2011 11th IEEE ACM International Symposium Cluster, Cloud and Grid Computing.

[11] S. Basishtha, and S. Boruah, “Cloud Computing and Its Security Aspects”.

[12] T. Dillon, C. Wu, and E. Chang. (2010, April). Cloud computing: issues and challenges (AINA), 2010 24th IEEE International Conference.

[13] D. Agarwal, and S. Jain, “Efficient optimal algorithm of task scheduling in cloud computing environment,” *IJCTT*, 2014.

[14] Weiss, “Computing in the clouds,” *Networker*, vol. 11, no. 4, pp. 16-25, 2007.

[15] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, “Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility,” *FGCS*, vol. *25, no.* 6, pp. 599-616, 2009.

[16] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, and M. Zaharia, (2010). “A view of cloud computing,” Commun. ACM, vol. *53, no.* 4, pp. 50-58, 2010.

[17] S. Subashini, and V. Kavitha, “A survey on security issues in service delivery models of cloud computing,” *JNCA*, vol. *34, no.* 1, pp. 1-11, 2011.

[18] D. Chappell, “Introducing the Azure services platform,” *White paper, vol. 34, no. 1, pp. 1-11, Oct. 2011.*

[19] Hoefer, C. N. & Karagiannis, G. (2010, December). Taxonomy of cloud computing services (GC Wkshps), 2010 IEEE GLOBECOM Workshops.

[20] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. De Rose, and R. Buyya, “CloudSim: a toolkit for modeling and simulation of cloud

computing environments and evaluation of resource provisioning algorithms," SPE, vol. *41, no.* 1, pp. 23-50, 2011.

[21] M. A. Yahya, "A Middleware Approach for Secure Data Outsourcing in the Cloud," M.S. thesis, *Dept* Comp. Info Sci, Prince Sultan Univ., Riyadh, Saudi Arabia, 2016.

[22] M. Malhotra, "Simulation for enhancing the response and processing time of datacenter," *IJCCR*, vol. *1, no.* 3, 2011.

[23] F. Fittkau, S. Frey, and W. Hasselbring. (2012, September). CDOSim: Simulating cloud deployment options for software migration support (MESOCA), 2012 IEEE 6th International Workshop.

[24] W. Zhao, Y. Peng, F. Xie, and Z. Dai. (2012, November). Modeling and simulation of cloud computing: A review (APCloudCC), 2012 IEEE Cloud Computing Congress.

[25] T. T. Sá, R. N. Calheiros, and D. G. Gomes, "CloudReports: An extensible simulation tool for energy-aware cloud computing environments," in Cloud Computing: Challenges, limitations, and R & D Solutions, Ist ed. Switzerland: Springer, 2014, pp. 127-142 [online], Available: https://link.springer.com/chapter/10.1007/978-3-319-10530-7_6

[26] C. Kim, J. Kim, and W. J. Lee, "Design of simulator for cloud computing infrastructure and service," *IJSH*, vol. *8, no.* 6, pp. 27-36, 2014.

[27] K. Goga, O. Terzo, P. Ruiu, and F. Xhafa. (2014, July). Simulation, Modeling, and Performance Evaluation Tools for Cloud Applications (CISIS), 2014 8th International Conference.

[28] A. Ahmed, and A. S. Sabyasachi. (2014, February). Cloud computing simulators: A detailed survey and future direction (IACC), 2014 IEEE International Computing Conference.

[29] Núñez, J. L. Vázquez Poletti, A. C. Caminero, G. G. Castañé, J. Carretero, and I. M. Llorente, "iCanCloud: A flexible and scalable cloud infrastructure simulator," *JGC,* vol. 10, no. 1, pp. 185-209, 2012.

[30] M. A. Sharkh, A. Kanso, A. Shami, P. Öhlén, "Building a cloud on earth: a study of cloud computing data center simulators," *Comput. Netw*, vol. *108*, pp. 78-96, 2016.

[31] E. Shamsinezhad, A. Shahbahrami, A. Hedayati, A. K. Zadeh, and H. Banirostam, "Presentation Methods for Task Migration in Cloud Computing by Combination of Yu Router and Post-Copy," *IJCSI*, vol. *10, no.* 4, pp. 98-102, 2013.

[32] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *FGCS, vol. 25,no.* 6, pp. 599-616, 2009.

[33] W. Long, L. Yuqing, and X. Qingxin. (2013, December). Using cloudsim to model and simulate cloud computing environment (CIS), 2013 9th International Conference.

[34] J. Shetty, M. R. Anala, G. Shobha, "A survey on techniques of secure live migration of virtual machine," *IJCA*, vol. *39, no.* 12, pp. 34-39, 2012.

[35] L. Liu, H. Wang, X. Liu, X. Jin, W. B. He, Q. B. Wang, and Y. Chen. (2009, June). GreenCloud: a new architecture for green data center (ACM), 2009 6th international conference.

[36] K. Chandrasekara. (2013, December). Tutorial: Resource Management in Cloud Computing (ACM), 2013 IEEE 6th International Conference.

[37] B. Prakash, S. Manjunatha, H. M. Balakrishna, "A Detailed Survey on Various Cloud Computing Simulators," *IJER*, pp. 2319-6890, 2016.

[38] S. K. Garg, and R. Buyya. (2011, December). Networkcloudsim: Modelling parallel applications in cloud simulations (UCC), 2011 4th IEEE International Conference.

[39] B. Wickremasinghe, R. N. Calheiros, and R. Buyya. (2010, April). Cloudanalyst: A cloudsim-based visual modeller for analysing cloud computing environments and applications *(*AINA), 2010 24th IEEE International Conference.

[40] A. Medina, A. Lakhina, I. Matta, and J. Byers. (2001). BRITE: An approach to universal topology generation. In Modeling, Analysis and Simulation of Computer and Telecommunication Systems 2001 9th IEEE International Symposium.

[41] J. C. Huet, and I. E. Abbassi. (2013, December). Green cloud computing modeling methodology (UCC), 2013 IEEE/ACM 6th International Conference.

[42] M. I. Tariq, and V. Santarcangelo, "Simulator's Requirements For Modeling and Simulation Of Cloud Environment" 2015.

[43] X. Li, X. Jiang, K. Ye, and P. Huang. (2013, June). DartCSim+: Enhanced cloudsim with the power and network models integrated (CLOUD), 2013 IEEE 6th International Conference.

[44] R. Benali, H. Teyeb, A. Balma, S. Tata, and N. B. Hadj-Alouane. (2016, June). Evaluation of traffic-aware VM placement policies in distributed Cloud using CloudSim (WETICE), 2016 IEEE 25th International Conference.

[45] B. Louis, K. Mitra, S. Saguna, and C. Åhlund. (2015, December). Cloudsimdisk: Energy-aware storage simulation in cloudsim (UCC), 2015 8th IEEE/ACM International Conference.

[46] T. Guérout, T. Monteil, G. Da Costa, R. N. Calheiros, R. Buyya, M. Alexandru, "Energy-aware simulation with DVFS," *SMPT,* vol. *39*, pp. 76-91, 2013.

[47] B. Hasselmann, "Investigation of Multi-Resource Cloud Simulators," 2015.

[48] OMNeT, http://omnetpp.org/,[Accessed online Dec 2017].

[49] R. Kumar, "Cloud Computing: An Introspection" *IJARCCE*, vol. 2, no. 11, Nov. 2013.

[50] Hussain, A., Aleem, M., Khan, A., Iqbal, M.A., Islam, M.A., "RALBA: a computation-aware load balancing scheduler for cloud computing", Cluster Computing, pp. 1-14, Mar. 2018