A Performance Analysis of a Typical Server running on a Cloud

Tangila I. Tanni
The Department of Computer Science and Engineering
University of Dhaka, Dhaka, Bangladesh

Mohammad S. Hasan
School of Computing and Digital Technologies
Staffordshire University, Stoke-on-Trent, UK

Abstract—**In recent years, Cloud computing has become very popular from individuals to big enterprises because of the reduced cost, minimal management effort etc. Cloud providers like Amazon, Google are now offering resources for web deployment, storage, servers etc. However, due to the varying load and cost, evaluating the performance of task scheduling policies in these real Cloud environments is very challenging. In this paper, the performance of Earliest Deadline First (EDF) scheduling policy has been investigated using CloudSim and the hardware configuration of Amazon Web Services (AWS) and Google Cloud Platform (GCP) with the time-out of a web and FTP servers. In addition, a comparison between space-shared and time-shared task provisioning policies have been examined which shows that the average execution time can be minimised by using space-shared policy in both AWS and GCP.**

Keywords—CloudSim, EDF algorithm, Cloudlets, Virtual Machine, Server Load, Execution Time etc.

I. Introduction

Cloud computing has gained huge popularity in recent years. It is a shared pool of virtual computing resources e.g. networks, servers, applications, storage, services etc. which are available for the users. By using the Internet, businesses can host various services e.g. web, file transfer etc., store data etc. which reduces the need for software and hardware [1]. From start-ups to large companies, a majority of the businesses are now moving into the Cloud because of the minimal management effort and reduced cost [2]. Therefore, the number of companies adopting Cloud Computing is growing at a higher rate every year. Big enterprises like Vodafone, Expedia, Frame, GeekWire, CocaCola, Motorola etc. are now adopting Cloud technology [3], [4], [5]. Cloud providers like Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP), IBM etc. are now ruling the world market. Figure 1 shows that in 2016, the public Cloud Computing market was worth around 114 billion U.S. dollar worldwide and it is predicted to be about 130 billion U.S. dollar in 2017 [6].

As Cloud Computing offers a very dynamic computing environment, evaluating the performance of various task scheduling policies in a real Cloud e.g. AWS, GCP etc. for different services can be difficult as well as expensive in terms of varying load, different applications pattern and available resources. In addition, it is time-consuming to reconfigure the parameters to perform multiple test runs. Therefore, a simulation tool e.g. CloudSim [7] offers a practical way to estimate or predict the performance of a Cloud platform. This paper represents realistic models for Cloud environments and task and evaluates the performance of Earliest Deadline First (EDF) task scheduling policy using CloudSim. The key contributions of this paper are:

- A mathematical relationship between a task deadline for services such as web and File Transfer Protocol (FTP) request tasks.
- A comparison between two task provisioning policies namely Space-shared and Time-shared policy when different loads are applied and the deadline for each task is known.
- Realistic simulation models for AWS and GCP.

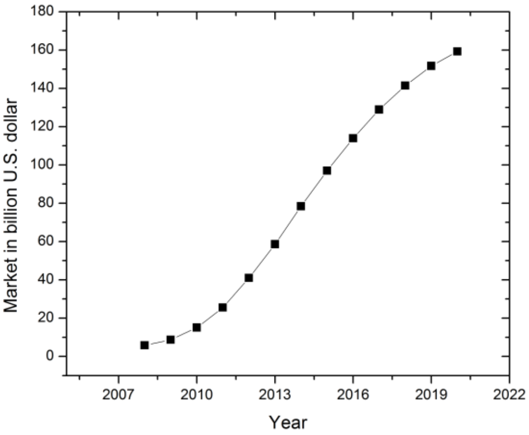

Figure 1: Public Cloud Computing market worldwide [6].

The rest of the paper is organised as follows. Section II presents a review of previous works that have been done using CloudSim. Section III discusses CloudSim, terminologies associated with this simulator, models for AWS and GCP, tasks etc. Section IV shows the performance of the task scheduling policy corresponding to different loads Cloud. Finally, section V presents conclusion and points to future research directions.

II. Previous Work

Paper [8] proposes an algorithm to schedule different tasks by combining two characteristics such as length and deadline. Their analysis shows that the average waiting time and task completion time can be reduced as compared to other existing algorithms which either considers length or deadline but not both. But in their study, they have used random lengths and deadlines for different tasks.

Paper [9] presents a Round Robin (RR) technique to distribute the work-load between a broker and Virtual Machine (VM). It also compares RR and First Come First Served (FCFS) policy and shows the cost and completion time associated with each task.

Paper [10] proposes a scheduling algorithm that classifies the tasks based on three criteria i.e. execution time, cost and priority. Then, the task with minimum turnaround time or cost is scheduled first. However, the algorithm does not consider task types or realistic task length.

Paper [11] presents Ant Colony and compares the result with FCFS and RR. The results show that Ant Colony optimisation can minimise the makespan of a cluster of tasks.

Paper [12] proposes VM allocation policy to map tasks to VMs. The study focuses on the load distribution by binding the task to VM such that all the tasks can get the high performance or less execution time.

Paper [13] introduces an efficient way to assign VMs to hosts based on the task priorities.

Paper [14] shows a comparison between space-shared and time-shared task provisioning policies to minimise the cost and execution time to obtain a maximum resource utilisation.

Paper [15] presents a comparative analysis of task and VM scheduling algorithm based on criteria such as waiting time, throughput, response time etc.

Paper [16] proposes a new priority-based task scheduling algorithm based on Analytical Hierarchy Model which is a decision making mathematical model. The proposed algorithm shows minimum makespan and maximum throughput.

Paper [17] introduces a cost-effective priority based task scheduling algorithm to minimise the resource utilisation and to reduce cost.

A considerable amount of work has been done on resource allocation, task scheduling and load balancing to reduce task completion time. Most of the research in this area does not consider a realistic Cloud Computing model and does not provide practical task properties e.g. length, deadline etc.

III. Simulation Model

A. CloudSim

CloudSim is a software for modelling and simulation of Cloud Computing infrastructure and services. It provides the following features.

- It allows modelling of Cloud Data Centres (DC) including different hardware configuration such as memory, storage, bandwidth etc.
- It supports the creation of Cloud entities e.g. host, broker, Cloud Information Service (CIS) and VM and provides a communication mechanism between these entities as shown in Figure 2.
- It provides flexibility to switch between space-shared and time-shared allocation of CPU cores to different virtualised services.
- It allows execution of different provisioning policies of a host to VMs [7], [18].

CloudSim is designed as a multi-layer framework where each layer provides several core functionalities such as the creation of Cloud system entities (CIS, host, DCs, broker, task unit), submission of a task to the appropriate entities, handling each event with appropriate response and management of simulation clock [7]. Some major components are explained below.

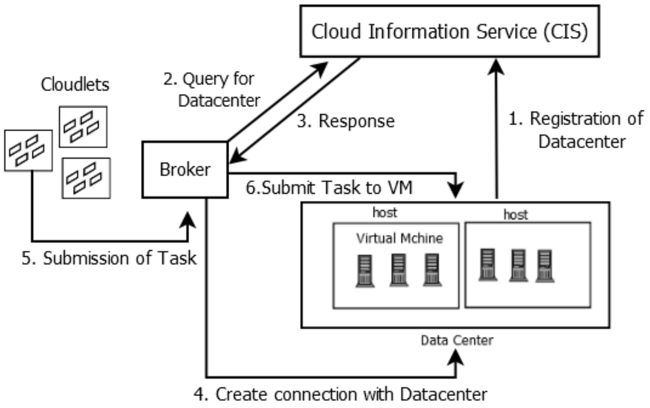

Figure 2: Communication flow between various entities in CloudSim [12].

1) Datacentre

It can represent Cloud service providers such as AWS, Microsoft Azure, Google, IBM etc. which offer services for database, storage, analytics, applications and deployment of various services [3]. Each datacentre has some characteristics e.g. the total number of homogenous or heterogeneous hosts based on CPU power, memory, storage capacity, bandwidth etc. [7].

2) Host

Each host represents a physical server with associated resources e.g. memory, bandwidth, storage capacity, number of processing cores, and allocation policies for sharing processing cores among VMs and provisioning policies for memory and bandwidth [7]. Furthermore, the processing capability of each core also needs to be assigned in terms of Millions of Instruction Per Second (MIPS) [7].

3) Cloudlet

It represents a task e.g. a web or FTP, social networking service that is serving a user etc. Each Cloudlet has some characteristics such as length which is defined as Millions of Instruction (MI), input file size, output file size, number of cores needed to finish the task [7].

4) VM

It represents a logical core which is managed and hosted by a host. It also has some characteristics e.g. memory, storage, bandwidth requirements and provisioning policies to execute tasks inside the VM.

5) Provisioning Policy

CloudSim provides VM provisioning at two levels i.e. host and VM. At host level, the overall processing power of each core can be specified which will be assigned to the VM. At VM level, a VM assigns a fixed amount processing power to an individual task that is hosted. CloudSim implements the time-shared and space-shared provisioning policies at both levels [7]. Figure 3 shows, the effect of time-shared and space-shared policies at the VM level.

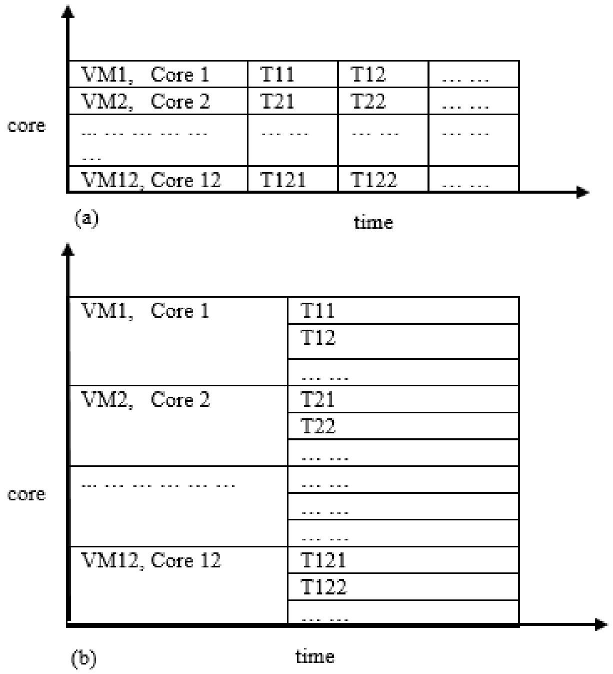

Figure 3: The Space-shared and Time-shared provisioning at the VM level.

a) Space Shared Policy

Figure 3(a), represents a scenario where the space-shared policy is applied at the VM and host levels and to twelve VMs with associated Cloudlets. In this figure, a host with twelve cores receives a request for hosting twelve VMs where each VM needs one core. As each VM requires one core, in SpaceShared mode, all of the VMs can run simultaneously. In this policy, the estimated finish time (T) of a task (t) executed by a VM is given by (1) [7]. Here, Task_length is the number of instructions for a Cloudlet in MI. And the total Capacity of a host having np processing elements (PE) can be defined by (2) [7] where cap(i) is the processing power of individual elements.

$$\text{T(t) = Estimated\_Start\_Time} + \frac{Task\_Length}{Capacity*cores} \qquad (1)$$

$$\text{Capacity} = \sum_{i=1}^{np} \frac{cap(i)}{np} \qquad (2)$$

b) Time Shared Policy

In Figure 3(b), a time-shared policy is applied at the host level and space-shared policy is applied to VM. This policy considers that every submitted task will be added to the execution queue, though the total number of Cloudlets is greater than the total number of processing cores. This model executes all the submitted tasks simultaneously and for this reason, the waiting list is always empty. The estimated finish time of a Cloudlet for this policy is measured by (3) [7]. And the capacity in this scenario can be calculated by (4) where cap(i) is the processing power of individual elements.

$$\text{T(t) = Current Simulation Time} + \frac{Task\_Length}{Capacity*cores} \qquad (3)$$

$$\text{Capacity} = \frac{\sum_{i=1}^{np} cap(i)}{\max(\sum_{j=1}^{cloudlets} cores(j), np)} \qquad (4)$$

B. Cloud model

Cloud service provider is a third-party company that offers Cloud platform such as Infrastructure as a Service (IaaS), Software as a Service (SaaS), Platform as a Service (PaaS) [19], [20]. To model and simulate the performance of real Cloud, the hardware configurations of AWS free tier, GCP are used in this paper.

1) AWS

Amazon Elastic Compute Cloud (EC2) offers an Intel Xeon E5-2676 v3 with 2.40 GHz which provides 12 cores in total [21]. Although the free tier offers a maximum of 1 core, this paper considers the maximum power of EC2 i.e. 12 cores. The average Clock per Instruction (CPI) is 2 for a Xeon processor [22], [23]. So, the MIPS for Xeon E5-2676 can be calculated by (5).

MIPS = (2.4 ∗ 〚10〛^9)/(2 ∗ 〚10〛^6) = 1200 (5)

2) GCP

Google free tier offers various types of processors for different regions. For closer geographical location, Intel Broadwell Xeon E5 v4 with 2.2 GHz is used which is situated in Belgium [24]. Customers can use up to 8 cores. The MIPS for this processor is calculated by (6).

MIPS = (2.2* 〚10〛 ^9)/(2* 〚10〛 ^6) = 1100 (6)

C. Task Model

1) Web server

To evaluate the performance of a web server, three crucial characteristics are defined in this paper namely length, deadline and Request Per Second (RPS).

a) Task Length

A typical execution time of a web server task is between 200 and 500 ms [25], [26]. Therefore, the task lengths are generated randomly between the minimum and maximum lengths which is calculated using (7). For example, for AWS EC2, the task length is between 240 and 600 MI.

$$\text{Task\_Length} = \text{MIPS\_of\_VM} * \text{execution\_time} \quad (7)$$

b) Deadline

In this paper, each task is given a certain amount of extra time to finish which depends on the task length. The formula for defining the deadline is given by (8) where N is an integer multiplication factor. For the Apache server, the default timeout is 300 seconds [27]. Therefore, for the longest task i.e. 500 ms, the maximum value of N can be 600 to satisfy the timeout i.e. 300s.

$$\text{Deadline} = \text{Arrival\_Time} + \frac{Task\_length}{MIPS\_of\_VM} * \text{N} \quad (8)$$

c) Request per Second (RPS)

A typical load for a server varies from application to application. For example, an average load for Wikipedia is 200 Requests Per Second (RPS) [28]. For this reason, 200 RPS is chosen as a minimum load for a web server. And normal load for a server can vary from 1000 to 2000 RPS [29]. So the maximum load is chosen 2000 RPS in this investigation.

2) FTP server

FTP is a standard protocol which is used to exchange files over the Internet [30]. It uses client-server architecture. There are some popular FTP software e.g. FileZilla which is a free FTP solution for both server and client to upload and download files e.g. E-book, MP3 songs, movies etc. [31], [32]. To evaluate the performance of FileZilla, MP3 file type is considered in this paper. Bitrate varies from 96 to 320 kilobits per second (Kbps) for MP3 files. The bitrates between 128 and 320 Kbps (maximum bit rate) MP3 file is considered in this investigation [33].

a) Task Length

File size is measured by using the bit rates of the MP3 file and the time frame for that file. The duration for an MP3 file varies from 3 to 4 minutes [34]. In this paper, 4 minutes (240 seconds) is considered for the duration. Therefore, the smallest and largest MP3 file sizes can be calculated as 3.84 MB and 9.6 MB, respectively using (9).

$$\text{File\_size} = \text{bit\_rate} * \text{duration} \quad (9)$$

The execution time of an FTP task is calculated by using (10). The bandwidth of the network is considered as 100 Mbps [35]. Therefore, the smallest and largest FTP task execution times are 307 and 768 ms, respectively. Now using (7), the task length for the MP3 file transfer is generated between 360 and 900 MI.

$$\text{Execution\_Time} = \frac{File\_size}{Bandwidth\_of\_the\_network} \quad (10)$$

b) Deadline

The default timeout for FileZilla is 200 seconds [36]. So the maximum values of N for AWS and Google are 266 and 244, respectively which is calculated using (8).

c) Request per second (RPS)

Same amount of load e.g. 200, 500, 1000, 1500, and 2000 is applied to FTP server as well.

IV. Results and Analysis

In this simulation, it is assumed that all the tasks have arrived at the broker at the same time. So, the arrival time for all tasks is assigned to 0 second. In addition, Round Robin (RR) scheduling policy is applied to select the VMs. As context switch can take few nanosecond to microsecond depending on the CPU architecture, in this simulation the time for a context switch is given 0.001 seconds [37], [38].

A. Web load 200, 500, 1000, 1500, 2000 RPS

1) Load vs Deadline

Figure 4 shows the lowest value of N to satisfy all the task deadlines for different server loads for both Cloud platforms i.e. AWS and GCP. The simulations show that with the increasing server load, deadline needs to increase accordingly to satisfy the deadline for all tasks. It can be noted that when the same amount of load is applied to both SpaceShared and TimeShared provisioning policies, the value of N in TimeShared policy has to be larger than the SpaceShared policy. In SpaceShared policy, each task gets the full processing power of a VM until the task is completed whereas, in TimeShared policy, each VM executes a task for a constant amount of time and then context switches to run other tasks. So, there is a significant amount of delay in frequent context switching in TimeShared policy as the server load increases.

Figure 4 also shows that in both SpaceShared and TimeShared provisioning policies and under the same server load, AWS performs better than GCP. GCP shows poor performance because in this paper it is assumed that AWS offers 12 processing cores whereas GCP offers 8 cores as discussed in section III.B.

2) Turnaround Time

Turnaround time can be defined as the total amount of time required between the submission of a task for execution and the completion of the task [39]. As the arrival time for all tasks is zero in this paper, the turnaround time for each task is the same as the finish time. Figure 5 shows that AWS performs better than GCP for both SpaceShared and TimeShared policies. It can be noted that SpaceShared policy performs because in SpaceShared policy the number of context switches is the minimum.

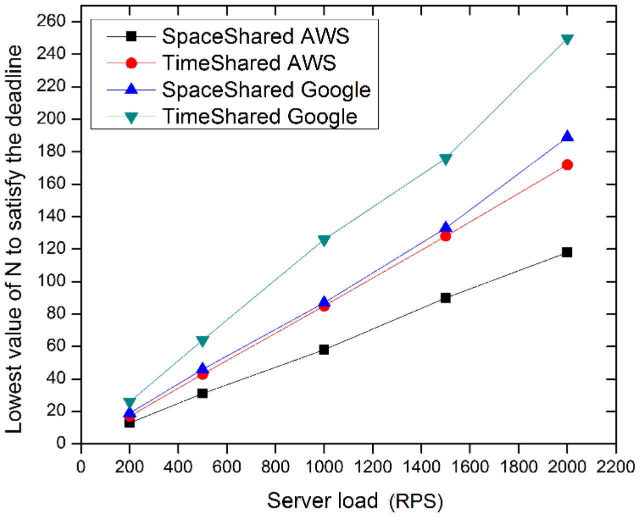

Figure 4: Load vs. Deadline for SpaceShared and TimeShared provisioning policies for the Web server.

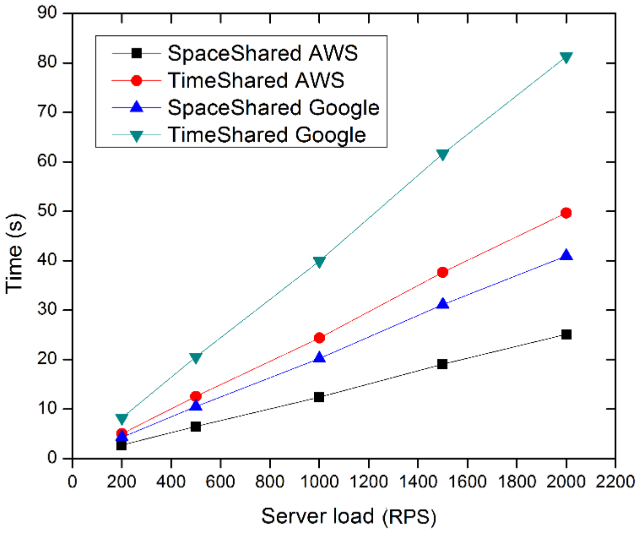

Figure 5: Average Turnaround time for varying load for the Web server.

B. FTP load 200, 500, 1000, 1500, 2000 RPS

1) Load vs. Deadline

Figure 6 shows that for highest server load i.e. 2000 RPS, the maximum value of N exceeds the boundary value i.e. 200 s in TimeShared policy for GCP. Thus it cannot meet the deadline in TimeShared policy.

2) Turnaround Time

Figure 7 compares two provisioning policies for AWS and GCP. It shows that SpaceShared policy performs better for both Cloud platforms and points out that AWS shows better performance than GCP.

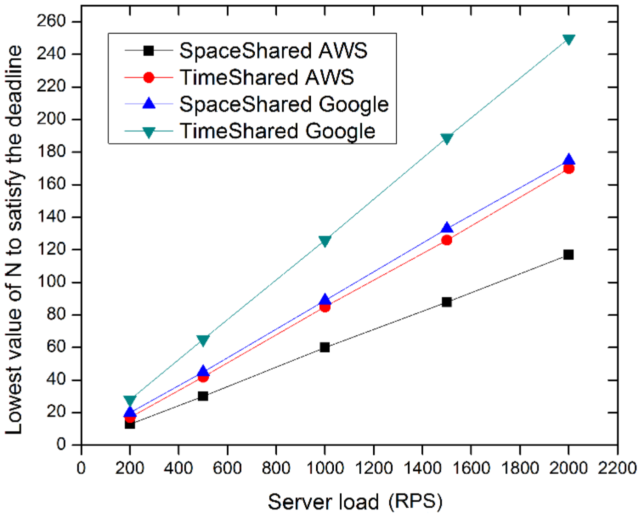

Figure 6: Load vs. Deadline for SpaceShared and TimeShared provisioning policies for the FTP server.

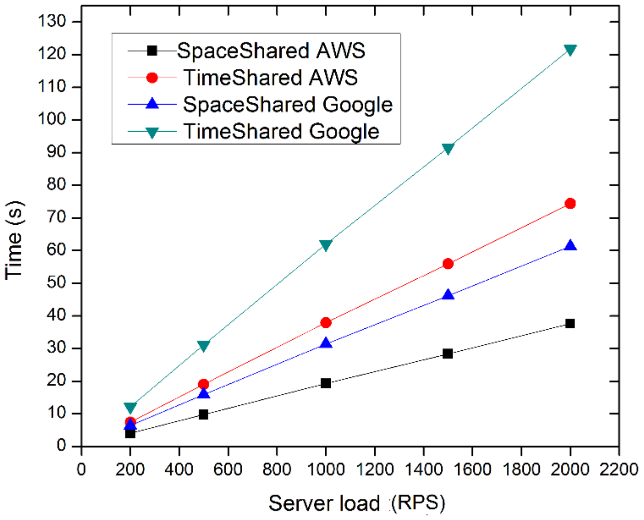

Figure 7: Average Turnaround time for varying load for the FTP server.

V. CONCLUSION

In this paper, the effect of different server load is investigated for two different real Cloud platforms. The result shows that AWS performs better than GCP for both web and FTP server. In addition, SpaceShared policy shows desirable performance than TimeShared policy. Moreover, for FTP server load 2000 RPS, the TimeShared policy fails to meet the deadline for the tasks. Finally, as future research can focus on random task arrival time and dynamic mapping of the tasks to VMs at run-time.

ACKNOWLEDGMENT

The authors would like to acknowledge an Erasmus+ International Credit Mobility (ICM) fund for Bangladesh awarded to Dr Mohammad Hasan at Staffordshire University, UK in 2016.

REFERENCES

[1] "Why Businesses are moving to the Cloud. Streamlining, cutting costs and going green." [Online]. Available: http://www.cns-it.co.uk/businesses-moving-cloud/.

[2] "Moving to the Cloud: What to Know & Who to Hire." [Online]. Available: https://www.upwork.com/hiring/development/moving-to-cloud-servers/.

[3] "Amazon Web Services (AWS) - Cloud Computing Services." [Online]. Available: https://aws.amazon.com/. [Accessed: 18-Jul-2017].

[4] "Customer and Partner Success Stories for Microsoft Azure." [Online]. Available: https://azure.microsoft.com/en-gb/case-studies/. [Accessed: 18-Jul-2017].

[5] "Companies Using Google Cloud Services | Google Cloud Platform." [Online]. Available: https://cloud.google.com/customers/. [Accessed: 18-Jul-2017].

[6] "Public cloud computing market worldwide 2008-2020 | Statistic." [Online]. Available: https://www.statista.com/statistics/510350/worldwide-public-cloud-computing/. [Accessed: 18-Jul-2017].

[7] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Pr. Exper*, vol. 41, pp. 23–50, 2011.

[8] D. T. Arnav Wadhonkar, "A Task Scheduling Algorithm Based on Task Length and Deadline in Cloud Computing," *IJSER*, vol. 7, no. 4, pp. 1905–1909, 2016.

[9] S. Santra and K. Mali, "A new approach to survey on load balancing in VM in cloud computing: Using CloudSim," *IEEE Int. Conf. Comput. Commun. Control. IC4 2015*, pp. 4–8, 2016.

[10] M. Choudhary, "A Dynamic Optimization Algorithm for Task Scheduling in Cloud Environment Sateesh Kumar Peddoju," vol. 2, no. 3, pp. 2564–2568, 2012.

[11] M. Tawfeek, A. El-Sisi, A. Keshk, and F. Torkey, "Cloud Task Scheduling Based on Ant Colony Optimization," *Int. Arab J. Inf. Technol.*, vol. 12, no. 2, pp. 64–69, 2015.

[12] K. Parikh, N. Hawanna, H. P. K., and J. R., "Virtual Machine Allocation Policy in Cloud Computing Using CloudSim in Java," *Int. J. Grid Distrib. Comput.*, vol. 8, no. 1, pp. 145–158, Feb. 2015.

[13] S. Kumar Srivastava and K. Rangasamy, "Priority Based Resource Scheduling Algorithhm In CloudSim," *Int. J. Sci. Res.*, vol. 3, no. 4, 2014.

[14] H. S. Sidhu, "Comparative Analysis of Scheduling Algorithms of Cloudsim in Cloud Computing," *Int. J. Comput. Appl.*, vol. 97, no. 16, pp. 975–8887, 2014.

[15] P. M. Himthani E Scholar, A. Saxena Asso, and M. Manoria, "Comparative Analysis of VM Scheduling Algorithms in Cloud Environment," *Int. J. Comput. Appl.*, vol. 120, no. 6, pp. 975–8887, 2015.

[16] Kamalakar M and Moulikat, "A Priority Based Job Scheduling Algorithm in Cloud Computing," 2015.

[17] I. Dharmendra and R. Srinath, "Cost-Effective User Priority Based Task Scheduling in Cloud," *Int. J. Adv. Res. Comput. Sci. Technol.*, vol. 55, no. 2, 2016.

[18] S. K. Garg and R. Buyya, "An environment for modeling and simulation of message-passing parallel applications for cloud computing," *Softw. Pract. Exp.*, vol. 43, no. 11, pp. 1359–1375, Nov. 2013.

[19] "What is a Cloud Service Provider - Definition | Microsoft Azure." [Online]. Available: https://azure.microsoft.com/en-us/overview/what-is-a-cloud-provider/. [Accessed: 24-Jul-2017].

[20] "What is cloud provider? - Definition from WhatIs.com." [Online]. Available: http://searchcloudprovider.techtarget.com/definition/cloud-provider. [Accessed: 24-Jul-2017].

[21] "Amazon EC2 Instance Types – Amazon Web Services (AWS)." [Online]. Available: https://aws.amazon.com/ec2/instance-types/. [Accessed: 27-Jul-2017].

[22] "CPI Rate (Intel® Xeon Phi^{TM} Coprocessor) | Intel® Software." [Online]. Available: https://software.intel.com/en-us/node/712282. [Accessed: 27-Jul-2017].

[23] "Clockticks per Instructions Retired (CPI) | Intel® Software." [Online]. Available: https://software.intel.com/en-us/node/544403. [Accessed: 27-Jul-2017].

[24] "Regions and Zones | Compute Engine Documentation | Google Cloud Platform." [Online]. Available: https://cloud.google.com/compute/docs/regions-zones/regions-zones. [Accessed: 27-Jul-2017].

[25] "Improve Server Response Time | PageSpeed Insights | Google Developers." [Online]. Available: https://developers.google.com/speed/docs/insights/Server. [Accessed: 15-Aug-2017].

[26] P. A. Laplante and S. J. Ovaska, *Real-time systems design and analysis : tools for the practitioner*. Wiley-IEEE Press, 2012.

[27] "core - Apache HTTP Server." [Online]. Available: http://httpd.apache.org/docs/2.0/mod/core.html#timeout. [Accessed: 25-Jul-2017].

[28] "optimization - What's the &quot;average&quot; requests per second for a production web application? - Stack Overflow." [Online]. Available: https://stackoverflow.com/questions/373098/whats-the-average-requests-per-second-for-a-production-web-application. [Accessed: 25-Jul-2017].

[29] "Requests per second. A server load reference – Wrong Side of Memphis." [Online]. Available: https://wrongsideofmemphis.wordpress.com/2013/10/21/requests-per-second-a-reference/. [Accessed: 25-Jul-2017].

[30] "What is File Transfer Protocol (FTP)? - Definition from WhatIs.com." [Online]. Available: http://searchenterprisewan.techtarget.com/definition/File-Transfer-Protocol. [Accessed: 02-Aug-2017].

[31] "FileZilla - The free FTP solution." [Online]. Available: https://filezilla-project.org/. [Accessed: 02-Aug-2017].

[32] "How to transfer large video files using acceleration | FileCatalyst." [Online]. Available: http://filecatalyst.com/todays-media-file-sizes-whats-average/. [Accessed: 02-Aug-2017].

[33] "MP3 Bit Rates - MP3 Bit Rates | HowStuffWorks." [Online]. Available: http://computer.howstuffworks.com/mp32.htm. [Accessed: 02-Aug-2017].

[34] "Solved: Size of average song in MB - The Spotify Community." [Online]. Available: https://community.spotify.com/t5/Android/Size-of-average-song-in-MB/td-p/1200704. [Accessed: 02-Aug-2017].

[35] "The Average Bandwidth on WiFi | Chron.com." [Online]. Available: http://smallbusiness.chron.com/average-bandwidth-wifi-69070.html. [Accessed: 02-Aug-2017].

[36] "Filezilla FTP - Connection Timeout Error - Interserver Tips." [Online]. Available: https://www.interserver.net/tips/kb/filezilla-ftp-connection-timeout-error/. [Accessed: 02-Aug-2017].

[37] "(1) How long does a context switch take? - Quora." [Online]. Available: https://www.quora.com/How-long-does-a-context-switch-take. [Accessed: 31-Jul-2017].

[38] "Tsuna's blog: How long does it take to make a context switch?" [Online]. Available: http://blog.tsunanet.net/2010/11/how-long-does-it-take-to-make-context.html. [Accessed: 31-Jul-2017].

[39] "What is Turnaround Time (TAT)? - Definition from Techopedia." [Online]. Available: https://www.techopedia.com/definition/23798/turnaround-time-tat. [Accessed: 31-Jul-2017].