A Review Energy-Efficient Task Scheduling Algorithms in Cloud Computing

Saleh Atiewi, Salman Yussof, Mohd Ezanee
Tenaga National University
Jalan IKRAM-UNITEN, 43000 Kajang, Selangor,
Malaysia
s.atiewi@gmail.com; {Salman; Ezanee }@ uniten.edu.my
Muder Almiani
Al-Hussein Bin Talal University
Ma'an, Jordan
malmiani@my.bridgeport.edu

Abstract—**Cloud computing is a model for delivering information technology services, wherein resources are retrieved from the Internet through web-based tools and applications instead of a direct connection to a server. The capability to provision and release cloud computing resources with minimal management effort or service provider interaction led to the rapid increase of the use of cloud computing. Therefore, balancing cloud computing resources to provide better performance and services to end users is important. Load balancing in cloud computing means balancing three important stages through which a request is processed. The three stages are data center selection, virtual machine scheduling, and task scheduling at a selected data center. User task scheduling plays a significant role in improving the performance of cloud services. This paper presents a review of various energy-efficient task scheduling methods in a cloud environment. A brief analysis of various scheduling parameters considered in these methods is also presented. The results show that the best power-saving percentage level can be achieved by using both DVFS and DNS.**

Keywords—Cloud computing; Energy-Efficient; Task Scheduling; GreenCloud; DVFS; DNS; Datacenter; Virtual Machine; virtualization.

I. INTRODUCTION

Cloud computing has emerged as a computing infrastructure that enables rapid delivery of computing resources as a utility in a dynamically scalable, virtualized manner. The advantages of cloud computing over traditional computing include agility, low entry cost, device independence, and scalability [1].

Cloud models use the datacenter as the basic unit in its architecture [2] [3]. A cloud model can be viewed as a collection of massively distributed datacenters [4]. In other words, it is a set of cloud service providers that offer services via their datacenters located around the world.

A datacenter [5] or server farm is a massive, centralized repository for the storage, computation, and management of data. A datacenter is a farm for hosting a large number of servers or for processing elements, clusters, and/or considerable amounts of storage to serve customer requests.

Currently, cloud computing provides dynamic services over the Internet, such as applications, data, memory, bandwidth, and IT services. The reliability and performance of cloud services depend on various factors, which include task scheduling. Scheduling can be done at the task level, resource level, or workflow level. In this paper, we mainly focus on task scheduling approaches.

Cloud users send requests to the data center for computing jobs. These requests are called tasks. A task is a small piece of work that should be executed within a given period of time. Task scheduling dispatches the tasks provided by cloud users to the cloud provider, who will assign them to available resources [2].

II. VM SCHEDULING

The assignment of a task by the scheduler is subjected to a number of constraints. Constraints are typically either time constraints or resource constraints. A task may include data entry and processing, software access, and storage functions. The datacenter classifies tasks according to the service-level agreement and requested services. Each task is then assigned to one of the available servers. In turn, the servers perform the requested task. A response or result is transmitted back to the user [6].

Scheduling is a balancing scenario in which processes or tasks are scheduled as per the given requirements and used algorithm. The goal of scheduling algorithms in distributed systems is to spread the load on the processors and to maximize their utilization while minimizing total task execution time. Job scheduling, one of the most famous optimization problems, plays a key role to improve flexible and reliable systems. The main purpose is to schedule jobs to the adaptable resources in accordance with adaptable time, which involves finding out a proper sequence in which jobs can be executed under transaction logic constraints.

In Cloud Computing VM scheduling algorithms are used to schedule the VM requests to the Physical Machines (PM) of the particular Data Center (DC) as per the requirement fulfilled with the requested resources (i.e. RAM, Memory, Bandwidth etc). In today's era there are so many cloud providers in market that have different capacity of Data Centers and Physical Machines available. SalesForce, Amazon, Microsoft office 365 and Windows Azure, Oracle Cloud, Google Apps etc are the leading cloud providers in 2013 [7] [8]. In general scheduling algorithm works in three levels as given below [9]:

1. For the set of VMs find the appropriate Physical Machine.
2. Determine the proper provisioning scheme for the VMs.
3. Schedule the tasks on the VMs.

III. Scheduling Model in Cloud Datacenters

"Fig.1" shows the components of cloud computing scheduling.

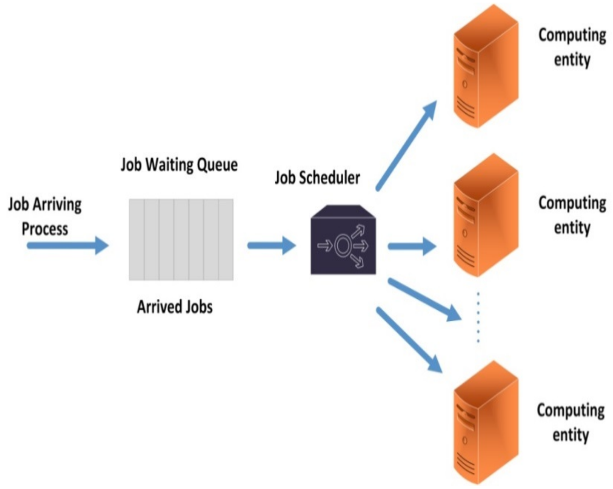

Fig. 1. Scheduling Model in Cloud Datacenters

As shown in the figure, the scheduling model in a cloud datacenter consists of four components, namely, computing entity, job scheduler, job waiting queue, and job arrival process [10].

1. Computing entity is provided through the implementation of a virtualization technique in the cloud computing system. A number of virtual machines that provide computing facilities, such as the operating system and software, are present in the cloud system to process the submitted tasks. A computing entity is characterized by its computing capacity, which indicates the number of instructions it can process in a second.
2. Job scheduler is an important component of the scheduling process in a cloud computing environment. A job scheduler determines the execution order of the jobs waiting in the queue.
3. Job waiting queue is the line of jobs for execution waiting to get assigned to a particular machine.
4. Job arrival process is the procedure in which jobs arrive into the scheduling system.

IV. Scheduling Parameters

A set of parameters are taken into account when building a VM scheduling algorithm. These parameters play an important role to increase overall cloud performance. We explain each parameter in this section.

1. Makespan is the total completion time of all tasks in a job queue. A good scheduling algorithm always tries to reduce the makespan. We consider the following terms in Table 1 to facilitate our understanding of makespan. As shown in "Fig.2" the makespan is defined as the maximum time to complete the ith task on the mth VM [11].

TABLE I. Makespan Parameters Definitions

PARMETERS	Definition
t_i	ith task
mj	mth virtual machine
ci	time when task ti arrives
aj	time when virtual machine mj is available
eij	execution time for ti on mj
cij	time when the execution of ti is finished on mj cij=aj + eij
Makespan	maximum value of cij

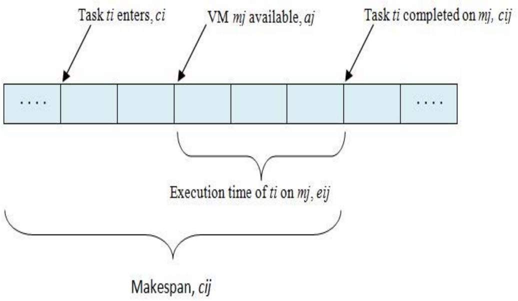

Fig. 2. Makespan and task life time

2. Deadline is defined as the period of time from submission of a task to its completion. A good scheduling algorithm always tries to keep tasks executed within the deadline constraint.
3. Execution time is the exact time taken to execute the given tasks. Minimizing execution time is the ultimate aim of a good scheduling algorithm.

4. Completion time is the time taken to complete the entire execution of a job. Completion time includes execution time and the delay caused by the cloud system. Minimizing the completion time of tasks is considered by many existing scheduling algorithms.

5. Energy consumption in cloud data centers is a current issue that should be given more consideration. Many scheduling algorithms were developed to reduce power consumption and improve performance. Thus, cloud services become environment-friendly.

6. Performance indicates the overall efficiency given by the scheduling algorithm in providing good services to the users per their requirements. A good scheduling algorithm should consider performance on the side of the end user and the cloud service provider.

7. Quality of service includes several user input constraints, such as meeting execution cost, deadline, performance, cost, makespan, and others. These concepts are defined in a service-level agreements (SLAs), which is a contract between the cloud user and the cloud service provider.

8. Load balancing is the method of distribution of the entire load in a cloud network across different nodes and links. No nodes and links should be under-loaded while at the same time there are several other nodes or links are overloaded. Most of the scheduling algorithms try to keep the load balanced in a cloud network to increase the efficiency of the system.

V. EXISTING ENERGY-EFFICIENT TASK SCHEDULING ALGORITHMS

The major energy-consumption components of a datacenter are servers, interconnecting telecommunication networks, and cooling systems [12]. An inefficient use of servers results in significant energy consumption [13] [14]. Energy-efficient scheduling algorithms play a significant role in reducing energy consumption in cloud datacenter.

The following scheduling algorithms are currently prevalent in cloud computing facilities. The main motivation of these scheduling algorithms is to reduce energy consumption within the cloud environment.

[15] proposed three algorithms that mainly focus on handling a request from the users in heterogeneous systems. The first algorithm is a benefit-driven one, in which the tasks are assigned on the best server machines based on a calculated benefit value. This algorithm works for heterogeneous networks. The appropriate methods for homogeneous systems are the power best fit algorithm, which considers the machine with the least power consumption increment for scheduling a task, and the load balancing approach, which is based on the power frequency ratio of each resource. Power frequency ratio indicates the computing capacity of a server.

[16] proposed DENS or data center energy-efficient network-aware scheduling. In this system, the scheduling of tasks is performed by combining network awareness and energy efficiency. DENS satisfies QoS requirements and improves job performance. This system reduces the number of computing servers and avoids hotspots. Network awareness is obtained by using feedback channels from the main network switches. This method has less computational and memory overhead.

[17] proposed e-STAB or Energy-Efficient Scheduling for Cloud Computing Applications with traffic load balancing. The researchers mainly focused on energy-efficient job scheduling that considers traffic load balancing in cloud datacenters. They also looked at the traffic requirements of cloud applications. e-STAB minimizes congestion and communication delays in the network.

[12] implemented an optimized scheduling strategy to reduce power consumption while satisfying task response time constraints during scheduling. This strategy is a greedy approach that selects the minimum number of the most efficient server for scheduling. The tasks are heterogeneous in nature such that they constitute different energy consumption levels and have various task response times. The optimal assignment is based on minimum energy consumption and minimum completion time of a task on a particular machine.

[18] Proposed a green energy-efficient method of scheduling using the Dynamic Voltage Frequency Scaling (DVFS) technique. DFVS reduces the power consumption of infrastructure. Minimizing the number of computing servers and time reduces energy usage and improves resource utilization. The servers are run at different combinations of frequencies and voltages. This method efficiently schedules the tasks to resources without compromising the performance of the system. This method meets the SLA requirements and saves energy.

[19] Proposed two online dynamic resource allocation algorithms for the infrastructure-as-a-service (IaaS) cloud system with pre-emptable tasks. The resource optimization mechanism with pre-emptable task execution can increase cloud utilization. These algorithms improve performance situation where resource contention is fierce. These algorithms are based on the updated information of the current task executions, and they dynamically adjust resource allocation.

[20] Presented the Adaptive Energy-efficient Scheduling (AES) technique, which combines the Dynamic Voltage Scaling (DVS) technique with the adaptive task duplication strategy. In the first phase, an adaptive threshold-based task duplication strategy is proposed, which can obtain an optimal threshold. In the second phase, the groups are scheduled on DVS-enabled processors to reduce processor energy whenever tasks have slack time due to task dependencies. This algorithm can effectively save energy while maintaining good performance.

[21] proposed a two-phase minimum completion algorithm (2PMC) that selects machines for task scheduling based on the expected minimum completion time of all available machines. This algorithm considers the load of the machine before scheduling the task. The task may not have a minimum

execution time on the same machine. The completion time of a task on the machine can be defined as the sum of the execution time of the task on that machine and the ready time of that particular machine.

[22] Proposed an energy-aware "green" scheduler. This "green" scheduler collects the workloads in the minimal computing servers. To ensure high-performance computing workloads, the scheduler continuously tracks the buffer occupancy of network switches on the path. Whenever congestion takes place, the scheduler stays away from the congested routes even if they are led to servers that can meet the computational requirement of the workloads. The idle servers are set into sleep mode (dynamic shutdown DNS scheme), whereas the supply voltage is minimized (dynamic voltage frequency scaling DVFS scheme) on the under-loaded servers.

The previously mentioned task scheduling algorithms consider different metrics for scheduling. In most of the methods, task scheduling is performed based on one or two parameters. A good scheduling algorithm always satisfies the requirements of users by providing them with quality service. The algorithm must consider the benefits received by cloud service provider. The algorithm should always try to reduce cost and power consumption while providing better performance [23]. Load balancing and energy consumption are two main parameters that should be considered in a scheduling algorithm. In addition, an algorithm should facilitate fairness to users when providing services. The combination of significant parameters always results in a good scheduling algorithm, which can be deployed in a cloud environment to provide better cloud services. Such combinations may lead to future enhancements [24]. Table 2 provides an analysis of these scheduling methods, including their findings, tools, and parameters.

TABLE II. COMPARISON OF EXISTING ENERGY EFFICIENT SCHEDULING ALGORITHMS

Algorithm/ Scheduling Method	**Comparison Parameter**				
	Scheduling Parameter	***Tool***	***Findings***	***Environme nt***	***Type of jobs***
DENS: data center energy-efficient network-aware scheduling (2011)	Traffic Load Balancing Energy Efficiency Congestion	GreenCloud	The proposed approach optimizes the tradeoff between job consolidation (to minimize the amount of computing servers) and distribution of traffic patterns (to avoid hotspots in the data center network).	Cloud	Data-Intensive Workloads (DIWs) produce almost no load at the computing servers, but require heavy data transfers
e-STAB: Energy-Efficient Scheduling for Cloud Computing Applications with Traffic Load Balancing (2013)	Energy efficiency Network Awareness Quality of service performance	GreenCloud	Load balancing and energy efficiency is achieved based on traffic load, congestion and delay are avoided.	Cloud	Data-Intensive Workloads (DIWs) produce almost no load at the computing servers, but require heavy data transfers
Task Scheduling & Server Provisioning (2013)	Energy Consumption, Task response time, Deadline	Matlab	Results show that a data center using the proposed task-scheduling scheme consumes on average over 70 times less on server energy than a data center using a random-based task-scheduling scheme.	Cloud	Balanced Workloads (BWs) aim to model the applications having both computing and data transfer requirements.
Minimum Completion Time Algorithm	Completion time Energy Consumption DC load	GreenCloud	The results show an improvement in the datacenter load and power consumption by using 2 phase minimum completion time algorithm to reduce the load and utilized servers.	Cloud	High Performance Computing (HPC).
Energy efficient method using DVFS (2013)	Energy Efficiency Execution Time	CloudSim	Experimental results show that using our method is efficient in reducing the energy consumption and losing light performance of the system.	Cloud	High Performance Computing (HPC).
Benefit Driven, Power Best Fit, Load Balancing (2013)	Energy Consumption, Cost, Load balancing	Cloudsim	Power consumption is reduced and cost is reduced even more number of servers used	Cloud	Balanced Workloads (BWs) aim to model the applications having both computing and data transfer requirements.

Algorithm/ Scheduling Method	**Comparison Parameter**				
	Scheduling Parameter	***Tool***	***Findings***	***Environment***	***Type of jobs***
Green Scheduler (2012)	Energy Efficiency	GreenCloud	Power consumption is reduced by reducing the total number or servers	Cloud	High Performance Computing (HPC).
Dynamic Resource Allocation Algorithms (2012)	Energy consumption, load balancing, response time	Own written simulation environment that acts like the IaaS cloud system.	The energy-aware local mapping in the proposed dynamic scheduling algorithms can significantly reduce the energy consumptions in the federated cloud system.	Cloud	High Performance Computing (HPC).
Adaptive Energy-Efficient Scheduling Algorithm	Energy consumption, Makespan	SimGrid	The algorithm can effectively save energy while maintaining a good Performance.	Cloud	Balanced Workloads (BWs) aim to model the applications having both computing and data transfer requirements.

TABLE III. POWER SAVING PERCENTAGE IN EXISTING ENERGY EFFICIENT SCHEDULING ALGORITHMS

#	**Table Column Head**			
	Algorithm	***Power saving percentage***	***DVFS***	***DNS***
1	DENS algorithm	50%	√	√
2	e-STAB algorithm	47%	√	√
3	Task Scheduling & Server Provisioning	40%	√	√
4	Minimum Completion Time algorithm	50%	√	√
5	Energy efficient method using DVFS	25%	√	√
6	Benefit Driven, Power Best Fit, Load Balancing	12%-13%	×	×
7	Green Scheduler	53%	√	√
8	Dynamic Resource Allocation Algorithm	15%-20%	×	×
9	Adaptive Energy-Efficient Scheduling Algorithm	31.7%	√	×

After an in-depth analysis and investigation of the previously mentioned energy-efficient scheduling algorithms, the power saving percentage among all algorithms were compared, as shown in Table 3. The scheduling algorithms showed a reduction in power consumption in a cloud environment. However, all scheduling algorithms provide a different percentage in power consumption at a range of 12% to 53%. As shown in Table 2, the highest energy-saving algorithms were those that used both DVFS and DNS. These algorithms have power consumption in the range of 25% to 53%. Among all algorithms, the Green Scheduler exhibited the best power consumption at 53%.

VI. CONCLUSION AND FUTURE WORK

Efficient scheduling algorithms play a significant role in the performance of a cloud computing system. This paper studies existing task scheduling algorithms and briefly analyzes each method. Most algorithms perform scheduling based on one or two parameters. A better scheduling algorithm can be developed from existing methods by adding more metrics, which can result in good performance and outputs that can be deployed in a cloud environment in the future. This paper summarizes some existing energy scheduling algorithms used in a cloud environment. and the power-saving percentage in existing energy-efficient scheduling algorithms. It has been determined that the best power-saving percentage level can be achieved by using both DVFS and DNS. In future studies, the Green Scheduler would be a good candidate to be used for comparison with a newly developed energy scheduling algorithm.

VII. REFERENCES

[1] Tsai, W. T., Sun, X., & Balasooriya, J. (2010, April). Service-oriented cloud computing architecture. In Information Technology: New Generations (ITNG), 2010 Seventh International Conference on (pp. 684-689). IEEE.

[2] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., & Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation computer systems, 25(6), 599-616.

[3] McFedries, P. The cloud is the computer. IEEE Spectrum Online, August 2008. Electronic Magazine, available at http://www.spectrum.ieee.org/aug08/6490.

[4] Weiss, A. (2007). Computing in the clouds. Computing, 16.

[5] Stryer, P. (2010). Understanding data centers and cloud computing. Global Knowledge Instructor.

[6] PushpaLatha, K., Shaji, R. S., & Jayan, J. P. (2014). A Cost Effective Load Balancing Scheme for Better Resource Utilization in Cloud Computing. Journal of Emerging Technologies in Web Intelligence, 6(3), 280-290.

[7] Prajapati, K. D. (2013). Comparison of Virtual Machine Scheduling Algorithms in Cloud Computing.

[8] Salot, P. (2013). A survey of various scheduling algorithm in cloud computing environment. IJRET: International Journal of Research in Engineering and Technology, ISSN, 2319-1163.

[9] Frincu, M. E., Genaud, S., & Gossa, J. (2013, May). Comparing provisioning and scheduling strategies for workflows on clouds. In Parallel and Distributed Processing Symposium Workshops & PhD Forum (IPDPSW), 2013 IEEE 27th International (pp. 2101-2110). IEEE.

[10] Yang, B., Xu, X., Tan, F., & Park, D. H. (2011, December). An utility-based job scheduling algorithm for cloud computing considering reliability factor. In Cloud and Service Computing (CSC), 2011 International Conference on (pp. 95-102). IEEE.

[11] Nagadevi, S., Satyapriya, K., & Malathy, D. (2013). A Survey on Economic Cloud Schedulers for Optimized Task Scheduling. Intemational Journal of Advanced Engineering Technology, 4(1), 58-62.

[12] Liu, N., Dong, Z., & Rojas-Cessa, R. (2013, July). Task scheduling and server provisioning for energy-efficient cloud-computing data centers. In Distributed Computing Systems Workshops (ICDCSW), 2013 IEEE 33rd International Conference on (pp. 226-231). IEEE.

[13] Glanz, J. (2012). Power, pollution and the internet. The New York Times, 22.

[14] Fettweis, G., & Zimmermann, E. (2008, September). ICT energy consumption-trends and challenges. In Proceedings of the 11th International Symposium on Wireless Personal Multimedia Communications (Vol. 2, No. 4, p. 6).

[15] Huai, W., Qian, Z., Li, X., Luo, G., & Lu, S. (2013). Energy aware task scheduling in data centers. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), 4(2), 18-38.

[16] Kliazovich, D., Arzo, S. T., Granelli, F., Bouvry, P., & Khan, S. U. (2013, August). e-STAB: energy-efficient scheduling for cloud computing applications with traffic load balancing. In Green Computing and Communications (GreenCom), 2013 IEEE and Internet of Things (iThings/CPSCom), IEEE International Conference on and IEEE Cyber, Physical and Social Computing(pp. 7-13). IEEE.

[17] Kliazovich, D., Bouvry, P., & Khan, S. U. (2013). DENS: data center energy-efficient network-aware scheduling. Cluster computing, 16(1), 65-75.

[18] Wu, X., Deng, M., Zhang, R., Zeng, B., & Zhou, S. (2013). A task scheduling algorithm based on QoS-driven in Cloud Computing. Procedia Computer Science, 17, 1162-1169.

[19] Li, J., Qiu, M., Ming, Z., Quan, G., Qin, X., & Gu, Z. (2012). Online optimization for scheduling preemptable tasks on IaaS cloud systems. Journal of Parallel and Distributed Computing, 72(5), 666-677.

[20] Liu, W., Du, W., Chen, J., Wang, W., & Zeng, G. (2014). Adaptive energy-efficient scheduling algorithm for parallel tasks on homogeneous clusters.Journal of Network and Computer Applications, 41, 101-113.

[21] Mehdi, N. A., Ali, H., Amer, A., & Abdul-Mehdi, Z. T. (2012). Two-Phase Provisioning for HPC Tasks in Virtualized Datacenters. In Proc. International Conference on Emerging Trends in Computer and Electronics Engineering (ICETCEE), Dubai.

[22] Kliazovich, D., Bouvry, P., & Khan, S. U. (2012). GreenCloud: a packet-level simulator of energy-aware cloud computing data centers. The Journal of Supercomputing, 62(3), 1263-1283.

[23] Mohialdeen, I. A. (2013). Comparative Study of Scheduling Al-gorithms in Cloud Computing Environment. Journal of Computer Science, 9(2), 252.

[24] ZHOU, L., CUI, X., & WU, S. (2013). An Optimized Load-balancing Scheduling Method Based on the WLC Algorithm for Cloud Data Centers. Journal of Computational Information Systems.