Scheduling and Load Balancing in Cloud-Fog Computing using Swarm Optimization Techniques: A Survey

May Myat Mon[1], May Aye Khine[2]
University of Computer Studies, Yangon
maymyatmon@ucsy.edu.mm[1] *, mayayekhine@ucsy.edu.com*[2]

Abstract

Cloud computing is an interested and big developing computing technology that maintain data servers and service huge applications to provide end users in many different organization. Although cloud computing gives many benefits of service, but it can hardly accurate the requirements of end users in humans' daily lives. A new computing paradigm called Fog Computing which is an emerging as a necessary and popular computing paradigm to perform Internet of Things (IoT). Fog computing is a middle layer of cloud and IoT. When fog computing is insufficient for the resource requirements of IoT, cloud computing can assist fog computing to get a handle of intensive applications. The IoT applications could choose fog or cloud computing nodes for responding to the resource requirements. Scheduling and load balancing algorithms are necessary for efficient and effective utilization of resources. This paper presents the survey of scheduling and load balancing algorithms in cloud and fog computing environment by using swarm-based optimization techniques.

Keywords*: Fog Computing, Resource Scheduling, Load balancing, Swarm Intelligence, Particle swarm optimization (PSO), Ant Colony Optimization (ACO), Artificial Bee Colony Optimization (ABC).*

1. Introduction

In recent years, the Internet of Things (IoT) become very popular in both industry and academia as a beneficial technology in the daily lives of people. The data extracted from the smart sensors are often transmitted to the cloud data centres and the applications are generally executed by the processors in the data centres [1]. Cloud computing offers many businesses number of benefits such as minimized cost, reduced time, reduced energy, increased storage, scalability and flexibility. Cloud computing can be efficient in provisioning computation and storage resources but it cannot be handle ever increasing amount of resource requirement of IoT, thus fog computing is extended at the edge of network of cloud computing to perform the resource requirements of IoT. The main characteristics of fog computing is to support applications that demand low latency, location awareness, distributed geographical distribution, support for mobility, real time interaction, scalability, heterogeneity and interoperability [2]. Although fog computing has many advantages, it also faces enormous challenges. Resource allocation and task scheduling are one of the key challenges in running IoT applications in a fog computing [3].

This paper is a survey on different resource scheduling and load balancing algorithms for various optimization problems in cloud and fog computing environments. This paper figures out achievements and limitations of scheduling and load balancing processes for improving performance in cloud and fog computing. This paper contains many issues and challenges of different resources in cloud-fog computing environments. The purpose of this paper is to convey the ideas and knowledge of many researchers on different topics.

An arranged collection of interacting creatures into a system is the description of swarm intelligence. Computational study of swarm intelligence includes fish in schools and birds in flocks, wasps, ants, bees and termites. Swarm intelligence is an artificial intelligence discipline. From the behaviour of the individuals in a swarm become the global behaviour. The interaction among individuals plays an essential part in appearing the swarm's behaviour. Optimization techniques can be classified into many ways based on its algorithms, which can be classified as standard optimization, heuristics and meta-heuristic. These techniques perform a major part due to changeability of their nature. Swarm-based optimization technique which can overcome the resource utilization or optimization problems.

Most widely used Swarm Based Algorithm for scheduling and load balancing are:

1) Ant Colony Optimization (ACO)[1]

2) Particle Swarm Optimization (PSO) [SEP]

3) Artificial Bee Colony Optimization (ABC)

This paper includes five parts and the reminders are as follow. Section 2 studies related to scheduling and load balancing. Section 3 provides overview of resource scheduling and load balancing in cloud-fog computing. Section 4 gives swarm-based optimization algorithms. In ending, conclusion is drawn.

2. Related Works

In reference [5], the proposed algorithm is used for resource provisioning in fog computing by using virtualization technique. ERA algorithm has completed bandwidth utilization, transfer cost and overall response time in fog computing environment. This paper can be drawn-out towards run time on request resource allocation during the user can not satisfy this algorithm for their resource requirements to execute.

In reference [6], the proposed algorithm has increased throughput and network utilization, fulfils real tasks within deadline, and maintain data consistency with less complexity to face the current-day requests of end users. This paper was not able to support QoS and security.

S Rasheed, et al. [7] proposed Max-Min scheduling algorithm with advanced service broker policy for allocation of resource on VMs. To manage energy distributions among consumers, smart grids (SGs) model was used in cloud-fog environment. Terminal layer consists clusters of buildings which each has 500 flats which each has a smart meter. This algorithm has outperformed cost, processing time and response time than Round-Robin algorithm.

In reference [8], the proposed algorithm was able to outperform the distribution of load in a more successful way than Delay-Driven Load Distribution in fog computing. This paper needed to evaluate disk I/O operations.

In reference [9], the proposed algorithm was significant in fog computing emerging paradigm to minimize the total response time and especially minimize the total cost.

3. Overview of Scheduling and Load Balancing in Cloud-Fog Computing

In cloud computing, resource scheduling and load balancing are an important part for both providers and users. Cloud computing is a useful and beneficial technology in many different organizations. Many cloud providers use scheduling and load balancing techniques to predict the performance of hardware and software and to give the best services to the end users. For end users especially business people, they do not spend their money and time in discouraging places. So, scheduling and load balancing techniques can give their resource requirements on time. Cloud computing can be efficient in provisioning computation and storage resources but it cannot be handle ever increasing amount of resource requirement of IoT. It needs to schedule resource and load balance of cloud resources.

Fog computing is the edge of the network and closes to IoT as presented in Figure 1. It services the networking services, storage and operation of computing between cloud data center and end devices (end users). The main key of fog computing is to utilize various requests of end users at the edge of network to reduce latency and processing cost. Fog computing can solve immediately some demands of consumers by itself without sending it to cloud computing, so for the customer side, the users are not going to waste many time on their requirements. However, fog computing depends on cloud computing to perform complex problems. But, the fog computing also needs to find the way for the huge number of users' requirements day by day. It needs to solve the requirements with balance, SLA and QoS-aware. So, the load balancing and resource scheduling are major role in fog computing. Fog computing has tremendous benefits such as minimize latency, bandwidth, reliability, move data to the best place for processing, minimize cost of using high computing power.

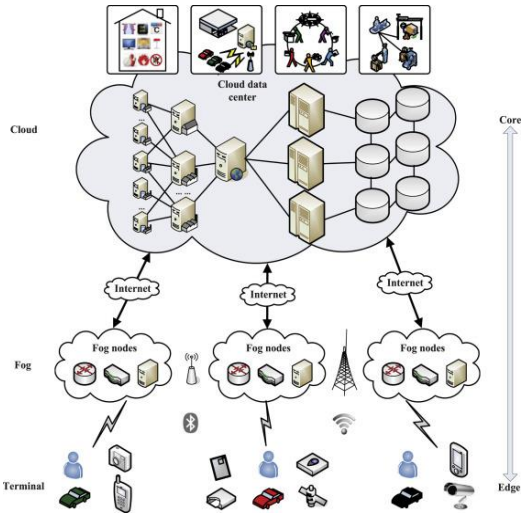

Figure 1: Fog computing architecture

3.1. Resource Scheduling

In cloud computing, to get the economic benefits, resource scheduling is the main objective. Resource scheduling is essential to cloud computing to increase the performance of the whole system and enhance the degree of customer contentment. The fact of the user does not need to increase on software and hardware system is the power of resource scheduling. The goal of scheduling is to optimize one or more objectives by mapping tasks to appropriate resources. There are types of resource scheduling in Figure 2. There are some of the provider-required and customer-required optimization benchmark while scheduling tasks in fog or cloud computing [4]. Provider-desired optimization principle are maximizing resource utilization, throughput, priority constraint, dependency constraint, deadline constraint and budget constraint. Customer-desired optimization principle are minimizing of make span, economic cost, minimization of flowtime, tardiness, waiting time, turnaround time and fairness.

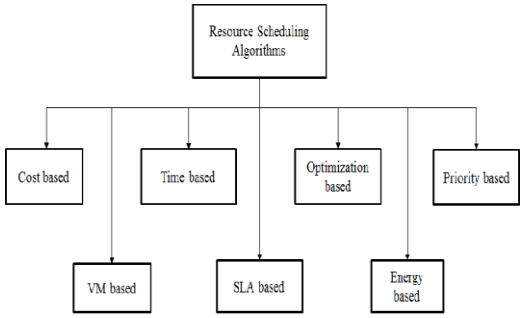

Figure 2: Types of Resource Scheduling

3.2. Load Balancing

Load balancing is focused on keeping the available resource equally busy and avoid overload of one machine with many tasks. Load balancing is very useful for cloud service provider to assign applications requests to various nodes of cloud or fog computing. The nature of load balancing is dynamic because load be different according to end user requests. Load balancing has the various factors such as throughput, makespan time, QoS, performance of system, average waiting time, response time, execution time, fault tolerance, resource utilization, network delay and CPU utilization based on CPU load, network load and using amount of virtual memory. The advantage is not only in terms of using more computational power but also spending less time in switching between tasks. Load balancing can be done in centralized or decentralized, periodic or non-periodic and static or dynamic. The goal of load balancing is to overcome its various factors in cloud and fog computing environment. Different types of load balancing techniques are shown in Figure 3.

4. Swarm Intelligence Algorithms

Swarm Intelligence (SI) is an artificial intelligence discipline, was introduced by Gerado Beni and Jing Wang in 1989. A swarm is considered to be an arranged collection of interacting creatures into a system. Computational study of swarm intelligence includes the algorithms of ants, bees, wasps, termites, fish in schools and birds in flocks.

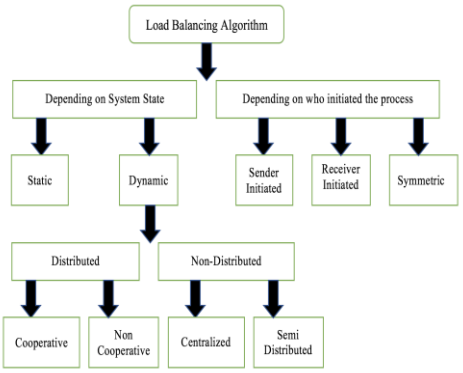

Figure 3: Types of Load Balancing Algorithm

4.1. ACO based Scheduling and Load Balancing Algorithms

The ACO algorithm is a meta-heuristic optimization technique inspired by biological systems to calculate a shortest path between the source and destination to finding optimal solutions. ACO can be used for both static and dynamic combinatorial optimization problems. The key objectives of this algorithm are: balance load of the system, highly efficient, minimize make span, reduce time and improve the ability of balance.

The authors discussed in [10], proposed resource scheduling technique by using Ant Colony Optimization algorithm for load distribution of workloads in cloud to detect overloaded and under loaded nodes in cloud and to improve cost efficiency and resource utilization. In this paper, ants updated continuously an isolated result set more than modernizing their own result set.

Li, et al. [11] discussed cloud task scheduling policy to balance the load of the whole system when attempting to minimize make span of given tasks set by using Load Balancing Ant Colony Optimization

algorithm (LBACO). To achieve task scheduling with load balancing, they used LBACO algorithm. But, this paper has limitation for the heterogeneous processing of the tasks.

X Lu and Z Gu [12] presented ACO based a load-adaptive cloud resource scheduling model to monitor the performance of VMs in real time. This algorithm can easily detect the overload performance, can find the closest idle node fast and the job can move from one VM to another for the next task.

P Mod and M Bhatt [13] proposed dynamic resource scheduling based on ACO for improving cloud performance which provided results efficiently during performance evaluation. This paper did not consider cost, make span and energy domains.

V S Kushwah and S K Goyal [14] proposed ACO algorithm for fault tolerant in cloud computing. This paper applied ACO algorithm for managing fault tolerant during the loads of resources equally.

4.2. PSO based Scheduling and Load Balancing Algorithms

The PSO algorithm incorporates local search methods with global search methods for trying to equalize exploration and exploitation. PSO is a popular technique due to its functionality in large range of applications with little computational cost and its simplicity. It is also a kind of meta-heuristic technique. The major objectives of PSO are: balance load of the system, minimize time and communication cost and increase scalability.

The authors discussed in [15], proposed PSO to overcome the main challenge of cloud computing. This challenge is load balancing because nowadays the number of users in different organizations using cloud computing are increasing for their requirements so the performance of cloud computing is decreasing. This paper focused on minimizing makespan time.

The authors in [16] proposed their scheduling method based on improved PSO (IPSO) for workflow applications and mainly focused on workflow scheduling in cloud-fog environment to gain the optimal solution such as more cost-effective and better performance than PSO approach.

Rodriguez, et al. [17] proposed resource provisioning and scheduling using PSO for minimizing the execution cost of the whole works passes through form the beginning until it is finished and makespan as deadline constraints.

W Gu, D Tang and K Zheng [18] proposed IAPSO algorithm for minimizing make span in job-shop scheduling problem (JSP). IAPSO algorithm can perform optimization, efficiency and stability.

A Dave, Pro. B Patel, Pro. G Bhatt and Y Vora [19] proposed PSO algorithm for finding efficient resources for the problem of resource allocation and load balancing in cloud computing. This algorithm reduced response time of running applications in VMs and appropriates the loads in servers.

4.3. ABC based Scheduling and Load Balancing Algorithms

The ABC algorithm is one of the most newly optimization. It is also a swarm based meta-heuristic algorithm. The good points of this algorithm include: simplicity, flexibility, robustness, need fewer control parameters compared to others searching techniques, and minimize cost with stochastic nature.

The authors in [20] proposed new optimization method called Bees Life Algorithm (BLA) to solve the issue of numerous computational resources requested by mobile users and to ensure the efficient execution of tasks in the fog computing environment. The job scheduling algorithm fixed an optimal task of various jobs submitted to be execute on tradeoff between CPU execution time and memory required by fog computing, using for infrastructure. This Algorithm resulted show better performance of CPU execution time and allocated memory than GA and PSO algorithms. But, this paper needed to consider dynamic job scheduling for the arrival of new requests are being executed in fog computing environment.

The authors in [21] proposed an effective Improved Honey Bee Algorithm in cloud computing to be better make span for both dynamic and static task scheduling. The weak point of this paper was lacking of QoS for both two nature of tasks such as dependent and independent tasks.

S Zahoor, S Javaid, N Javaid, M Ashraf [22] proposed hybrid approach of ACO and ABC (HABACO) model for resource management. The important idea is to supply types of computing services for SG resource management by determining the hierarchical structure of cloud-fog computing. HABACO outperformed response time, processing time and cost than PSO, ABC and ACO.

Bitam and S [23] proposed Bees Life Algorithm (BLA) for optimization, reliability of job scheduling, minimize make span and efficient execution time. To escape local best and agreement solution diversity, using crossover.

Mizan, et al. [24] proposed modified bee life

algorithm for the objective of optimistic value of service, proper utilization of resources. To obtain the best individual solution, using the greedy mechanism as a local search.

Table 1: Comparisons on Scheduling and Load Balancing Approaches of ACO

Algorithm	Authors	Achievement/ Limitation	Experimental Environment	Experimental Scale	Results Compared
ACO	Kumar Nishant, Pratik Sharma, Vishal Krishna, Chhavi Gupta, Kuwar Pratap Singh, Nitin, Ravi Rastogi [10]	Detect overloaded and under-loaded nodes in cloud and improve cost efficiency and resource utilization.	NA	4 resources	NA
	Kun Li, Gaochao Xu, Guangyu, Zhao, Yushuang, Dong, Dan Wang [11]	Minimize make span of given tasks set, but without heteroge-nous processing of the tasks.	CloudSim	100-500 tasks 50 resources	FCFS, Traditional ACO
	Xin Lu, Zilong Gu [12]	Detect the overloaded nodes, quickly finding the nearest idle node and the job can move from one VM to another for the next task.	Xen Server	4 PMs	NA
	Priyanka Mod, Prof. Mayank Bhatt [13]	provide efficient results during performance evaluation, but without cost, make span and energy domains.	CloudSim	20 VMs, 40 Cloudlets, 10000 MB VM image size, 512 MB RAM, 1000 MIPS, 1 Processing unit	Time shared, Space shared
	VS Kushwah, SK Goyal [14]	Manage fault tolerant during the loads of resources equally. Consider using the average best time can design new algorithm to improve reliability and durability of cloudlets.	CloudSim	20 ants	NA

Table 2: Comparisons on Scheduling and Load Balancing Approaches of PSO

Algorithm	Authors	Achievement/ Limitation	Experimental Environment	Experimental Scale	Results Compared
PSO	Jigna Acharya, Manisha Mehta, Baljit Saini [15]	Minimize makespan, but without other different parameters such as throughput, waiting time and so on.	Cloudsim	1 DC, 1 Host, 50 VMs, 10 particles	FCFS
	Rongbin Xu, Yeguo Wang, Yongliang Cheng, Yuanwei Zhu, Ying Xie, Dong Yuan [16]	Achieve the optimal solution such as more cost-effective and better performance than PSO approach. But, consider workflow scheduling algorithm based on multiple objectives optimization in hybrid cloud-fog environment.	MatLab	6 cloud servers, 4 fog servers	PSO
	Maria Alejandra Rodriguez, Rajkumar Buyya	Defines the make span as deadline constraints. However, this paper needs	CloudSim	50-1000 tasks, 6 resources	ICPCP, SCS

	[17]	to consider communication cost and computational cost.			
	W Gu, D Tang, K Zheng [18]	Minimize make span in job-shop scheduling problem (JSP). Perform optimization, efficiency and stability.	Adaptive modulation factor HF	Swarm size 100, Maximum iteration 300	PSO, CGA
	A Dave, Prod. B Patel, Prod. G Bhatt, Y Vora [19]	Reduce response time of running applications in VMs and allocates the loads on the servers.	Xen Server	1 server, 3 VMs	Balance algorithm

Table 3: Comparisons on Scheduling and Load Balancing Approaches of ABC

Algorithm	Authors	Achievement/ Limitation	Experimental Environment	Experimental Scale	Results Compared
ABC	S Bitam, S Zeadally, A Mellouk [20]	At customer side, achieve faster execution time of their tasks at a lowest cost. At service provider side, satisfy service level agreement (SLA) and minimize execution time. But, consider dynamic job scheduling for the arrival of new requests are being executed in fog computing environment.	BLA framework in C++	4 fog servers 20 fog nodes, 5 jobs, 5 tasks	PSO, GA
	SK Vasudevan, S Anandaram, AJ Menon, A Aravinth [21]	Minimize make span for both dynamic and static task scheduling. But consider the lacking of QoS for both two nature of tasks such as dependent and independent tasks.	CloudSim, WorkflowSim	25-1000 cloudlets 25-100 tasks	HBB-LB
	S Zahoor, S Javaid, N Javaid, M Ashraf, F Ishmanov, M K Afzal [22]	Perform response time, processing time and cost. But, consider to extend towards the management of multiple load balancing applications.	Cloud Analyst	2-5 VMs 2-5 fogs	PSO, ACO, ABC
	Salim Bitam [23]	minimize make span and efficient execution time. Next, it can be studied for the dynamic job scheduling and can be treated the real-time execution.	NA	5 jobs, 20 Data centers	GA
	Tasquai Mizan, Shah Murtaza Rashid AI Masud, Rohaya Latip [24]	proper utilization of resources. To obtain the best individual solution, using the greedy mechanism as a local search. This algorithm can be studied dynamic job scheduling and real- time job scheduling.	NA	3 jobs, 3 CSC and resources	FA, GA

5. Conclusion

The brief survey of different resource scheduling and load balancing for various optimization algorithms in cloud and fog computing has studied the most suitable research papers of scheduling and load balancing algorithms. This paper has figured out achievements and limitations of scheduling and load balancing processes for improving performance in cloud and fog computing. Every algorithm has pros and cons in many application areas. There is a need of scheduling and load balancing which considers QoS parameters like communication cost, computational cost, makespan, energy management, availability, execution time and security and SLA violation rate etc. In cloud computing environment, the researchers investigated many research papers for energy efficient, security, load balancing, QoS and SLA. But, further

investigation is needed to address above requirements in fog computing environment.

References

[1] Y.Kong, M.Zhang, and D.Ye, "A belief propagation-based method for task allocation is open and dynamic cloud environments," vol. 115, pp. 123-132, (2017).

[2] P Hu, S Dhelim, Huansheng Ning, T Qiu, "Survey on Fog Computing: Architecture, Key Technologies, Applications and Open Issues," JNCA (2017).

[3] Puliafito C., Mingozzi E., Anastasi G.: "Fog Computing for the Internet of Mobile Things: issues and challenges". Pp 1-6, IEEE, Hong Kong (2017).

[4] M. Kalra, S. Singh, "A review of metaheuristic scheduling techniques in cloud computing". Egyptian Informatics Journal (2015).

[5] S. Agarwal, S. Yadav, and A. K. Yadav, "An efficient architecture and algorithm for resource provisioning in fog computing" Int. J. Inf. Eng. Electron. Bus., vol. 8, no. 1, pp. 48–61, January, (2016).

[6] M. Verma, N. Bhardwaj and A. K. Yadav, "Real Time Efficient Scheduling Algorithm for Load Balancing in Fog Computing Environment" Int. J. Inf. Technol. Computer Sci, pp. 1-10, April (2016).

[7] S Rasheed, N Javaid, S Rehman, K Hassan, F Zafar and M Naeem "A Cloud-Fog Based Smart Grid Model Using Max-Min Scheduling Algorithm for Efficient Resource Allocation", pp. 273–285, (2018).

[8] C Li, H Zhuang, Q Wang and X Zhou "SSLB: Self-Similarity-Based Load Balancing for Large-Scale Fog Computing" Arabian Journal for Science and Engineering, pp. 1-12, (2018).

[9] T Choudhari, M Moh, T S Moh, "Prioritized Task Scheduling in Fog Computing", March 29-31, Richmond, KY, USA, (2018).

[10] Nishant, K., Sharma, P., Krishna, V., Gupta, C., Singh, K.P., Rastogi, R. "Load balancing of nodes in cloud using ant colony optimization". UKSim 14th International Conference on Computer Modelling and Simulation (UKSim), pp. 3–8. IEEE (2012).

[11] Li, K., Xu, G., Zhao, G., Dong, Y., Wang, D., "Cloud Task Scheduling Based on Load Balancing Ant Colony Optimization", Sixth Annual China grid Conference, Liaoning, IEEE, pp. 3–9 (2011).

[12] X Lu and Z Gu," A load-adaptive cloud resource scheduling model based on ant colony algorithm", IEEE, (2011).

[13] P Mod, M Bhatt, "ACO based Dynamic Resource Scheduling for Improving Cloud Performance", IJSETR, (2014).

[14] V S Kushwah, S K Goyal, "A Basic Simulation of ACO Algorithm Under Cloud Computing for Fault Tolerant".

[15] J. Acharya, M Mehta, B. Saini. "Particle Swarm Optimization Based Load Balancing in Cloud Computing". Communication and Electronics Systems (ICCES), International Conference (2016).

[16] R Xu, Y Wang, Y Cheng, Y Zhu, Y Xie, D Yuan, "Improved Particle Swarm Optimization based Workflow Scheduling in Cloud-fog Environment," (2018).

[17] M A Rodriguez, R Buyya, "Deadline Based Resource Provisioning and Scheduling Algorithm for Scientific Workflows on Clouds", IEEE, (2014)

[18] W Gu, D Tang, K Zheng "Minimizing makespan in Job-shop Scheduling Problem Using an Improved Adaptive Particle Swarm Optimization Algorithm", IEEE, (2012).

[19] A Dave, Pro. B Patel, Pro. G Bhatt, Y Vora, "Load Balancing in Cloud Computing Using Particle Swarm Optimization on Xen Server", IEEE, 2017.

[20] Bitam, S., Zeadally, S., Mellouk, A.: "Fog computing job scheduling optimization based on bees swarm". Enterp. Inf. Syst. 12, 1–25 (2017).

[21] SK Vasudevan, S Anandaram, AJ Menon, A Aravinth. "A Novel Improved Honey Bee Based Load Balancing Technique in Cloud Computing Environment". Asian Journal of Information Technology 15 (9), 1425-1430, (2016).

[22] S Zahoor, S Javaid, N Javaid, M Ashraf, F Ishmanov, M K Afzal, "Cloud-Fog Based Smart Grid Model for Efficient Resource Management", Sustainability,19 June (2018).

[23] Bitam, S, "Bees Life Algorithm for Job Scheduling in Cloud Computing", ICCIT, (2012).

[24] Mizan, T, Masud, S.M.R.A, Latip, R, "Modified Bee Life Algorithm for Job Scheduling in Hybrid Cloud", International Journal of Engineering and Technology. 974-979, Vol. 2, No. 6, June, (2012).