C.Arun *et al*, International Journal of Computer Science and Mobile Applications,
Vol.6 Issue. 7, July- 2018, pg. 1-6 **ISSN: 2321-8363**
UGC Approved Journal
Impact Factor: 5.515

SURVEY ON LOAD BALANCING IN MOBILE CLOUD COMPUTING

Mr. C.Arun
Research Scholar, PG & Research Department of Computer Science, Sudharsan College of Arts & Science, Pudukkottai
E-Mail: arunсmails@gmail.com

Dr. K.Prabu
Assistant Professor, PG & Research Department of Computer Science, Sudharsan College of Arts & Science, Pudukkottai
E-Mail: kprabu.phd@gmail.com

ABSTRACT: Mobile Cloud Computing (MCC) is applications, Internet-based data, and related services accessed via smartphones, laptop computers, tablets and other portable devices. The grouping of cloud computing, wireless communication, mobile computing devices is called mobile cloud computing which allows users an online access. By using MCC, the processing and the storage of intensive mobile device jobs will take place in the cloud system and the results will be back to the mobile device. But the mobile cloud computing have some issues like power consumption, bandwidth, mobility and security. In this paper we discuss about various load balancing techniques. Using the mobile devices for accessing the cloud it needs an efficient load balancing technique for offloading the data to the users.

***Keywords:* mobile device, mobile computing, cloud computing, power consumption, bandwidth, mobility, security.**

1. INTRODUCTION

Mobile cloud computing (MCC) at its simplest, refers to an infrastructure where both the data storage and data processing happen outside of the mobile device. Mobile cloud applications move the computing power and data storage away from the mobile devices and into powerful and centralized computing platforms located in clouds, which are then accessed over the wireless connection based on a thin native client. Mobile devices face many resource challenges (battery life, storage, bandwidth etc.) Cloud computing offers many advantages to users by allowing them to utilize infrastructure, platforms and software by cloud providers at low cost and elastically in an on-demand fashion. Mobile cloud computing provides mobile users with data storage and processing services in clouds, obviating the need to have a powerful device configuration (e.g. CPU speed, memory capacity etc), as all resource-intensive computing can be performed in the cloud in figure 1. According to a recent study by ABI Research, more than 240 million businesses will use cloud services through mobile devices by 2015. That traction will push the revenue of mobile cloud computing to $5.2 billion. Mobile cloud computing is a highly promising trend for the future of mobile computing. Mobile devices are connected to the mobile networks via base stations that establish and control the connections and functional interfaces between the networks and mobile devices. Mobile users' requests and information are transmitted to the central processors that are connected to servers providing mobile network services. The subscribers' requests are delivered to a cloud through the Internet. In the cloud, cloud controllers process the requests to provide mobile users with the corresponding cloud services.

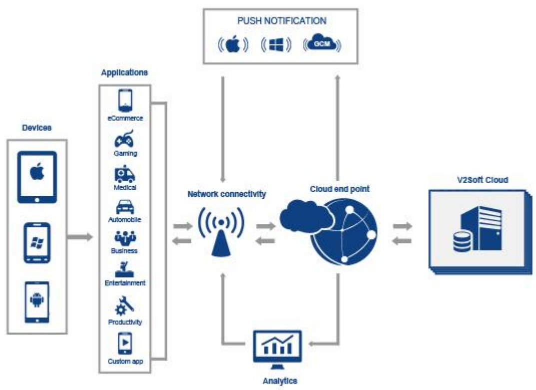

Figure 2. Architecture of Mobile Cloud Computing

2. USE OF CLOUDLET IN MOBILE CLOUD COMPUTING

A cloudlet is a computer or a group of computers connected to the Internet and accessible to nearby mobile devices. If the mobile devices do not wish to offload to the cloud due to cost and delay, a nearby cloudlet can be used [1]. Hence, mobile users can meet the demand for interactive response by reduced-delay, single-hop, and high-bandwidth wireless access to the cloudlet [1]. If no cloudlet is found nearby, the mobile device may access the distant cloud or, in the worst possible case, make use of its own resources. Despite the fact that cloudlets successfully deal with the limitations of high WAN latency, they still have two disadvantages [1]. First, mobile users remain dependent on the service provider for providing such cloudlet infrastructure in LAN networks. To alleviate this constraint, a more dynamic cloudlet is created to connect all devices in the LAN network can cooperate in the cloudlet. The second shortcoming of virtual machine (VM)-based cloudlets [1] is the coarse granularity of VMs as an element of allotment. Instead of executing the whole application remotely in the VM, improved performance can be realized by dynamically partitioning the application into components. Moreover, as cloudlet resources are limited, there is a strong probability that the cloudlet runs out of resources when many users run their applications entirely in the cloudlet infrastructure. This limitation can be dealt with if the applications are offloaded in components rather than as a whole. These application components are distributed between the cloudlets. This cloudlet is not fixed; mobile devices can join or leave the cloudlet at runtime. These features eliminate the disadvantages of conventional cloudlets as well as provide a solution to the high WAN latency problem associated with the cloud in figure 3.

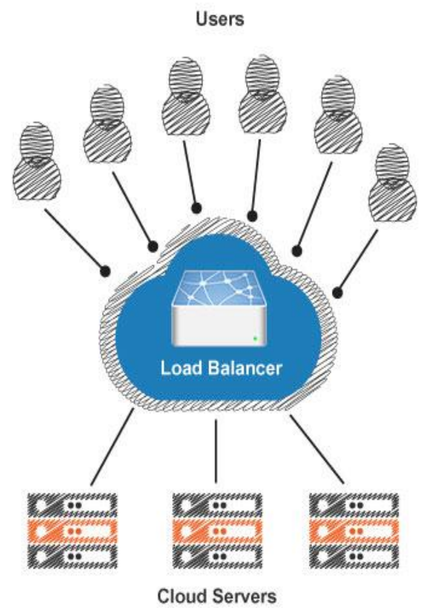

Figure 4. Load balancing in Cloud

3. IMPROVING QUALITY OF SERVICE WITH CLOUDLETS

The basic idea of MCC is to run computation-intensive applications in the resource-rich cloud servers rather than in the resource-limited mobile devices. This trait of offloading applications from mobile devices to the cloud via the Internet results in significant energy saving of the mobile devices. However, long WAN latency is a major disadvantage with the employment of MCC [2]. WAN delays in the crucial course of user interaction can impair usability and affect the QoS. As the latency increases, interactive response suffers. In order to obtain the benefits of MCC without being WAN limited, cloudlets can be used [1, 2]. When mobile devices do not want to offload to the cloud because of delay and cost, a nearby cloudlet can be used [2] in figure 5.

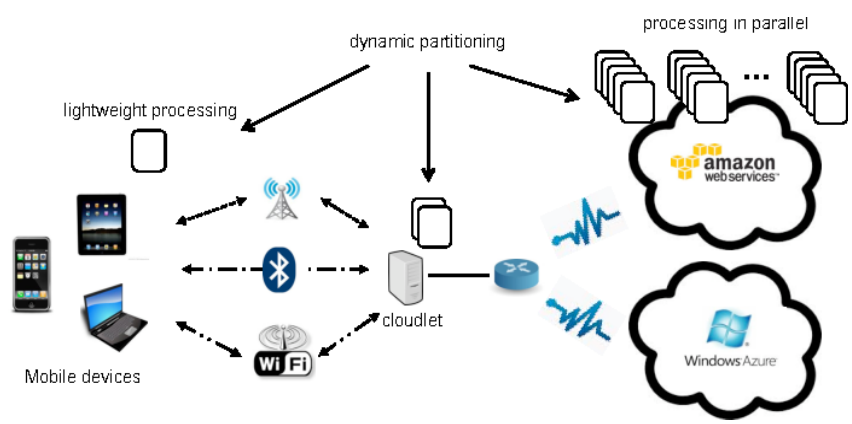

Figure 6. Load balancing in Mobile Cloudlet Environment

ISSN: 2321-8363
UGC Approved Journal
Impact Factor: 5.515

4. LOAD BALANCING AMONG CLOUDLETS

A cloudlet may be overloaded at some point in time and may share its load with other cloudlets. An overloaded cloudlet may experience delay in providing service or may even provide incorrect results. This situation may hamper the QoS and in turn the QoE. Thus, load balancing among cloudlets is very essential to avoid such circumstances. For load balancing it is important to maintain a load threshold value, which would indicate a maximum load value a cloudlet may have based on its processing capacity of the CPU. A cloudlet cannot accept any load if its current load is at or above the load threshold. Cloudlets with a load below the threshold can accept more load. In this way, the load of each cloudlet can be balanced.

5. LOAD SHARING AMONG CLOUDLETS

The cloudlets can also share the load with other cloudlets. Hence, a cloudlet to which an application is offloaded can split the application into multiple components in such a way that the split components have negligible interdependency. Interdependency refers to the condition in which the output of one component is the input of another component. It is minimized so that the communication between the cloudlets is minimized, and in turn the cost of communication in terms of number of messages transferred between the cloudlets will be less. In addition, the energy required for transferring and receiving messages between cloudlets will be reduced as well. Each component thus obtained will be offloaded to a nearby cloudlet whose load is below the threshold. Each cloudlet will send the result of computation to the cloudlet to which the application was offloaded. This cloudlet will finally combine all the intermediate results to generate the final result and then send it to the requesting device.

P. Vijay et. al., [3] proposed a dynamic mobile cloud computing architecture framework to use global resources instead of local resources. The proposed framework the usefulness of job sharing workload at runtime reduces the load at the local client and the dynamic throughput time of the job through Wi-Fi Connectivity.

E. Arun et. al., [4] a novel method for efficiently allocating requirements to the available resources is proposed. Based on the prediction method a single cloud domain is selected from the multi cloud domain system. Then prioritize the nodes, based on the results obtained from the prediction approach. The priority ranks are arranged in descending order. The best resource allocation in the selected node is determined by using genetic algorithm. Multi objective optimization method gives the constrained values for repetition of genetic approach. Using this priority based selection of nodes; unimportant node selection time is reduced, irrelevant node usage eliminated and the network congestion is reduced. The proposed model was efficiently place the request to the appropriate resources on the nearest cloud domain and find out the best VM (virtual machine) for the individual request. Usage of rejected requests handling queues effectively handles the rejected requests. So the request rejection problem is eliminated.

B.Kim et. al., [5] proposed a job load balancing scheme is that considers personal usage patterns and the dynamic resource state of the mobile devices. The delay time for computer job processing is minimized through dynamic job reallocation and adaptive job allocation in the disability state that occurs due to unexpected problems and to excessive job allocations by the mobile devices providing the resources for the mobile cloud computing. In order to validate the proposed load balancing scheme, an adaptive mobile resource management without cloud server (AMRM) protocol was designed and implemented, and the improved processing speed was verified in comparison with the existing offloading method. The improved job processing speed in the mobile cloud environment is demonstrated through job allocation based on AMRM and by taking into consideration the idle resources of the mobile devices. The resource waste of the mobile devices is minimized through adaptive offloading and consideration of both insufficient and idle resources.

S.Mittal et. al., [6] introduced an Optimized Task Scheduling Algorithm which adapts the advantages of various other existing algorithms according to the situation while considering the distribution and scalability characteristics of cloud resources.

X.Wei et. al., [7] presented a Hybrid Local Mobile Cloud Model (HLMCM) by extending the Cloudlet architecture also formulated the application scheduling problems in HLMCM and bringing forward the Hybrid Ant Colony algorithm based Application Scheduling (HACAS) algorithm and validated the efficiency of the HACAS algorithm by simulation experiments.

K.Singh et. al., [8] proposed a new algorithm is load balancing for mobile clouds. The load balancing during job placement plays a critical role in energy consumption of cloud computing environment.

D.Yao et. al., [9] presented an energy efficient task scheduling strategy (EETS) to determine what kind of task with certain amount of data should be chosen to be offloaded under different environment also evaluated the scheduler by using an Android smartphone. It was achieved 99% of accuracy to choose the right action in order to minimize the system energy usage.

C.Chen et. al., [10] proposed a Heterogeneous Mobile Cloud (HMC) computing design that efficiently utilizes the communication and computation resources to support data storage and data processing services in a group of mobile devices. Each mobile device may have different energy, communication and computation capabilities, but our Mobile Storage & Processing System (MSPS) ensures that: i) the communication and computation tasks are executed in an energy efficient manner, ii) task allocation considers device heterogeneity and achieves system-wide load balancing, and iii) the stored data are fault-tolerant.

N. Gupta et. al., [11] proposed a load balancing scheme based on Genetic Algorithm (GA). The algorithm advances to balance the load of the mobile cloud infrastructure while trying to minimize the processing time or responsiveness of tasks with reduced number of migrations of virtual machines and improving the resource utilization by dividing the computing capacity of a datacenter into n number of virtual machines executing the number of requests at the same time and thereby improving the performance. The proposed load balancing scheme has been implemented using the cloudsim simulator.

X.Lin et. al., [12] presented a novel algorithm is, which starts from a minimal-delay scheduling solution and subsequently performs energy reduction by migrating tasks among the local cores or between the local cores and the cloud. A linear-time rescheduling algorithm was proposed for the task migration. The proposed algorithm was achieved a maximum energy reduction by a factor of 3.1 compared with the baseline algorithm. (Less)

H.Mansouri et. al., [13] formulated a utility maximization problem that takes the energy consumption, delay, and price of cloud services into account and obtain the optimal scheduling for both delay-sensitive and delay-tolerant applications. On the cloud service provider (CSP) side, we determine the optimal pricing strategy by formulating a profit maximization problem, which is non-convex in general also proposed an algorithm using convexification and primal-dual methods to mitigate the non-convexity. The proposed scheduler effectively balances the tradeoff between the energy consumption and delay. The CSP was improved its profit by up to 25% compared with static and dynamic pricing strategies.

Y.Li et. al., [14] proposed a novel Energy-aware Dynamic Task Scheduling (EDTS) algorithm to minimize the total energy consumption for smartphones, while satisfying stringent time constraints and the probability constraint for applications. The EDTS algorithm can significantly reduce energy consumption for CPS, as compared to the critical path scheduling method and the parallelism-based scheduling algorithm.

N.Kapoor et. al., [15] proposed a model for job-oriented resource scheduling in a mobile cloud computing environment. The workflow was allocated or scheduled to the process which gives the available resources such as RAM, Bandwidth, MIPS etc. They construct the analysis of resource scheduling algorithms. The waiting time and turn-around time of two algorithms, viz. First Come, First Serve and Priority have been taken into consideration. From this, it has been computed that Priority Algorithm has the lowest time parameters and is the most efficient algorithm for resource scheduling.

6. CONCLUSION

Mobile devices moved from one place to another place. So the dynamic cloud selection algorithm must be used for selecting cloud or cloudlet. Mobile offload it data on cloud or cloudlet. Cloud load balancing is the process of distributing computing resources and workloads in a cloud computing environment. Load balancing permits enterprises to manage application or workload demands by allocating resources among servers, multiple computers and networks. Cloud computing supports to share data and deliver many resources to clients. Client has to pay only for those resources as much them utilized. Cloud computing stores the data and distributed resources in the open location. The amount of data storage rises quickly in open environment. So, load balancing is a main issue in cloud environment. Load balancing is used to distribute the dynamic workload over multiple systems to confirm that no single system is overloaded. It helps in appropriate utilization of resources .It also increase the performance of the whole system.

REFERENCES

[1]. M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, The case for VM-based cloudlets in mobile computing, IEEE Pervasive Computing, 8(4), 14–23, 2009.
[2]. T. Verbelen, S. Pieter, D. T. Filip, and D. Bart, Cloudlets: Bringing the cloud to the mobile user, in Proceedings of the Third ACM Workshop on Mobile Cloud Computing and Services, New York, pp. 29–36, 2012.
[3]. P.Vijay and V. Verma,"Opportunistic job sharing for mobile cloud computing", DOI: 10.5121/ijccsa.2014.4402. 9.
[4]. E. Arun, A.Reji,, P. M.Shameem and R. S. Shaji,"A Novel Algorithm for Load Balancing in Mobile Cloud Networks: Multi-objective Optimization Approach", DOI 10.1007/s11277-017-4665-6.
[5]. B.Kim, H.Byun, Y. Heo and Y.Jeong," Adaptive Job Load Balancing Scheme on Mobile Cloud Computing with Collaborative Architecture", doi:10.3390/sym9050065.
[6]. S.Mittal and A. Katal," An Optimized Task Scheduling Algorithm in Cloud Computing", DOI 10.1109/IACC.2016.45
[7]. X.Wei, J.Fan, Z.Lu, and K. Ding," Application Scheduling in Mobile Cloud Computing with Load Balancing", http://dx.doi.org/10.1155/2013/409539
[8]. K.Singh, "Energy Efficient Load Balancing Strategy for Mobile Cloud Computing", International Journal of Computer Applications (0975 – 8887) Volume 132 – No.15, December2015
[9]. D.Yao, C.Yu, H.Jin, and J.Zhou," Energy Efficient Task Scheduling in Mobile Cloud Computing", DOI: 10.1007/978-3-642-40820-5_29.
[10]. C. Chen, R.Stoleru and G.ie," Energy-efficient Load-balanced Heterogeneous Mobile Cloud", DOI: 10.1109/ICCCN.2017.8038422.
[11]. N. Gupta and P.Singh, "Load Balancing Using Genetic Algorithm in Mobile Cloud Computing", International Journal of Innovations in Engineering and Technology (IJIET), Vol. 4 Issue 1 June 2014, PP 157-162
[12]. X.Lin, Y.Wang, Q.Xie and M.Pedram," Energy and Performance-Aware Task Scheduling in a Mobile Cloud Computing Environment", DOI: 10.1109/CLOUD.2014.35
[13]. H.Mansouri , V.W. S. Wong and Robert Schober," Joint Optimal Pricing and Task Scheduling in Mobile Cloud Computing Systems", DOI: 10.1109/TWC.2017.2707084
[14]. Y.Li, M.Chen ; W.Dai and M. Qiu ,"Energy Optimization With Dynamic Task Scheduling Mobile Cloud Computing", DOI: 10.1109/JSYST.2015.2442994
[15]. N.Kapoor and P.Kakkar,"Workflow Scheduling in Mobile Cloud Computing Environment", International Conference on Communication, Computing & Systems (ICCCS–2014)

