Productive Hybrid Cloud Management In Distributed Environment

Mr. Nitin Kamble[1], Prof. P. B. Sahane[2]
Department of Computer Engineering
[1]Research Scholar,PK Technical Campus, Pune, India
[2]Assistant Professor,PK Technical Campus, Pune, India

Abstract-*Enormous growth of the mobile applications and emerging of cloud computing concept, mobile cloud computing (MCC) has been introduced to be a potential technology for mobile services. MCC integrates the cloud computing into the mobile environment and overcomes obstacles related to the performance (e.g., battery life, storage, and bandwidth), environment (e.g., heterogeneity, scalability, and availability), and security (e.g., reliability and privacy) discussed in mobile computing. Multimedia Cloud Service Provider (MCSP) provides services to the multimedia users which may involve caching, rendering, transcoding etc. of multimedia data. The major issue or limitation of MCSP is it cannot provide multimedia services to the geographically distributed users as its own private clouds are physically available at one or very few locations. Due to the high demand in this area, it is possible to extend these services provided by MCSP to the geographically distributed users by hiring resources from public clouds. Here, we will focus on how we can extend MCSP services to the users who are located at different locations geographically by hiring resources from hybrid cloud and how to maximize the profit gain. To achieve this goal, first step is designing of a model which will help in managing the resources in private as well as public clouds. This paper will also focus on the routing and access controlling of the multimedia requests. Based on the demand, the system will allocate resources from hybrid clouds. To make the service model more and more practical, different issues such as time granularities are considered in reserving resources. Lyapunov technique is used to propose an online algorithm to manage requests and resources efficiently in a cloud. Each request coming from geographically distributed users associated with a latency which is also considered while designing the service model.*

Keywords- MCSP, Hybrid Cloud, Data center management

I. INTRODUCTION

Mobile devices (e.g., smartphone, tablet pcs, etc) are increasingly becoming an essential part of human life as the most effective and convenient communication tools not bounded by time and place. Mobile users accumulate rich experience of various services from mobile applications (e.g., iPhone apps, Google apps, etc), which run on the devices and/or on remote servers via wireless networks. Due to the fast and rapid development in Internet and Internet services, the need of multimedia services increases. Especially, mobile networks are rapidly growing area which is widely using online multimedia services such as sharing images and videos, video conferencing, social networks, online games etc. These multimedia services are nothing but a sequence of media processing task which should be carried out sequentially. These tasks are dynamic in nature and it may need good amount of resources to process [1]. Also due to the size and shape of the multimedia data, there is a need of huge amount of memory to store. To process multimedia services it needs lots of computing resources and memory. These tasks are very time consuming and it needs high amount of energy [2]. Looking at mobile devices where memory, computing power and energy are constraints, supporting multimedia services on smart phones are really a challenging task [3]. One of the solution to this problem is MCSP may adopt hybrid cloud structure through which whenever required computing resources are available on rent and it can respond to the request on timely fashion. As computing tasks are moved to the cloud resources, we can save the energy consumption of smart phones or user level devices [4]. One thing should be noted that the quality of the services provided by multimedia cloud service providers is heavily dependent on the network infrastructure uses by end users. MCSP has its own data centers. It is obvious that the data centers which are closed to the end user may reduce the network latency, improve the user experiences [7]. The user could be located at geographically distributed locations and hence there is a need to have a Multimedia Cloud Service Provides at different locations geographically. But managing operational cost as well as expenditure cost for the data center are very very high and having multiple data center at distributed locations are not possible for most of the MCSPs. There are good amount of multimedia cloud service providers and they have their own private cloud. Each multimedia cloud service providers have capacity to handle specific amount of user requests.

If user requests increases, it can rent out the resources available in public cloud and serves the user requests. The figure Fig.1 shows the hybrid cloud environment and multimedia services. Using hybrid cloud it is possible to extend the multimedia cloud service providers services to the users who are distributed to multiple locations geographically.

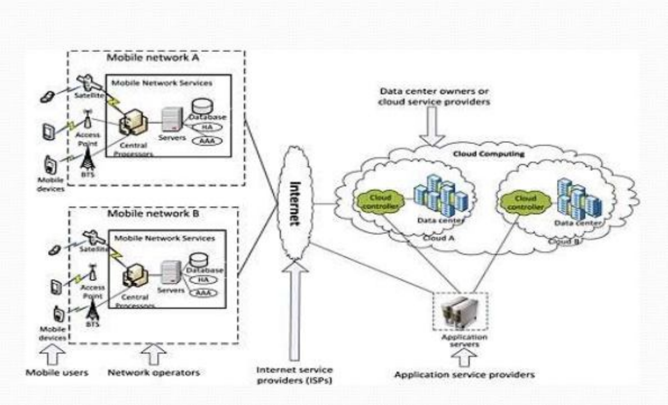

Fig. 1. Hybrid cloud environment and multimedia services.

But to improve the quality of services provided by MCSP as well as the performance and efficiency, we need to handle several issues. First of all, the users who are geographically distributed across multiple locations may use different network infrastructure. This means that the every user has different network latency and it should be considered carefully. So all users are bounded to prefer worst case network latency. Second issue is management of servers/VMs in a public as well as private cloud. The servers of private clouds should coordinate with the VMs of public clouds so that SLA will not violate. Third, we should have highly efficient online request handling and scheduling algorithm to handle dynamic requests and based on the system status it will route the requests and update the system status Fourth, management of resources/servers from private and public cloud very efficiently. Finally, the MCSP provides services to the geographically distributed users; and hence the system should be highly scalable. The contribution of the work can be defined as 1. An online algorithm which handles user requests more efficiently. 2. An online algorithm to manage resources in private cloud. 3. An online algorithm to manage Virtual machines in public cloud efficiently.

II. LITERATURE SURVEY

The multimedia processing tasks needs high storage and computing resources [1]. Processing multimedia processing tasks on smart phones are always challenging tasks. It consumes energy a lot [2] [3]. It has been observed that moving out multimedia processing tasks from end user devices such as smart phones to cloud reducing the energy consumption significantly[5][6]. Previous research focuses on managing multimedia services in a single data center of a private cloud [7]-[9]. Nan et al. did research on request scheduling and allocation of resources and used queuing theory to find out the performances of multimedia services [7]-[9]. Video transcoding and related mechanism is studied in details in [9]. User requests for multimedia processing are handled in details and designed in [8] where this mechanism routes the requests to virtual machines which are scalable to minimize the service provisioning cost. Lyapunov optimization algorithm has been defined in [6] which is a powerful and robust algorithm for task scheduling. The content delivery scheme which is based on cloud is defined in [5]. Hu at al elaborating more on issues of online request handling and processing in (CDN) Content Delivery Network.

III. SYSTEM IMPLEMENTATION

We will leverage the Lyapunov optimization technique which is defined in [6]. This technique is explaining an online request handling and scheduling algorithm.

The proposed system is divided into following algorithms

1. Online distributed scheduling algorithm
2. Server Management in Private DC *j*
3. VM Management in Public DC *j*
4. Online distributed hybrid cloud management

TABLE 1: NOTATION

Notation	***Meaning***
$Q_{i,j(t)}$	Total number of buffered request from zone i in TS *t* in DC *j*
$G_{i,j(t)}$	Virtual Queue for $Q_{i,j(t)}$
$x_{i,j(t)}$	The number of working servers /VMs that are processing the requests
$x_{i(t)}$	Total number of working servers/VMs in DC *i* in TS *t*
$\gamma_{i,j}$	Average communication cost per request from zone *i* in DC *j*
V	Parameter to balance the queue stediness
e_j	Resource in each server/VM in a DC *j*
S_j	Number of working servers or rented VMs in public cloud in a DC *j*
$y_{j(t)}$	Total number of working servers/VMs in DC *j* in TS *t*
$y_{j,k(t)}$	The state of the kth server/VM in DC j in TS t
∇t	Time interval (time slot)
T_g	Time granularity for VM rentals
ηi	Revenue per served request in zone *i*

A. Online Distributed Scheduling Algorithm

Based on the Lyapunov optimization technique and its formulation [6], we suggest an online request scheduling algorithm which can work in the distributed manner. This algorithm can handle both the request scheduling and the resource management for the VMs/Servers of the public and private clouds.

Lyapunov optimization technique shown that for a zone *i*, the proxy have to route all the admitted requests to the Data center DC, whose request queue has the shortest length. If the shortest queue $Q_{i,j*(t)}$ is longer than $V.\eta i$then the proxy in zone *i* supposed to drop all the incoming multimedia requests in the given time slot TS(*t*). Otherwise all the other requests should be pushed into the queue and should be routed to DC *j*

B. Server Management Algorithm in Private Cloud

For a DC *j* in the private cloud, there is a need to decide the server allocation variable $x_{i,j(t)}$for queue. Meanwhile, once the server is allocated, the DC should update the states of the servers (i.e., the decision variable $y_{j,k(t)}$). Then, we obtain the optimization sub-problem for each private DC.

Algorithm 1 explains the server management in a private DC *j*. Initialization is shown in Line 1 and 2. The queue is sorted in ascending order. Basically we sort all queues $\{Q_{i,j(t)}\}$ in data center DC *j* in descending order of $Q_{i,j(t)}+ G_{i,j(t)} - V.y_{i,j}$ After that, in the sorted order, we will allocate servers to each queue which is shown in lines 3-9. Specifically we can allocate at the most $\lceil \frac{Q_{i,j(t)}}{e_j} \rceil$ servers to the each queue. Then Line 10-12 updates the server's state μ_j

Algorithm 1: Server Management in Private DC j
Input:
$Q_{i,j(t)}$and $G_{i,j(t)}$

1. $x_{i,j(t)} = x_{j(t)} = 0, \ \forall i$;
2. Sort the queues $\{ Q_{i,j(t)}\}$ in descending order of $Q_{i,j(t)}+ G_{i,j(t)} - V.y_{i,j}$;
3. **for** each $Q_{i,j(t)}$in the sorted order **do**
4. **if** $[Q_{i,j(t)}+ G_{i,j(t)} - V.y_{i,j};].e_j >= V. \mu_j$ **then**
5. $x_{i,j(t)}= \min \{ \frac{Q_{i,j(t)}}{e_j}, S_j - x_{j(t)} \}$;
6. **end if**
7. allocate $x_{i,j(t)}$servers to queue$Q_{i,j(t)}$;
8. $x_{j(t)} = x_{j(t)} + x_{i,j(t)}$;
9. **end for**
10. $y_{j(t)} = x_{j(t)}$;
11. $y_{j,k(t)} = 1, \forall k \in \{ 1, \dots , y_{j(t)} \}$;
12. $y_{j,k(t)} = 0, \forall k \in \{ y_{j(t)} + 1, \dots . S_j \}$;

C. *Server Management Algorithm in Public Cloud*

For a DC *j* in the public cloud, it needs to decide the VM allocation variable $x_{i,j(t)}$for queue $Q_{i,j(t)}$ and VM state variable $y_{j,k(t)}$. Then, we obtain the optimization sub-problem for each public DC *j* Algorithm 2 explains the server management in a public cloud for DC *j*.

Lines 1 to 3 are for initialization in which line 2 updating the aux parameters $\{ y_{j,k(t)} \}$ to determine the states of the virtual machines. Similar to the algorithm 1, Lines 4 to 11 are allocating virtual machines to each queue. Whenever MCSP needs extra VMs from public cloud, the situation is handled in lines 12 to 18.

Algorithm 2: Server Management in Public DC j
Input:
$Q_{i,j(t)}$, $G_{i,j(t)}$ and $Y_{j,k(t-1)}$

1. $x_{i,j(t)} = x_{j(t)} = 0, \ \forall i$;
2. $Y_{j,k(t-1)} = \max \ \{Y_{j,k(t-1)} - 1, 0\}, \forall k$;
3. $Y_{j,k(t)} = [\ \frac{Y_{j,k(t)}}{T_g}], \forall k$;
4. Sort the queues $\{ Q_{i,j(t)}, \forall i \}$ in descending order of $Q_{i,j(t)}+ G_{i,j(t)} - V.y_{i,j}$;
5. **for** each $Q_{i,j(t)}$in the sorted order **do**
6. **if** $[Q_{i,j(t)}+ G_{i,j(t)} - V.y_{i,j};].e_j >= V. \mu_j$ **then**
7. $x_{i,j(t)}= \min \{ \frac{Q_{i,j(t)}}{e_j}, S_j - x_{j(t)} \}$;
8. **end if**
9. allocate $x_{i,j(t)}$servers to queue $Q_{i,j(t)}$;
10. $x_{j(t)} = x_{j(t)} + x_{i,j(t)}$;
11. **end for**
12. **while** $\Delta > 0$ do
13. k = k +1 ;
14. **if**$Y_{j,k(t)} = 0$ **then**
15. $Y_{j,k(t)} = 1$
16. $\Delta = \Delta - 1$;
17. **end if**
18. **end while**

D. Distributed Online Hybrid Cloud Management

Based on the above discussion we can find out the optimal solution for effective request handling and scheduling as well as efficient resource management in hybrid cloud. Here is the algorithm 3 for efficient service model for distributed multimedia cloud computing. This algorithm is covering the operations in both data center and proxies. Lines 1 to 5 are for managing the proxies where they determine the schemes for admitting requests and forwarding them to the DCs. Lines 6-17 are for the DCs. In this DC manages the servers/VMs from public and private cloud as per requirements. We drop a proper number of requests in each zone as shown in Line 14. Note that this operation would not change the queue-updating strategy described in (7).

Algorithm 3: Distributed Hybrid Cloud Management

1. **for** each zone $i\in$ Z **do**
2. proxy receives information on $\{Q_{i,j(t)}\}$ from DC
3. proxy obtains $\{a_{i,j(t)}\}$
4. proxy sends request to DCs according to $\{a_{i,j(t)}\}$
5. **end for**
6. **for** each DC $j\in$ Z **do**
7. Send information on $\{Q_{i,j(t)}\}$ to proxy in zone i
8. **if** DC j is a private DC **then**
9. manage servers in private cloud with Algorithm 1
10. **else**
11. manage VMS with distributed hybrid cloud With Algorithm 2
12. **end if**
13. Calculate $\{ b_{i,j(t)} \}$
14. drop min$\{\max\{ Q_{i,j(t)} - x_{i,j(t)}.e_j , 0\}, \{b_{i,j(t)}\}$ request in each queue $Q_{i,j(t)}$
15. update $Q_{i,j(t)}$
16. update $G_{i,j(t)}$
17. **end for**

IV. TEST RESULTS

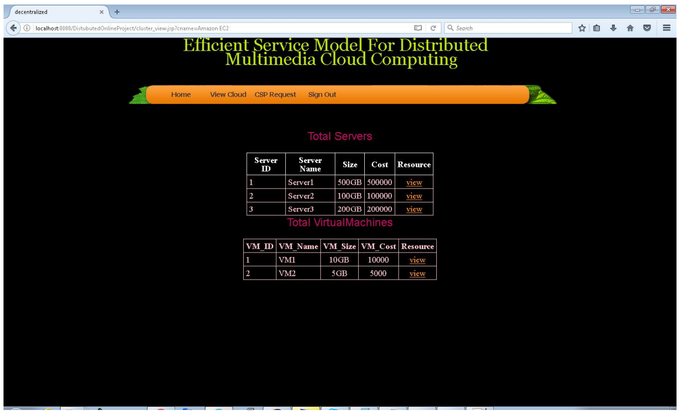

Fig 2: List of VMs and Servers

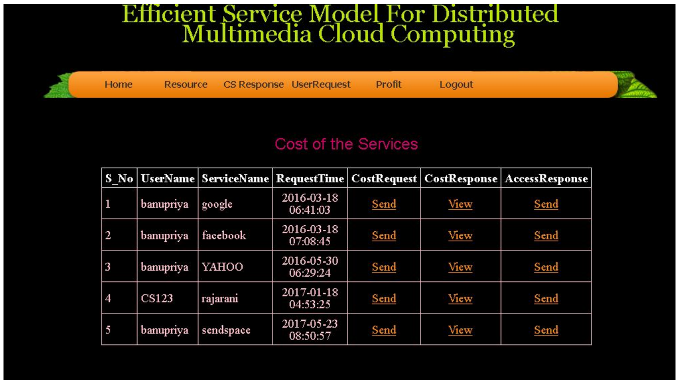

Fig 3: Cost of services

Fig 4: Available Services

sid	sname	slink	msize	validity	cname	server	scost
1	google	https://www.google.co.in	500 GB	>SixMonth	Amazon EC2	Server1	500000
2	facebook	www.facebook.com	500 GB	>SixMonth	Institution Cluster	Server2	500000
3	linkdin	www.linkedin.in	100 GB	>SixMonth	Institution Cluster	Server1	100000
4	gmail	https://www.gmail.com	500 MB	>SixMonth	Enterprise Cluster	Server1	500
5	YAHOO	WWW.YAHOO.CO.IN	500 GB	>SixMonth	Enterprise Cluster	Server1	500000
6	sendspace	http://www.sendspace.com/	100 GB	>SixMonth	Amazon EC2	Server2	100000
8	skive	www.skiveprojects.com	600 MB	>SixMonth	Public	Server1	600

Fig 5: Data Centre Details Data

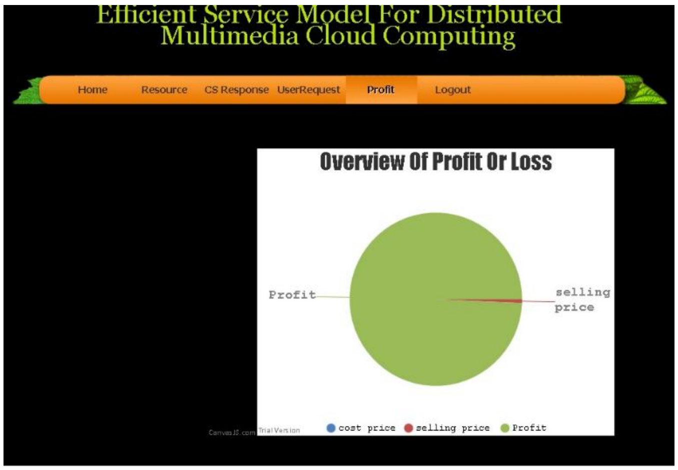

Fig 6: Profit

V. CONCLUSIONS

Using hybrid cloud we can extend the services of MCSP to the geographically distributed users. To extend the services of MCSP with quality, we have proposed a solution in which first we have explained on efficient service model for distributed multimedia cloud computing. This model is capable of handling the user request coming from geographically different locations and handle / route them efficiently. Aslo it is focusing on handling servers/VMs/resources efficiently in private as well as public cloud.

REFERENCES

[1] W. Zhu, C. Luo, J. Wang, and S. Li, “Multimedia cloud computing,” IEEE Signal. Process. Mag., vol. 28, pp. 59–69, May 2011.

[2] I. Bisio, R. Pan, F. Lavagetto, M. Marchese, A. Sciarrone, C. Fra, and M. Valla, “Smartphone-based automatic place recognition with Wi-Fi signals for location-aware services,” in Proc. ICC, Jun. 2012, pp. 4943–4948.

[3] I. Bisio, F. Lavagetto, M. Marchese, and A. Sciarrone, “Comparison of situation awareness algorithms for remote health monitoring with smartphones,” in Proc. GLOBECOM, Dec. 2014, pp. 2454–2459.

[4] M. Altamimi, R. Palit, K. Naik, and A. Nayak, “Energy-as-a-Service (EaaS): On the efficacy of multimedia cloud computing to save smartphone energy,” in Proc. CLOUD, Jun. 2012, pp. 764–771.

[5] H. Hu, Y. Wen, T. Chua, Z. Wang, J. Huang, W. Zhu, and D. Wu, “Community based effective social video contents placement in cloud centric CDN network,” in Proc. ICME, Jul. 2014, pp. 1–6.

[6] M. Neely, Stochastic etwork Optimization With Application to Communication and Queueing Systems. San Rafael, CA, USA: Morgan and Claypool, 2010.

[7] X. Nan, Y. He, and L. Guan, “Optimal resource allocation for multimedia cloud in priority service scheme,” in Proc. ISCAS, May 2012, pp. 1111–1114.

[8] X. Nan, Y. He, and L. Guan, “Optimization of workload scheduling for multimedia cloud computing,” in Proc. ISCAS, May 2013, pp. 2872–2875.