Optimization of Bandwidth using Load Balancing Algorithm in Data Centers

[1]T. Subashini [2]T. Mahesh Bhoopathy [3]G. P. Arvind Kumar [4]V. Ganesh Kumar
[1]Assistant Professor [2,3,4]Student
[1,2,3,4]Department of Computer Science and Engineering
[1,2,3,4]Velammal Engineering College, Chennai

Abstract

Data-center network (DCN) plays a fundamental role in cloud computing. As applications grow fastly, capacity utilization of data-center network becomes a big challenge to cloud service. In this paper, we present Genetic load balancing algorithm to balance the loads and controller to enable the central control of the entire network, and propose an optimization model to consider high bandwidth utilization for provider and low delay for users. To the best of our knowledge, existing approaches, however, either solely focus on minimizing total cost for provider, or guaranteeing QoS for end-users.
Keyword- Data Center Network, Genetic Algorithm, Load Balancing Algorithm

---

I. INTRODUCTION

Concepts of virtualization are used in cloud which leads to have a load balancing in the cloud. Virtualization means giving a logical name to physical resources and whenever this name is referred it will point towards corresponding physical resource. Multiple users will access cloud at same time and it is very necessary to serve them all with minimum response time and better service. For this reason load balancing is taken in to effect to balance the request of multiple users on virtual machines evenly.

It is said that Load balancing is a NP-Complete Problem method because as the size of the problem increases the size of solution will increase too.

Genetic Algorithm (GA) is much popular for solving NP-Complete problems. GA is one of technique which belongs to the class of evolutionary algorithms which generates solutions inspired by natural evolution. Effective load balancers intelligently determine which device within a given server farm is best able to process an incoming data packet. Doing so requires algorithms programmed to distribute loads in a specific way.

The main challenge to network layer load balancers is a lack of visibility into traffic flow, limited to information stored in network packet headers.

II. EXISTING SYSTEM

Recently, request allocation problem has gained considerable research interest over the past few years. However, existing solutions solely focus on the benefit of either provider or end-users. We review first solutions considering the benefit of end-users adopt a general fairness criterion based on Nash bargaining solutions, and present a general optimization framework that models the realistic environment and practical constraints that a cloud faces. Proposed to optimize traffic engineering across all upstream ISPs, assuming requests are simply allocated to the closest ingress point. Supported locality policies based on on-demand network probing.

A. Drawbacks

- This user assignment algorithm does not attempt to minimize the distance between users and the datacenters.
- Leads to poor performance for users far away from the infrastructure.
- Their work mainly targets in a single datacenter.

III. PROPOSED SYSTEM

We focus on the emerging request allocation problem in geographically distributed datacenters, and propose a joint optimization model to consider high bandwidth utilization for provider and low delay for end-users. We formulate the request allocation under those requirements as an optimization problem. Such optimization can be NP-hard. To solve it, we propose an efficient request

allocation algorithm by introducing the auxiliary variable method to eliminate inequality, rather than directly applying the Logarithmic Smoothing technique.

We perform theoretical analysis to prove the existence and uniqueness of our solution, and the convergence of our algorithm. With the simulation results, we show that our algorithm outperforms the conventional greedy and locality algorithms, and can efficiently improve the bandwidth utilization for provider and reduce the delay for end-users.

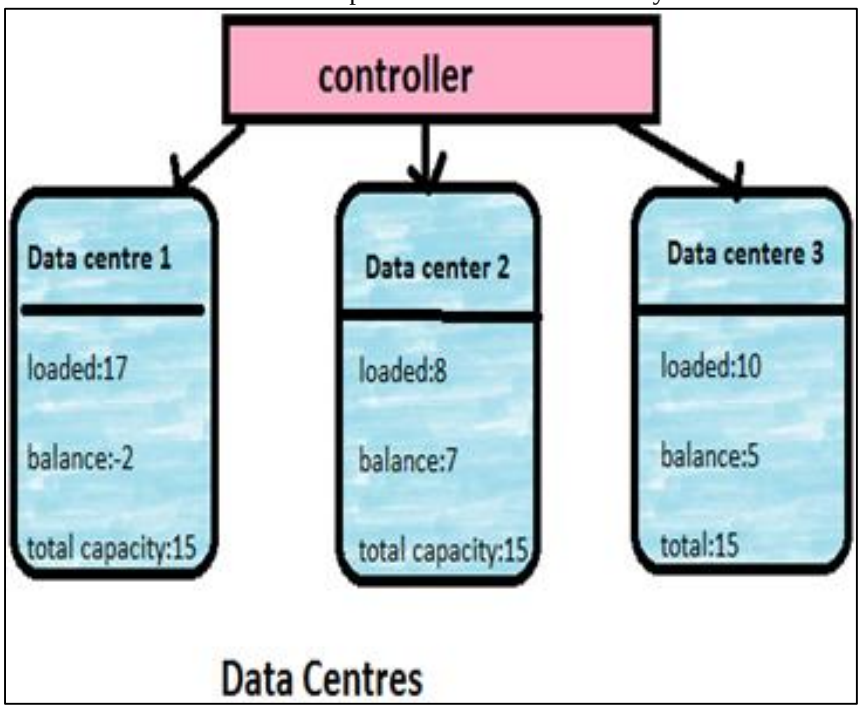


Fig. 1: Data Centers

A. Advantages

- Shifting workload from overloaded datacenters with low utilization.
- Their Delay doesn't contain the response time inside data centers.
- Our algorithm can efficiently improve the bandwidth utilization for the provider and reduce the delay for users, compared with both greedy and locality algorithms.

B. Architecture

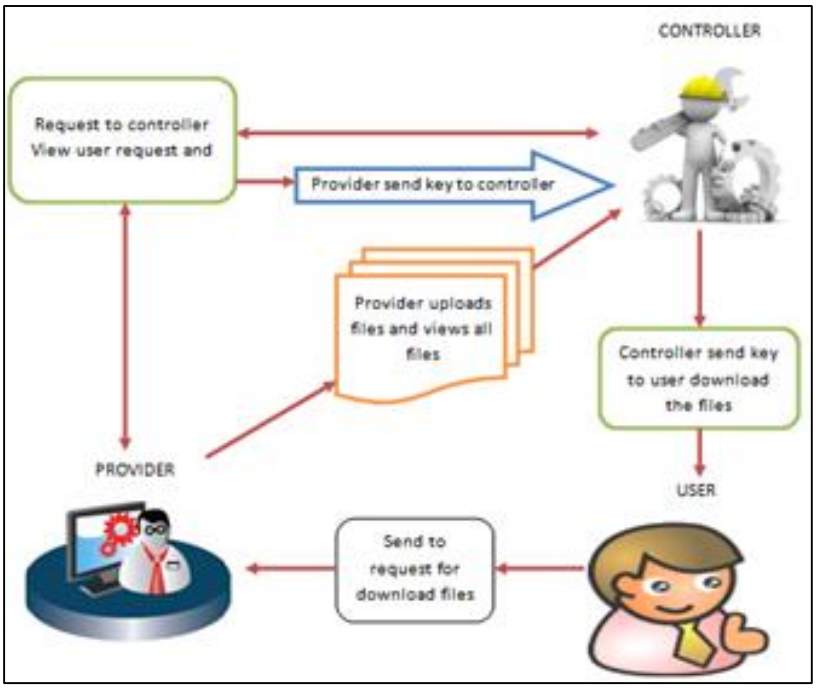


Fig. 2: Architecture

C. Modules

This Project has the following modules,

- Upload & Storage
- Authentication of User Request
- Key request processing
- Controller Maintenance

D. Hardware and Software Specification

1) Hardware Requirements

System - Pentium IV 2.4 GHz.
Hard Disk - 40 GB.
Floppy Drive - 1.44 Mb.
Monitor - 15 VGA Color.
Ram - 512 MB

2) Software Requirements

Operating system - Windows.
Coding Language - Java, Swing, JSP.
Front End Tool - Net beans 7.0
Database - MS Sql.
Back End Tool - SQL Yog.

E. Figures

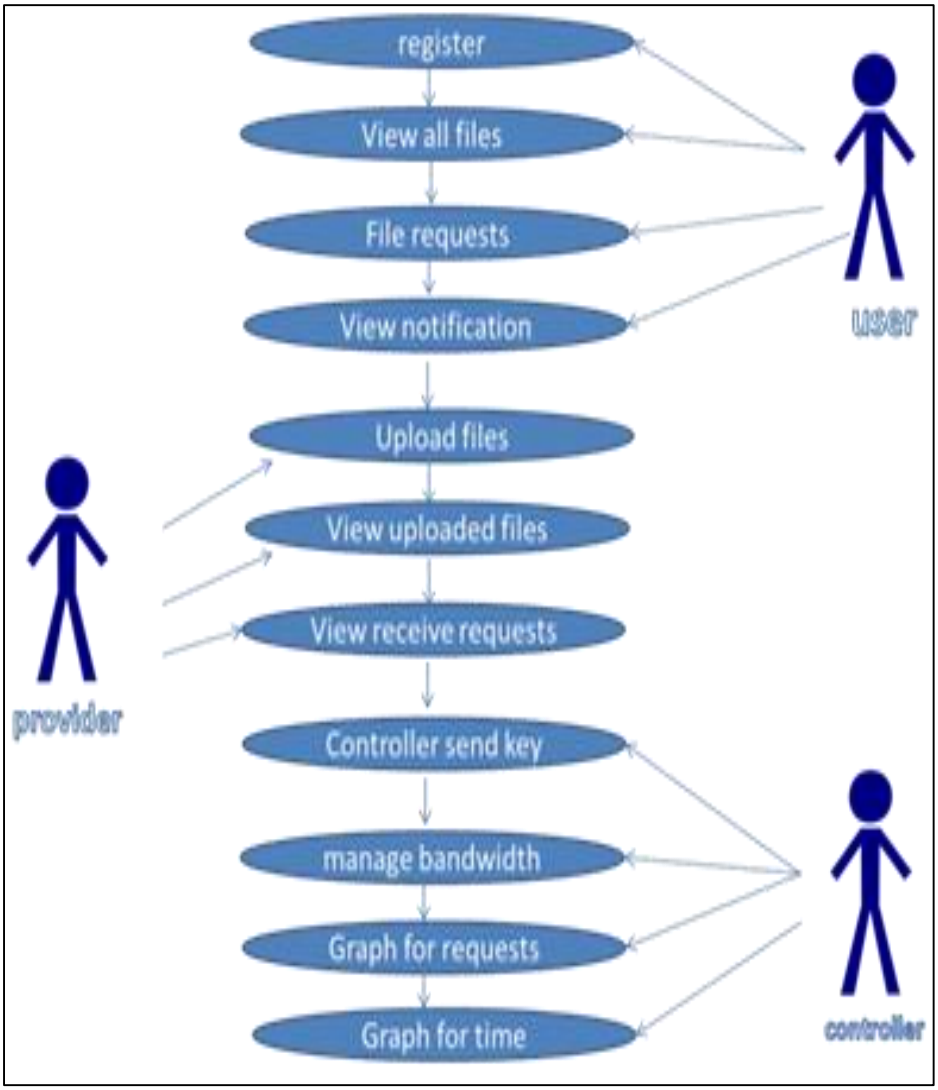


Fig. 3: Use Case Diagram

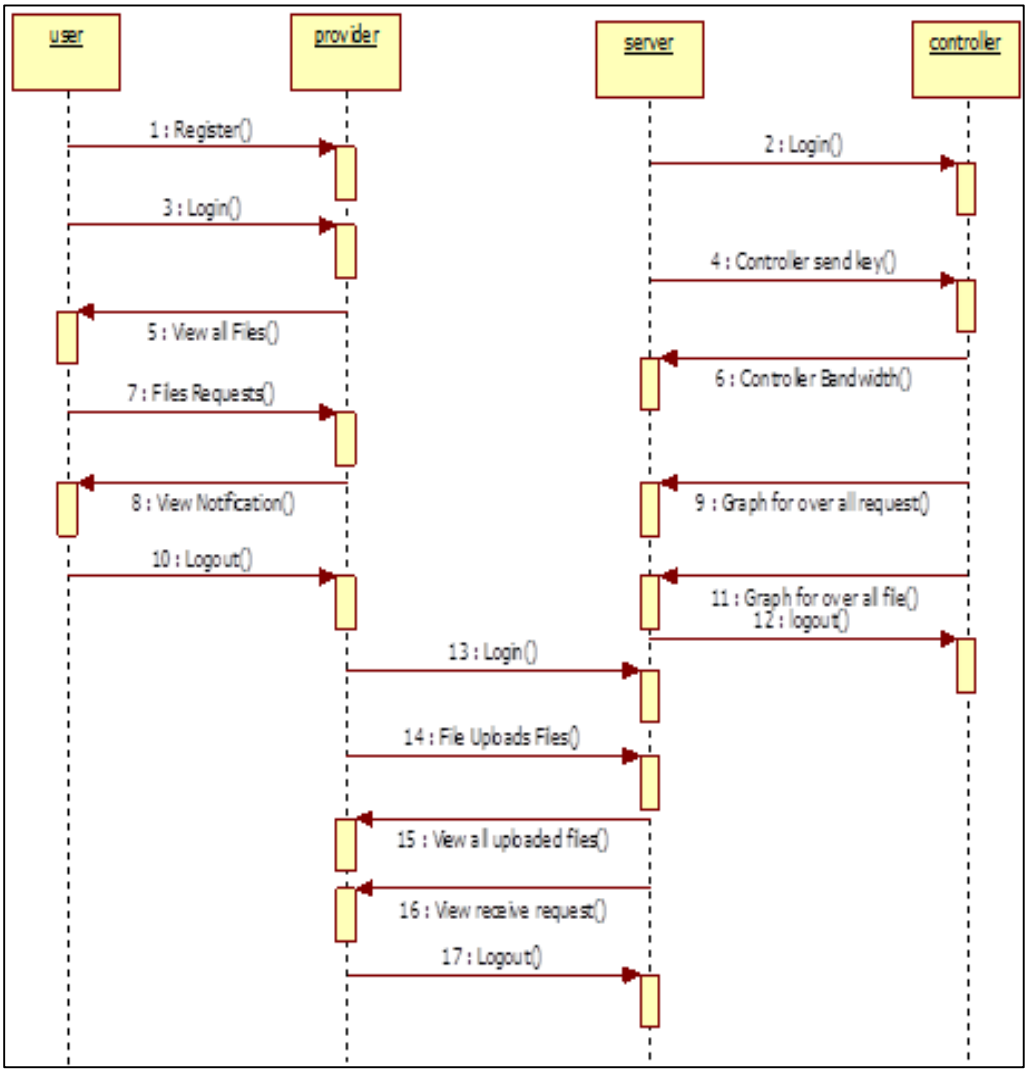


Fig. 4: Sequence Diagram

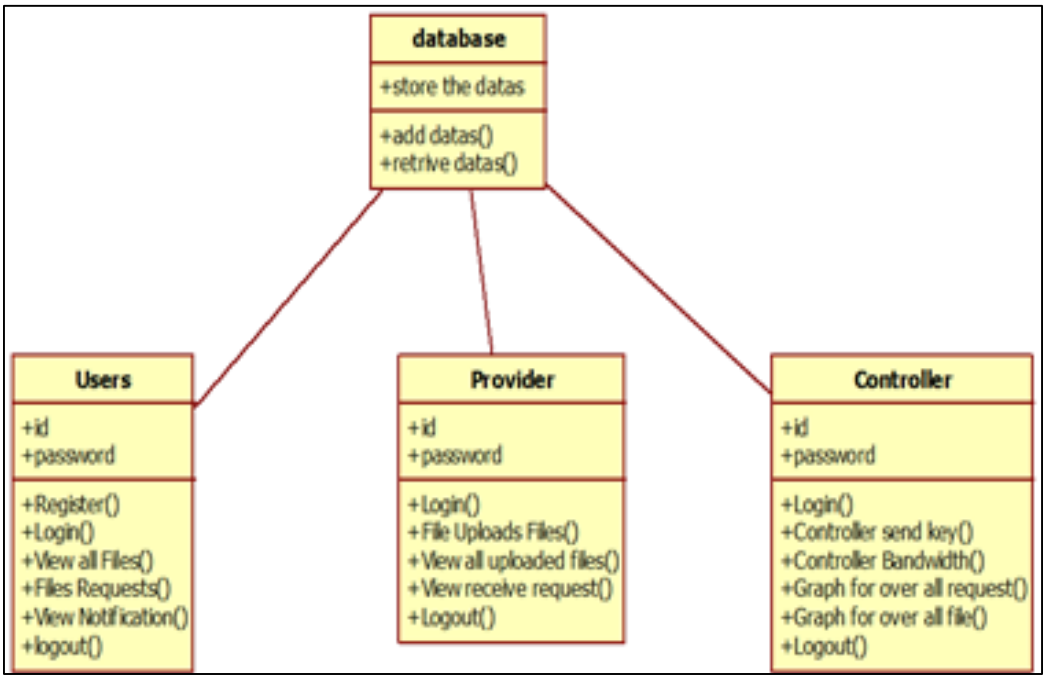
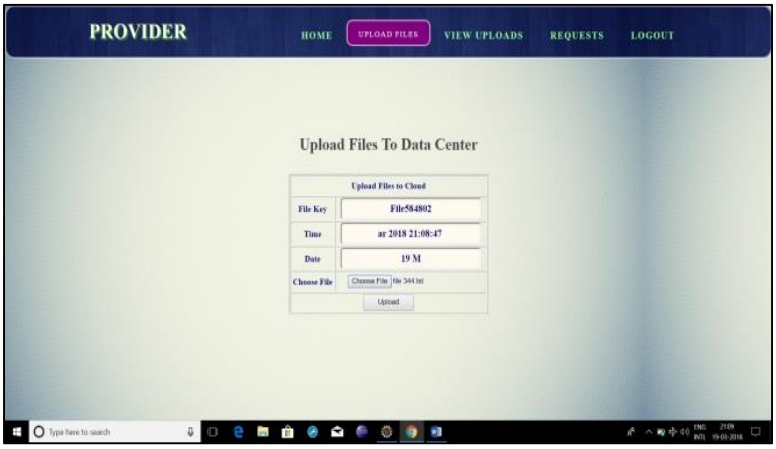
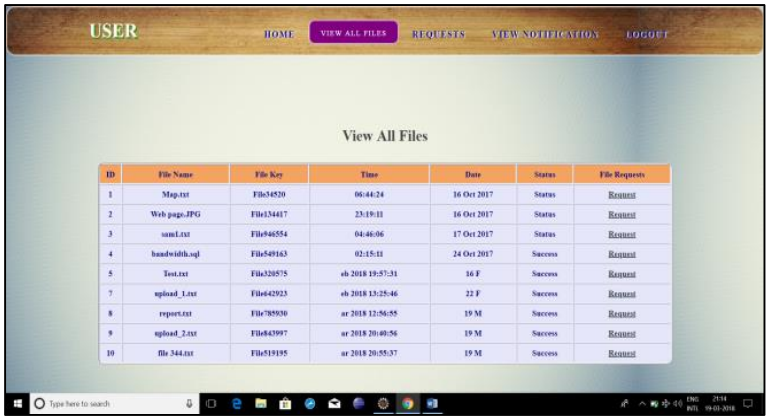
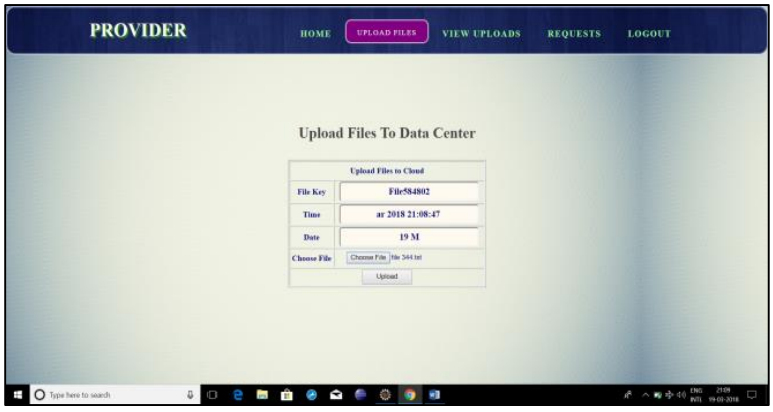


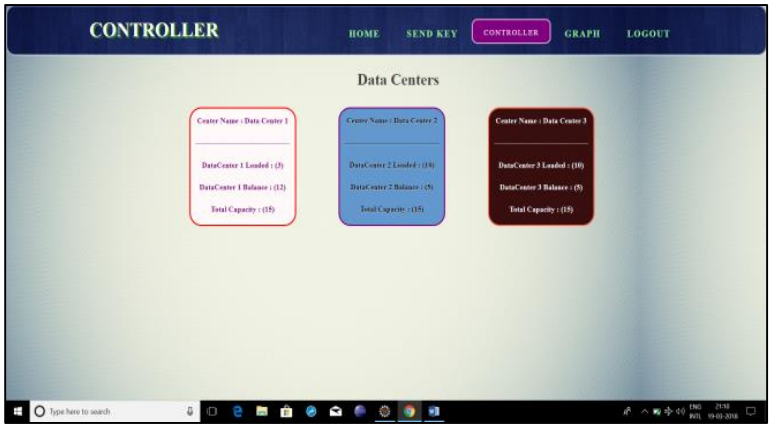
Fig. 5: Class Diagram

F. Results









Algorithm	*Environment*	*Parameters*
Min - Min	*Static*	*Response time*
Max - Min	*Static*	*Waiting time*
Round Robin	*Static*	*Waiting time*
Genetic Algorithm	*Dynamic*	*Process utilization*
Ant Colony	*Dynamic*	*Throughput*
Opportunistic Load Balancing	*static*	*Waiting time*
Game Theory	*Dynamic*	*Reliability*
Stochastic Hill Climbing	*Dynamic*	*Response time*
Firefly	*Dynamic*	*Resource utilization*

Table 1: Result Comparisons

IV. FUTURE WORK

We plan to more thoroughly study the decentralized implementation of request allocation by deploying one controller in each data centre. Future work can proceed in couple of directions. We are interested in looking into other video streaming delivery systems such as Hulu, to see if cloud sourcing and/or multiple CDN strategy have been adopted.

REFERENCES

[1] Amazon web services [Online]. Available: http://aws.amazon.com, 2014.
[2] A. Cockcroft. (2011). Netflix in the cloud [Online]. Available: http://velocityconf.com/velocity 2011/public/schedule/detail/ 17785
[3] H. Ballani, P. Costa, T. Karagiannis, and A. Rowstron, "Towards predictable datacentre networks," in Proc. ACM SIGCOMM Conf., Toronto, ON, Canada, 2011, pp. 242–253.
[4] N. Laoutaris, M. Sirivianos, X. Yang, and P. Rodriguez, "Interdatacenter bulk transfers with netstitcher," in Proc. ACM SIGCOMM Conf., Toronto, ON, Canada, 2011, pp. 74–85.
[5] R. Buyya, R. Ranjan, and R. N. Calheiros, "Intercloud: Utilityoriented federation of cloud computing environments for scaling of application services," in Proc. 10th Int. Conf. Algorithms Archit. Parallel Process., 2010, pp. 13–31.