Improved Load Balancing in Cloud Computing Environments

Dr. Narayan A. Joshi
Dharmsinh Desai University
(E-mail: joshinarayan@gmail.com)

Abstract—Recent enhancements in storage, computing and communication technologies have triggered several improvements in cloud computing technologies. Cloud computing technologies have gained a lot with help of improvements in security aspects and increased transparency in service level agreements. Efficient management of cloud computing environments depends on efficient utilization of cloud based resources and smooth execution of service level agreements. Ample research work on load balancing in cloud computing is going on. A novel load balancing technique in cloud computing environment is presented in this paper.

Keywords—*cloud computing, hybrid cloud, collaborated cloud, load balancing, task migration*

I. INTRODUCTION

Technological advancements in cloud computing technology and communication technology has stimulated adoption of cloud computing in the Information Technology (IT) Sector. Various characteristics such as zero maintenance, continuous availability, and least cost are few driving forces behind wide acceptance of cloud computing in various IT and non-IT sectors. Reduced IT solution providers have already started delivering cloud based solutions for various problem domains. Moreover, several Internet of Things (IoT) based computing solutions also nowadays bank upon cloud computing for not only data storage but also computation.

On other side, increase in demand for cloud based IT solutions, has resulted into increase in requirement for cloud based resources such as storage and compute. Managing such increase in demand for cloud based resources is becoming challenging for the cloud service providers day by day. Due to uncertainty in various service request parameters such as – time of service utilization, service request arrival time, quantity of required resources, and time duration of the resource requirement – the puzzle of efficient resource management becomes very difficult for the resource managers. Moreover, the inappropriate resource management may result into violation in Service Level Agreement (SLA).

Various load balancing techniques exist for optimized resource management. One of the load balancing technique in cloud computing environments is migration of virtual machines over physical machines. On other side, few load balancing techniques rely on redirection of incoming requests to different virtual machines other than the overloaded virtual machine which is actually required to handle the incoming request.

Though such techniques are suitable for intra cloud load balancing, there is serious requirement of inter cloud load sharing techniques. Nowadays, cloud service providers have started joining hands by collaborating their respective cloud computing environments for the sake of optimized resource utilization, improved cost saving, reduced energy consumption and higher availability. Such collaborated clouds help service providers to withstand SLAs in better way and thereby generate more profit. A load balancing technique for collaborated cloud computing environments has been presented in this paper.

Related survey is presented in Section II. Suggested mechanism is presented in section III. Resultant outcomes are described in section IV.

II. RELATED WORK

Load balancing plays vital role in harmonizing financial benefits and user contentment. The cloud computing environments incorporate several physical, logical and architectural components. The physical infrastructure consists of data centers having high-end compute and storage capabilities. Moreover, the collaborated cloud environments are connected by means of routers and switches. On other side, such huge physical infrastructure for continuous functioning and availability, requires enormous amount of uninterrupted energy supply.

In a work suggested by Azar et al., resource management is based on f-restricted algorithm. The technique incorporates first fit based algorithm [1]. A load balancing technique based on weighted signature is suggested by Ajit et al. works on reducing response time. The technique collects load assignment value for the available machines and accordingly associates virtual machines to the machines [2]. The honeybee theory based load balancing technique suggested by Vasudevan et al. assigns the available resources to cloud network for reducing the service makespan [10].

A load balancing technique suggested by Zhu et al. works on improved Particle Swarm Optimization in cloud computing environment [11]. A cloud resource management technique suggested by Kapur is based on Load Balanced Resource Scheduling for reserved and on-demand resources [7]. A Lyapunov optimized theory based technique is suggested by Deng et al. [4]. They have proposed an algorithm EcoPower which implements optimized power management and resource management. A load balancing technique suggested by Joshi et al. works on shifting of desired tasks from highly loaded machine to lightly loaded machine [6]. The technique works within a solitary cloud computing environment.

A load balancing algorithm is proposed by Chien et al. enhances performance on the basis of required processing time [3]. For optimization purpose, the work is based on job completion time. A technique suggested by Papadopoulos is based on control-based self-adaptive randomized optimization [9]. The technique balances workload of cloud servers. An improved virtual machine placement technique is suggested by Nan S. et al. [8]. The technique is based on an algorithm meant for improving annealing simulations. The suggested technique works on resource utilization, dynamic placement and optimized annealing.

Based on the literature study, it is felt that some techniques are limited to certain computing environments. On other side, there exist techniques for migrating whole virtual machines across the network and servers. Some of the techniques work on static system aspects. Whereas some of the load balancing solutions are closed source. Some of the techniques are constrained to closed source cloud computing environments. Few of the existing techniques do require prior knowledge of request's service time. Few of the available techniques are based on simulation work. An open source load balancing solution for cloud computing environments is suggested in this paper.

III. MECHANISM

The mechanism described here is improvement on the work suggested in [5] by letting virtual machines freedom to set workload thresholds themselves. Moreover, the mechanism makes sure that the status of a particular virtual machine should not get altered during the intermediate time of VM selection and actual load balancing. In such a case, the mechanism ignores concerned overloaded or under loaded virtual machines and re-identifies appropriate virtual machine of desired type.

Both the threads load balancer and state manager work parallel to each other. VM's workload can be either current one or the average of last three readings. Virtual machines which are either not available or not willing to participate in load balancing have been treated as INACTIVE. The algorithm works with IPv6 addresses.

```
Module LoadBalancer
{
   enum vm_stat
   {INACTIVE=0, ACTIVE=1, PASSIVE=2};
   int Ts; // snooze time interval
   struct VMInfo
   // VM's current information
   {
      float load, TOver, TUnder;
      // workload & threshold values
      unsigned short cores;
      unsigned long max_mem, free_mem;
      long bandwidth;
      vm_stat state;
      // current VM state
      char vmIP[40];
      unsigned long passive_time;
      …
   };
VMInfo vms[n];
// there are n virtual machines
// setup vms virtual machines array
// Initialize vm_stat for-
// participating virtual machines
void threadLoadBalance()
{
   struct VMInfo* pVMU;//underloaded VM
   struct VMInfo* pVMO;//overloaded VM
   Ts = 5; // snooze for 5 units
   while(true)
   {
      UNDERLOADED_VM:
      pVMU=null;
      pVMU=determine_underloaded_VM();
      if (!pVMU || pVMU->load > TUnder)
      {
         sleep(Ts);
         goto OVERLOADED_VM;
      }
      OVERLOADED_VM:
      pVMO = null;
      pVMO = determine_overloaded_VM();
      if (!pVMO || pVMO->load < TOver)
      {
         sleep(Ts);
         goto OVERLOADED_VM;
      }
      if(VMInfo[pVMO] <= VMInfo[pVMU]&&
         pVMO->load > TOver &&
         pVMU->load < TOver)
```

```
            {
                  pVMU->state = PASSIVE;
                  // set pVMU->passive_time to
                  // to the load-balancing time
                  balance(pVMO, pVMU);
                  …

            }

      }
      SNOOZE:
            sleep(Ts/2);
      };
      void threadManageState()
      {
            // sync with load_balance()
            pVM->state = ACTIVE;
            …
      }
};
```

IV. OUTCOMES

The algorithm was implemented on testbed open source cloud environment with virtual machines having deployed open source operating systems on them. Testbed cloud computing environment's private network is connected to the outside public network and clients by means of a router.

Before load balancing took place in the test bed cloud environment, it was observed that the most overloaded virtual machine (VMO) was running with very high 94% CPU work load. On other side, the most under loaded virtual machine (VMU) was nearly idle running with 11% work load.

The suggested mechanism was applied on these overloaded and under loaded virtual machines. Load balancer decided shifting of work load from VMO to VMU. After load balancing took place, the workload came down to 68% on the previously overloaded virtual machine VMO. On other side, after load balancing towards the previously under loaded virtual machine VMU, the new work load was 36%.

Likewise more load balancing results are listed in following Table-1.

V. CONCLUDING REMARKS

A dynamic load balancing algorithm in cloud computing environments has been presented in this paper. While balancing workload for overloaded virtual machines, the algorithm also takes into consideration the workload thresholds of individual virtual machines. The algorithm allows virtual machines to decide the workload threshold values. Moreover, the mechanism prevents previously under loaded machines from instantaneous over loading.

REFERENCES

[1] Azar, Y., Cohen, I.R., Fiat, A., Roytman, A., "Packing small vectors", Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, January 2016

[2] Ajit, M., Vidya, G., "VM level load balancing in cloud environment", IEEE Fourth International Conference on Computing, Communications and Networking Technologies, July 2013

[3] Chien, N. K., Son, N. H., Loc, H. D., "Load balancing algorithm based on estimating finish time of services in cloud computing", 18th IEEE International Conference on Advanced Communication Technology, January, 2016

[4] Deng, Y., Lau, R.W., "Dynamic load balancing in distributed virtual environments using heat diffusion", ACM Transactions on Multimedia Computing, Communications, and Applications, 10 (2), 2014

[5] Joshi N., "Load Balancing in Cloud Computing Using Process Migration", International Journal of Advanced Research in Engineering and Technology, April, 2014

[6] Joshi N., Choksi D. B., Kotecha K., Pandya S., "Implementation of Novel Load Balancing Technique in Cloud Computing Environment", International Conference on Computer Communication and Informatics, January, 2018

[7] Kapur, R., "A workload balanced approach for resource scheduling in cloud computing", Eighth IEEE International Conference on Contemporary Computing, August, 2015

[8] Nan S., Shi. A., Chen C., Chen E. and Wang Y., "Research on Virtual Machine Placement in the Cloud Based on Improved Simulated Annealing Algorithm", World Automation Congress, 2016

[9] Papadopoulos, A. V., Klein, C., Maggio, M., Drango, J., Dellkrantz, M., Hernndez, F., Arzn, K. E., Control-based load-balancing techniques: Analysis and performance evaluation via a randomized optimization approach. Control Engineering Practice, 52, 2016

[10] Vasudevan, S. K., Anandaram, S., Menon, A. J., Aravinth, A., "A novel improved honey bee based load balancing technique in cloud computing environment", Asian Journal of Information Technology, 15 (9), 2016

[11] Zhu, Y., Zhao, D., Wang, W., He, H., "A Novel Load Balancing Algorithm Based on Improved Particle Swarm Optimization in Cloud Computing Environment". International Conference on Human Centered Computing, Springer, January, 2016

ORE & AFTER APPLICATION OF LOAD BALANCING

Sr. No.	Workload Before load balancing		Workload After load balancing	
	Source VM	Destination VM	Source VM	Destination VM
1	192.168.10.7 : 94%	192.168.20.1 : 11%	192.168.10.7 : 68%	192.168.20.1 : 36%
2	192.168.20.4 : 88%	192.168.10.8 : 19%	192.168.20.4 : 62%	192.168.10.8 : 40%
3	192.168.10.3 : 91%	192.168.10.1 : 37%	192.168.10.3 : 68%	192.168.10.1 : 57%
4	192.168.10.5 : 82%	192.168.20.9 : 15%	192.168.10.5 : 67%	192.168.20.9 : 38%
5	192.168.20.2 : 93%	192.168.10.9 : 38%	192.168.20.2 : 68%	192.168.10.9 : 68%