International Journal of Latest Trends in Engineering and Technology
Special Issue SACAIM 2017, pp. 453-457
e-ISSN:2278-621X

A STUDY ON RESOURCE ALLOCATION IN CLOUD

FreedaD'costa[1], Rajatha K.J[2] & SREENIVAS B.L[3] & Dr. S Sathyanarayana[4]

Abstract- Cloud computing ia a type of web based computing that relies on sharing of computer resource from any place and whenever. It is similar to utility computing.Cloud comprises of a datacenter which in turn comprises of a few physical machines. Each machine is shared by numerous clients and virtual machines are utilized to use these physical machines. With a large number of data centers and each data center having a large number of physical machines,the VM allocation turns into a difficult issue.In this paper we do a study on various policies for dynamic resource allocation in cloud based on Topology Aware Resource Allocation, Linear Scheduling Strategy for resource allocation and Dynamic Resource Allocation for Parallel Information Processing, Genetic algorithm.
Keywords: Dynamic Resource Allocation, Cloud Computing, Resource Management, Resource Scheduling

1. INTRODUCTION

The cloud computing technology makes the resource as a solitary purpose of access to the customer and is implemented as pay per use. In spite of the fact that there are different advantages in cloud computing, for example, prescribed and abstracted framework, totally virtualized condition, furnished with dynamic framework, pay per utilization, free of software and hardware installations, the major concern is the order in which the requests are fulfilled. This advances the scheduling of the resources. This allocation of resources must be made productively that boosts the system utilization and overall performance. Cloud computing is sold on request on the basis of time constraints fundamentally indicated in minutes or hours. In this way scheduling should be made such that the resource ought to be used productively.

In cloud platforms, resource allocation (or load balancing) happens at two levels. Initially, when an application is uploaded to the cloud, the load balancer assigns the requested examples to physical computers, attempting to balance the computational load of different applications across physical computers. Second, when an application gets different incoming requests, these requests should be each assigned to a particular application instance to balance the computational load across a set of instances of the same application. For instance, Amazon EC2 utilizes elastic load balancing (ELB) to control how incoming requests are taken care of. Application designers can coordinate requests to instances in particular accessibility zones, to particular instances, or to instances showing the shortest response times.

The rest of the paper is organized as follows: Resource allocation strategies and algorithms are explained in section II. Conclusion and future work are given in section III.

2. RESOURCE ALLOCATION STRATEGIES &ALGORITHMS

According to this study, numerous resource allocation plans have come up in the writing of cloud computing as this innovation has begun improving. Scientists all around have proposed and/or implemented different sorts of resorce allocation mechanisms. Maybe a couple of the mechanisms for resource allocation in cloud computing are briefly explained below.

2.1 Topology Aware Resource Allocation (TARA)–

Different mechanisms for resource allocation are proposed in cloud computing. The one mentioned gives the architecture for optimized resource allocation in Infrastructure-as-a-Service (IaaS) based cloud systems. Present IaaS systems are normally unaware of the hosted[2] application's requirements and thus allocate resources independently of its requirements,which can significantly affect the performance for distributed data-based applications.To tell about this resource allocation problem,an architecture that adopts a —'what if' methodology is used to show allocation.The results proved that TARA minimised the task completion time of those applications by up to 59% of that when compared with application independent allocation strategies.

1.Architecture of TARA**:** TARA consists of two essential components,i.ea prediction engine and a fast genetic algorithm-based search technique.[2] The prediction engine is responsible for optimizing resource allocation. The moment it receives a resource request, the prediction engine searches through the possible subsets of available resources (each distinct subset is

[1] Department of Software Technology, Mangalore, St. Aloysius Institute of Management and Information Technology, Beeri, Mangaluru 575022, Karnataka, India
[2] Department of Software Technology, Mangalore, St. Aloysius Institute of Management and Information Technology, Beeri, Mangaluru 575022, Karnataka, India
[3] Assistant Professor, Department of Information Technology, St.Aloysius Institute of Management and Information Technology, Beeri,Mangaluru 575022, Karnataka, India
[4] Asst Professor, First Grade Women's College, Mysore, Karnataka

generally referred to as a candidate) and finds out an allocation that optimizes calculated job/task completion time.However, even with a lightweight prediction engine,exhaustively searching through all possible candidates is infeasible due to the range of IaaS systems. Thus a genetic algorithm-based search technique that allows TARA to guide the prediction engine through the search space efficiently is used.

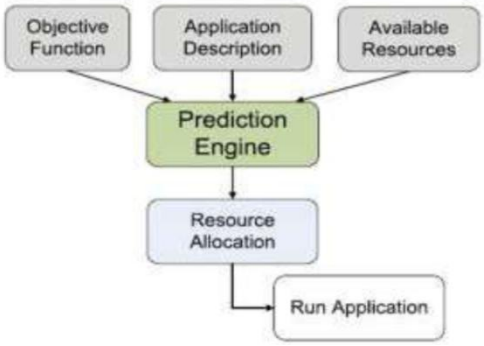

Fig No:A.1 Architecture of TARA[2]

2) Prediction Engine**:** The prediction engine does the mapping of resource allocation candidates to scores that rates their certainness with respect to a specified objective function, so that TARA can compare and rank the different candidates. The inputs which are used in the scoring process can be seen in Architecture of TARA.
3) Objective Function**:** The objective function states the metric that TARA should optimize. For instance, given the raising cost and lack of power in the data center, an objective function might measure the raise in power consumption due to a particular allocation.
4) Application Description:The application description involves three parts: 1)the framework type which recognizes the framework model to use, 2) work load specific parameters which describe a particular application's resource usage and 3) a request for resources including the number of VMs, storage, etc.
5) Available Resources**:** The final input needed by the prediction engine is a resource picture of the IaaS data centre. This includes information derived from both the virtualization layer and the IaaS monitoring service.Theinformation collected ranges from a list of available servers, current load and available capacity on individual servers to data centre topology and a recent measurement of available bandwidth on each network link.

2.2 Linear Scheduling Strategy for Resource Allocation-

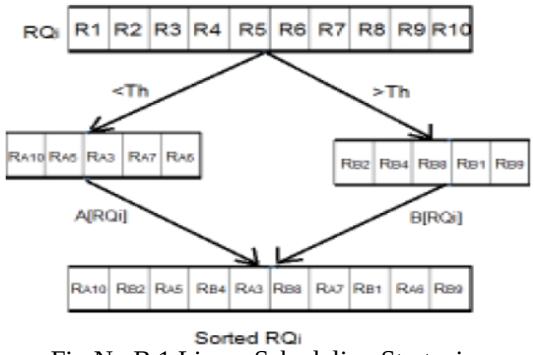

Fig No:B.1 Linear Scheduling Strategies

The cloud environment with the service node to control all clients request can provide maximum service to all clients if they consider the processing time, resource utilization based on CPU usage, memory usage and Throughput. Allocating the resource and jobs separately involves more waiting time and response time.Thus a scheduling algorithm namely Linear Scheduling for Tasks and Resources (LSTR) is proposed, that performs tasks and resources allocating respectively. Here, a server node is required to establish the IaaS cloud environment and KVM/Xen virtualization along with LSTR scheduling to allocateresources which will increase the system throughput and resource utilization.Resource consumption and resource allocation have to be integrated so as to improve the resource utilization. The algorithms mostly focus on the distribution of the resources among the requestors that will maximize the selected QoS parameters. The QoS parameter selected in our study is the cost function. This scheduling algorithm was designed considering the different tasks and the available virtual machines and thus was named as LSTR scheduling strategy. This algorithm was designed to maximize the resource utilization.
Algorithm [3]:

1) The requests are collected between every predetermined interval of time
2) Resources Ri => {R1,R2, R3,...,Rn}
3) Requests RQi => {RQ1, RQ2, RQ3,...,RQn}
4) Calculate Threshold (static at initial)
5) Th = Σ Ri
6) for every unsorted array A and B
7) sort A and B
8) for every RQi
9) ifRQi<Th then
10) addRQiin low array, A[RQi]
11) else if RQi>Th then
12) addRQiin high array B[RQi]
13) for every B[RQi]
14) allocate resource for RQiof B
15) Ri= Ri- RQi; Th = Σ Ri
16) satisfy the resource of A[RQi]
17) for every A[RQi]
18) allocate resource for RQiof A
19) Ri= Ri- RQi; Th = Σ Ri
20) satisfy the resource of B[RQi]

2.3 Dynamic Resource Allocation for Parallel Data Processing-

Dynamic Resource Allocation for Efficient Parallel data processing [4] introduces a new processing framework explicitly that is designed for cloud environments called Nephele.Nephele is the first data processing framework to include the possibility of dynamically allocating/de-allocating different compute resources from a cloud in its scheduling and during job execution. Particular tasks of a processing job can be assigned to different types of virtual machines which are automatically instantiated and terminated during the job execution.

1.Architecture:

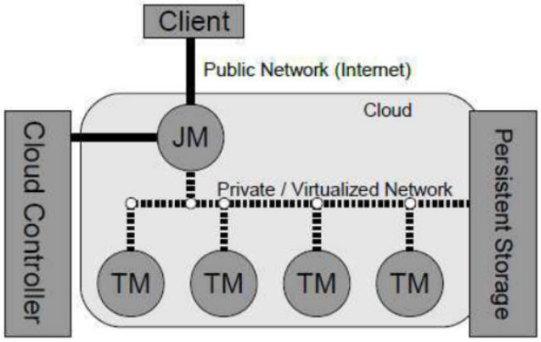

Fig No:C.1 Nephele Architecture[4]

Nephele's architecture [4] follows a classic master-worker pattern as shown in the figure. Before submitting a Nephele compute job, a user must start a VM in the cloud which runs the so called Job Manager (JM). The Job Manager receives the client's jobs, is responsible for scheduling them, and coordinates their execution. It is capable of communicating with the interface the cloud operator provides to control the instantiation of VMs. This interface is called the Cloud Controller. By the Cloud Controller the Job Manager can allocate or de-allocate VMs according to the current job execution phase.The actual execution of tasks which a Nephele job consists of is carried out by a set of instances. Each instance runs a so called Task Manager (TM). A Task Manager receives one or more tasks from the Job Manager at a time, executes them, and after that it informs the Job Manager about their completion or possible errors.

2.4 Genetic Algorithm

Genetic algorithm is not dependent on the auxiliary data of searching space. It just depends upon the fitness capacity to evaluate individuals, in this manner it gives a system for solving complex issues and is being broadly utilized as a part of different fields at present, for example: function optimization, combination optimization,jobscheduing and so on. The benefits of the algorithm are that it starts to look from the population, generally covering and supporting to get the globally ideal solution. Its disadvantages are uncommon programming realization, weaker local researching,longer searching time and globally ideal solutions are impacted by operator parameters.[16]

1. Selection Operation

Chromosomes are chosen from the population to be parents to crossover. The issue is how to choose these chromosomes. There are numerous methods how to choose the best chromosomes, for instance roulette wheel selection, Boltzmann selection,tournament selection, rank selection, steady state selection and some others.

2. Encoding Scheme

Genetic Algorithm has various numbers of chromosomes and every chromosomes has different gens that can be represented as V that indicates to the virtual Machine. P denoted the physical Machine on which VMs are to be allocated. P chromosome in this GA consists of |V | genes, each of which remains for a virtual machine. The value of a gene is a positive whole number between 1 and |P|,representing the physical machine where the virtual machine is distributed. Fig. 2 shows a example VM placement and its corresponding chromosome.

Fig 1: Figure representing chromosome for physical machines

3. Crossover Operation

The thought behind crossover is that the new chromosome might be better than both of the parents if it takes the test characteristics from each of the parents. In this more than one parent is chosen and at least one off-spring created using the genetic material of the parents. crossover is generally applied in a GA with a high probability. There are many crossover administrator example one point, two point, multi point,arithmetic.

I.One Point Crossover:In this one-point crossover, an random crossover point is chosen and the tails of its two parents are swapped to get new off-springs .

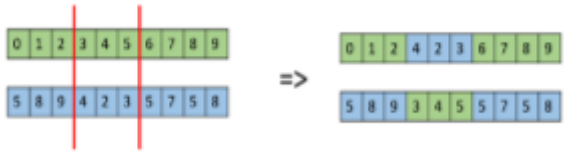

Fig 2: Figure showing single point crossover

II. Multi Point Crossover: Multipoint crossover is a generalization of the one-point crossover where in alternating segments are swapped to get new offsprings .

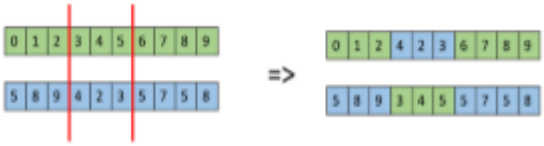

Fig 3: Figure showing multi point crossover

4. Mutation Operation

Mutation is an important part of the genetic search as it helps to prevent the population from stagnating at any local optima. Transformation happens during evolution according to a client definable mutation probability. On the off chance that it is set to high, the search will transform into a primitive random search. There are numerous mutation administrator types, for instance, flip bit, limit, uniform, non-uniform, Gaussian.

I. Bit Flip mutation - In this bit flip mutation, we select at least one random bits and flip them. This is utilized for binary encoded GAs.

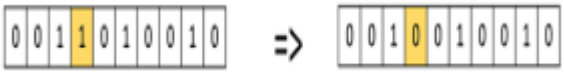

Fig 4: Figure showing mutation operation

5. Fitness Function

The fitness function guarantees that fitness estimation of infeasible solution is less than of any feasible or ideal solution. The more prominent fitness value leads to the minimization of energy consumption and VM migration.

The fitness of an individual x in the population of the GA is defined in Equation Below:

Fitness (x) = Emin/E(x), if x is feasible;
Emin/ (E(x) + Emax), otherwise .

3. CONCLUSION AND FUTURE WORK

Cloud computing technology is progressively being utilized as a part of enterprises and business markets.In cloud paradigm, an effective resource allocation methodology is required for accomplishing client fulfillment and maximizing the benefit for cloud service providers.Each machine is shared by numerous clients and virtual machines are utilized to use these physical machines. With a large number of data centers and each data center having a large number of physical machines, the VM allocation turns into a difficult issue.In this paper we studied on various policies for dynamic resource allocation in cloud based on Topology Aware Resource Allocation, Linear Scheduling Strategy for resource allocation and Dynamic Resource Allocation for Parallel Information Processing, Genetic algorithm.In future we practically compare these algorithms in different environmental setup to check the response time, user satisfaction and maximizing the profit for cloud service providers .

4. REFERENCES

[1] Vinothina V, Shridaran R, Ganpathi P; A survey on resource allocation strategies in cloud computing, International Journal of Advanced Computer Science and Applications, 2012; 3(6):97—104.

[2] Lee G, Tolia N; Ranganathan P, Katz RH; Topology aware resorce allocation for dataintensive workloads, ACM SIGCOMM Computer Communication Review, 2011; 41(1):120—124.

[3] Abirami SP, Ramanathan S; Linear schedulingstrategy for resource allocation in cloud environment, International Journal on Cloud Computing: Services and Architecture(IJCCSA), 2012; 2(1):9—17.

[4] Warneke D, Kao O; Exploiting dynamic resource allocation for efficient parallel data processing in the cloud, IEEE TransactionsOn Parallel And Distributed Systems, 2011.

[5] Inomata A, Morikawa T, Ikebe M, Rahman M; Proposal and Evaluation of Dynamic Resource Allocation Method Based on the Load Of VMs on IaaS, IEEE, 2010.

[6] Minarolli D, Freisleben B; Uitlity–based Resource Allocations for virtual machines in cloud computing, IEEE, 2011.

[7] Jiyani; Adaptive resource allocation for preemptable jobs in cloud systems, IEEE, 2010.

[8] Jung G, Sim KM; Location-Aware Dynamic Resource Allocation Model for Cloud Computing Environment, International Conference on Information and Computer Applications (ICICA), IACSIT Press, Singapore, 2012.

[9] Chandrashekhar PS, Wagh RB; A review of resource allocation policies in cloud computing, World Journal of Science and Technology, 2012; 2(3):165-167.

[10] Tayal S; Tasks Scheduling Optimization for the Cloud Computing systems, International Journal of Advanced Engineering Sciences and Technologies (IJAEST), 2011; 5(2): 111 – 115.

[11] Van HN, Tran FD, Menaud JM; Autonomic virtual resource management for service hosting platforms. In Software Engineering Challenges of Cloud Computing, ICSE Workshop on IEEE, 2009;1-8.

[12] Popovici FI, Wilkes J; Profitable services in an uncertain world. In Proceedings of the 2005 ACM/IEEE conference on Supercomputing (p. 36). IEEE Computer Society. 2005.

[13] Jiayin Li, MeikangQiu, Jian-Wei Niu, Yu Chen; Adaptive resource allocation for preemptable jobs in cloud systems (IEEE, 2010), 31-36.

[14] Melendez JO, Majumdar S; Matchmaking with Limited knowledge of Resources on Clouds and Grids. Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2010; 102– 110.

[15] A. Sidhu, S. Kinger, ―Analysis of load balancing techniques in cloud computing‖, INTERNATIONAL JOURNALOF COMPUTERS & TECHNOLOGY 4 (2) (2013) pages737–741.

[16] MeikangQiu, Senior Member, ―Phase- Change Memory Optimization for Green Cloud with Genetic Algorithm‖ In IEEE dec 2015.

[17] Nusrat Pasha, Dr. Amit Agarwal and Dr. Ravi Rastogi, ―Round Robin Approach for VM Load Balancing Algorithm in Cloud Computing Environment‖ International Journal of Advanced Research in Computer Science and Software Engineering Volume 4, Issue 5, May 2014.

[18] SreenivasVelagapudi, M.Prathap and Kemal Mohammed, ―Load Balancing Techniques: Major Challenge in Cloud Computing – A Systematic Review,‖IEEE, International Conference on Electronics and Communication Systems (ICECS) - Coimbatore, India (2014).