International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 02 Issue: 09 | Dec-2015 www.irjet.net p-ISSN: 2395-0072

# EFFICIENT RESOURCE ALLOCATION AND SCHEDULING ALGORITHM FOR SELF ORGANIZING CLOUD ENVIRONMENT

**Umamaheswari.M .[1], Ananthi.B.[2]**

[1] Research Scholar, Department of Computer Science, Vellalar College for Women,Tamilnadu, India
[2] Head & Associate Professor, Department of Computer Science, Vellalar College for Women,Tamilnadu, India

---

**Abstract** - ***Cloud computing environment provides on-demand access to shared resources that can be managed with minimal interaction of cloud service provider. It is a heterogeneous environment where number of users request for shared resources with different possible conditions. Cloud computing provides reliable and validated services to the users on pay as-you-use basis. In a cloud computing environment, resources are allocated in terms of virtual machines and allocating the virtual machine to an appropriate user. It is very important so as to efficiently utilize limited resources and to satisfy Quality of Service (QoS) requirements.***

***The leveraging virtualization is a key technology in cloud computing, cloud resource can be provisioned on demand in a fine grained multiplexed manner while enabling fault isolation and scalability. Proposed a new resources allocation scheme to utilize the unused computing powers that is efficient enough to allocate resource. A novel multi attribute range query protocol called Pointer Gossiping Protocol (PG-CAN) has been designed to locate the qualified nodes under a randomized policy that mitigates the contention among requesters. The Pointer Gossip is not only achieves maximized resource utilization but also delivers provably and adaptively optimal execution efficiency.***

***Trade of planner (ToF) is a cost model guided greedy approach to search for the optimal solution and is thus an approximation approach. The planner guides ToF to find the optimized transformation sequence for workflows. The selection of each transformation operation in the transformation sequence is guided by a cost model, which estimates the monetary cost and execution time changes introduced by individually applying each transformation operation in the transformation set. Finally concluded the proposed scenario yields superior performance than the existing scenario through Map Reduce and Dynamic Hash Table. The system is developed by NetBeans with Cloudsim packages.***

***Key Words:*** *Cloud resources, Task scheduling, Self Cloud environment and Resource allocation.*

## 1. OVERVIEW OF CLOUD COMPUTING

Internet is the most popular technology and its usage is increasing over a period. In industry, institutes, organizations, offices, business, government sectors, and many other sectors are using internet while running their business. Pay-and-use concept is applied in various businesses. This similar concept is used in cloud computing where through internet infrastructure, platform and software can be provided by vendors as service. This helps users who can be businessman to start his business at low cost by getting platform or infrastructure as a service. Cloud computing environment do not require user's high level equipment which reduces the user's cost for overall business setup.

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. National Institute of Standards and Technology (NIST) is responsible for defining standards in Science and Technology. The Computer Security Division of national institute of social defense (NISD) has provided a formal definition of Cloud computing as shown in figure-1Cloud computing is the model using which user can obtain access to storage spaces and computer resources available in the network. This service is provided by the cloud providers. It provides easy and affordable access to external computer resources through network.

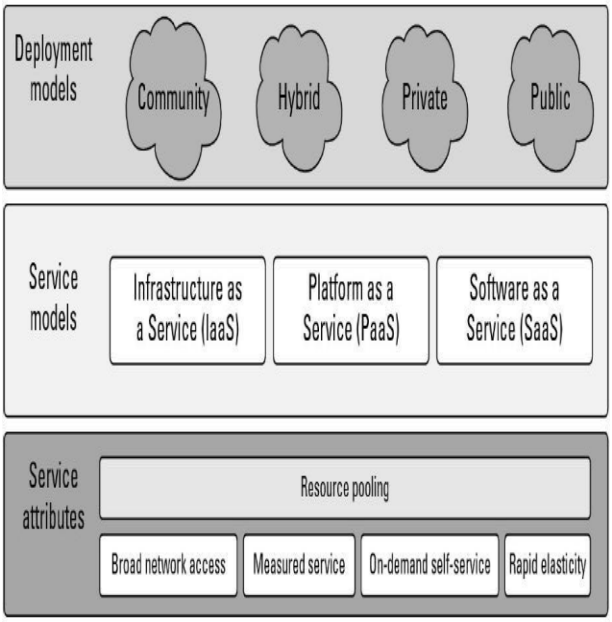


**Figure-1**: NIST definition of cloud computing

## 2. RELATED WORK

**L.Shakkeera., Latha Tamilselvan and Mohamed Imran.,[2013]** has proposed an Optimized Load balancing algorithm in IaaS virtual cloud environment that aims to utilize the virtual cloud resources efficiently. It minimizes the cost of the applications by effectively using cloud resources and identifies the virtual cloud resources that must be suitable for all the applications. The web application is created with many modules. These modules are considered as tasks and these tasks are submitted to the load balancing server. The server which consists our load balancing policies redirect the tasks to the corresponding virtual machines created by Kernel-based Virtual Machine (KVM) virtual machine manager as per the load balancing algorithm. If the size of the database inside the machine exceeds then the load balancing algorithm uses the other virtual machines for further incoming request. The load balancing strategy are evaluated for various Quality of Service (QoS) performance metrics like cost, average execution times, throughput, CPU usage, disk space, memory usage, network transmission and reception rate, resource utilization rate and scheduling success rate for the number of virtual machines and it improves the scalability among resources using load balancing techniques.

**Himani and Harmanbir Singh Sidhu.,[2014]**analyzes that scheduling polices space shared and time shared are compared on the profit, Task penalty, throughput and Net Gain. In that parameter based on space shared scheduling policy shows better results as compared to Time shared scheduling policy. They proposed further research can be done for enhancing the efficiency scheduling policy algorithm for better results by reducing scheduling complexities and improving computations.

**Vijayalakshmi.M and Venkatesa Kumar.V.,[2014]** has proposes optimization technique which resolves the issues of manage large amount of VM requests in the scheduling based algorithm and provisioning with the perspective of cost and security of VMs. It involves the development of proposing a method based on Ant Colony optimization to resolve the problem of load balancing in cloud environment. These job scheduling policies have been extensively evaluated that there is not a single scheduling algorithm that provides superior performance with respect to various types of quality services. Existing scheduling algorithm gives high throughput and cost effective but they do not consider reliability and availability. So that algorithm needs to improve availability and reliability in cloud computing environment.

**Savitha. P and J.Geetha Reddy.,[2013]** have surveyed the various existing workflow scheduling based on genetic algorithms in cloud computing and tabulated their various parameters are VMs and cloudlets, throughput, simulation time, average VM utilization, Average Response time, average processing cost and number of tasks, resource utilization number of job, CPU intensive, IO intensive virtual machine, CPU utility, network bandwidth, the process occupancy service response time along with tools.

**Amit Agarwal and Saloni Jain.,[2014]** analyzed various scheduling algorithm which efficiently schedules the computational tasks in cloud environment. They created First Come First Serve (FCFS), Round Robin (RR) scheduling algorithm and proposed Scheduling algorithm is Generalized Priority Algorithm (GPA). Priority is an important issue of job scheduling in cloud environments. The experiment is conducted for varying number of Virtual Machines and workload traces. The disadvantages of FCFS is that it is non preemptive. The shortest tasks which are at the back of the queue have to wait for the long task at the front to finish .Its turnaround and response is quite low. The CloudSim toolkit supports RR scheduling strategy for internal scheduling of jobs. The drawback of RR is that the largest job takes enough time for completion. Customer defines the priority according to the user demand you have to define the parameter of cloudlet like size, memory, bandwidth scheduling policy etc. In the proposed strategy, the tasks are initially prioritized according to their size such that one having highest size has highest rank. The Virtual Machines are also ranked (prioritized) according to their MIPS value such that the one having highest MIPS has the highest rank. Thus, the key factor for prioritizing tasks is their size and for VM are their MIPS.

## 3. LOAD BALANCING PROBLEM

Load balancing is used to scheduling task and efficient allocation of computing resources. Load balancing problem is to distribute the dynamic workload across multiple nodes of VMs to ensure that no single VMs is overloaded.

## 4. MOTIVATION

- Maximized resource utilization in self-cloud computing environments.
- Novel multi-attribute range query protocol for locating qualified nodes.
- To minimize the monetary cost of workflows while satisfying the predefined deadlines.
- To implement a pointer gossip algorithm in Infrastructure as a Service (IaaS) virtual cloud environment that aims to utilize the cloud resources efficiently.
- It minimizes the cost of the user task by effectively using cloud resources and identifies the virtual cloud resources that must be appropriate for all the applications.

## 5. PROPOSED METHODOLOGY

The proposed works that it's considered the application scenario of users purchasing instances from a public cloud provider to execute their workflows. Due to the pay-as-you-go nature of cloud computing, users need to pay to the public cloud provider. In this scenario, users to submit their workflows to the workflow management system, Pointer Gossip-ToF, with predefined goals on the monetary cost and performance. It assumes users are independent with each other, and there are no dependencies among users.

The Pointer Gossip algorithm is not only achieves maximized resource utilization and also delivers provably and adaptively optimal execution efficiency. It also designs a novel multi-attribute range query protocol for locating qualified nodes. It works effectively to find for each task and its qualified resources under a randomized policy that mitigates the contention among requesters. To aims minimize the monetary cost of workflows while satisfying the predefined deadlines.

This system assumes the users specify the deadline for the entire workflow. The users need not to specify the deadline for each task. The deadline constraint of each task is automatically determined by Pointer Gossip-ToF with an existing deadline assignment algorithm. The deadlines to represent the Quality of Service (QoS) requirements. In order to support flexible settings on the performance and cost constraints, it needs a general workflow optimization approach.

Therefore, the goal of this work is to develop a general and effective optimization framework for workflows in the cloud based on resource availability, which is able to address different cloud offerings and different user requirements.

### 5.1 Virtual Machine configuration

In computing, virtualization refers to the act of creating a virtual version of something, including virtual computer hardware platforms, operating systems, storage devices, and computer network resources. The resources managed by an operating system are virtualized. A multiple VMs can run on a single host at the same time, so, the large number of user requests are handled efficiently.

This module is creating Virtual Machine (VM). It contains instance of CPU, IO speed, bandwidth, Memory Size and Disk Size.Table.2 is shown limitation of resources specified.VM is created based on policies and user specified tasks.

**Table-1:** VM instance description

| Resource | Min.Value | Max.Value |
|---|---|---|
| CPU(unit) | 1 | 25.6 |
| IO Speed(Mbps) | 20 | 80 |
| Band Width(Hz) | 0.1 | 10 |
| Memory Size(GiB) | 512 | 4096 |
| Disk Size(GB) | 20 | 240 |

### 5.2 Query based Resource allocation

The query message is used to set users task in execution time. Time is considered as dead line of the user tasks. Cloud optimizer will try to complete the process of the given user tasks within time of dead line.

User sending request to cloud optimizer using query messages. The query message means that user specified which type of resource and their duration time. The two query messages are single query and range query. The single query is required to select type of resource and resource usage duration time that considered as deadline of the resource utilization. The range query is required to select the multi resources and time for optimize the user tasks.

### 5.3 Task Scheduling

The task scheduling is based on the time and available resources. The cloud optimizer gets the task from the user. The user gives the deadline for that task. The cloud optimizer is scheduled the task based on the deadline by available of the resources. The scheduling task is considered on which appropriate resource for that task.

## 6. PROPOSED ALGORITHM

### 6.1 Pointer-Gossiping Content Addressble Network

This approach designs a novel multiattribute range query protocol for locating qualified nodes. However due to dynamic resource provisioning technologies used in cloud, locating a node which contains a combination of available resources become the challenging problem due to frequent resources repartitioning and reallocation. To satisfy this problem the proposed Resources Discovery protocol has been included. This protocol aims to find the qualified resources with minimum contention among the users on task's demand. It is unique among other methodology in which during the entire course of discovery there is only one query message proposed in the network. The procedure of resources query is, when a node generates a query message, it will first be routed to its duty node, on the node each stored record will be checked against the message's demand. If the node keeps the enough qualified records for the query they will be returned to the requesting node and query will be terminated. If there are no matched records, a few other duty nodes pointed by the current duty node will be randomly selected and encapsulated.

In dynamic proportional share model multiple types of resources can be shared by every task on the same node, so the resources can be utilized more effectively. This model is different from traditional resource allocation model. Although there many strategies for VM-multiplexing they are not suited to Dynamic proportional share model which focus on few attributes like CPU, IO speed, Bandwidth, Disk space and Memory space. This model allows each task to utilize the resources dynamically and effectively.

#### 6.1.1 Algorithm 1: This program at node PG-CAN is invoked upon receiving a query message

1. if (the current node is the duty node) then
2. Search Resource record list and put the qualified records in Found List;
3. if (Found List is not empty) then
4. Send Found List to the requesting node;
5. Return;   /*Query is terminated here*/
6. else
7. Construct Jump List by randomly selecting a few pointed duty nodes(VM);
8. if (Jump List is not empty) then
9. Randomly take out a duty node and remove it from the Jump List;
10. Send the query message with Jump List to the selected duty node;
11. end if
12. end if
13. else
14. Forward the query message based on PG-CAN's routing rules;
15. End if.

### 6.2 Trade Of Planner (Tof)

Cloud providers lease computing resources in the form of VMs (or instances). Typically, cloud providers offer multiple types of instances with different capabilities such as CPU speed, RAM size, I/O speed and network bandwidth to satisfy different application demands. Different instance types are charged with different prices. Tables 1 and 2 show the prices and capabilities of four on-demand instance types offered by Amazon EC2 and Rackspace, respectively. Amazon EC2 mainly charges according to the CPU, whereas Rackspace mainly on the RAM size. Both cloud providers adopt the instance hour billing model, whereby partial instance hour usage is rounded up to one hour. Each instance has a non-ignorable instance acquisition time. For simplicity, we assume the acquisition time is a constant, lag. In this paper, we consider a single cloud provider with I instance types, with assigned IDs to the instance types in the increasing order of their prices. IDs are consecutive integers, from one to *I*.

**Table-2**: Price ($/hr) and capabilities of four of on-demand instance in Amazon EC2

| Type | CPU | Memmory (GB) | Disk (GB) | Net. | Price |
|---|---|---|---|---|---|
| Small | 1 | 1.7 | 1*160 | Low | 0.06 |
| Medium | 2 | 3.75 | 1*410 | Moderate | 0.12 |
| Large | 4 | 7.5 | 2*420 | Moderate | 0.24 |
| X large | 8 | 15 | 2*840 | High | 0.48 |

**Table-3**: Price ($/hr) and Capabilities of Four Instance Type in Rackspace

| RAM(GB) | CPU | Disk(GB) | Net(Mbps) | Price |
|---|---|---|---|---|
| 0.5 | 1 | 20 | 20/40 | 0.022 |
| 1 | 1 | 40 | 30/60 | 0.06 |
| 2 | 2 | 80 | 60/120 | 0.12 |
| 4 | 2 | 160 | 100/200 | 0.24 |

Having formulated the transformation set, introduce implementation of the cost-model based planner in ToF. ToF is a cost model guided greedy approach to search for the optimal solution and is thus an approximation approach. The planner guides ToF to find the optimized transformation sequence for workflows. The selection of each transformation operation in the transformation sequence is guided by a cost model, which estimates the monetary cost and execution time changes introduced by individually applying each transformation operation in the transformation set.

The order of applying transformation operations also matters for performance and cost optimizations. The searching space for an optimal transformation sequence is huge. Second, the optimization is an online process and should be lightweight. Find a good balance between the quality of the transformation sequence and the runtime overhead of the planner. Due to the huge space, a thorough exploration of the optimization space is impractical. Third, the planner should be able to handle different tradeoffs on the monetary cost and performance goals.

### 6.2.1 Algorithm 2: TRADE OF PLANNER

1. Queue all coming workflows in a queue $Q$;
2. for each workflow $w$ in $Q$ do
3. Determine the initial assigned instance type for each task in *w;* repeat
4. for each $O_m$ in main schemes (i.e., Merge and Demote) do
5. Pretend to apply $O_m$ and check whether the earliest start or latest end time constraint of any task in w is violated after applying $O_m$;
6. if No time constraint is violated then
7. Estimate the cost reduced by performing 0m using the Cost model;
8. Select and perform the operation in main schemes which has the largest cost reduction;
9. For each $O_\alpha$ in auxiliary scheme (i.e., Move, Promote and Co-scheduling) do
10. Pretend to apply $O_\alpha$ and check whether the earliest start or latest end time constraint of any task in w is violated after applying $O_\alpha$;
11. if No time constraint is violated then
12. Estimate the cost reduced by performing $O_\alpha$ using the Cost model
13. Select and perform the operation in auxiliary schemes which has the largest cost reduction;
14. until No operation has a cost reduction ;
15. return Optimized instance assignment graph for each workflow.

Algorithm 2 illustrates the overall optimization process of ToF in one plan period, which has implemented all the three above-mentioned designs. Each task in the workflow is assigned with an instance type determined by an instance assignment heuristic. Plan period is an important tuning parameter in the planner. If the period is long, more workflows are buffered in the queue. To make the optimization plan, more combinations of tasks need to be checked for transformations and the optimization space becomes much larger. When the period parameter is short, the optimization space gets smaller and the chance for operating transformations to reduce cost decreases.

## 7. RESULT AND DISCUSSION

Experimental analysis is indented to be of use to researchers from all fields who wants to study algorithms experimentally. It has two goals: first, to provide a useful guide to new experimentalists about how such work can best be performed and written up, and second, to challenge current researchers to think about whether their own work might be improved from a scientific point of view. Efficient implementations allow one to perform experiments on more and/or larger instances or to finish the study more quickly.

### EXPERIMENTAL RESULT

**Table.4.**Cost Estimation

| Job type | Cost($) | |
|---|---|---|
| | Existing | Proposed |
| Small job(0$-6$) | 5 | 2.5 |
| Medium job(6.1$-12$) | 10 | 7.5 |
| Large job(12$-20$) | 17.5 | 15 |

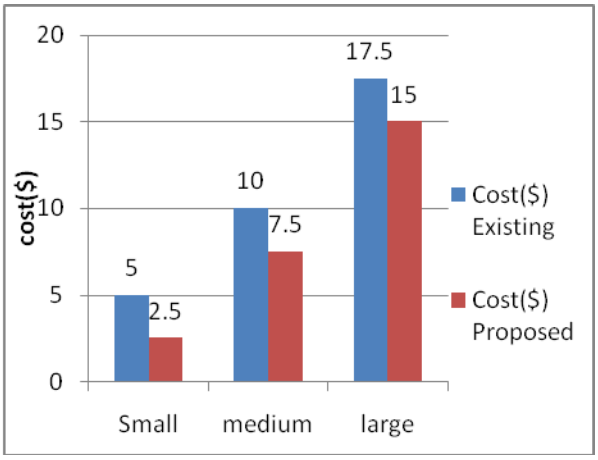


**CHART-1:** Cost Estimation graph

From the Table.6.2 and graph.6.2, comparison of existing and proposed system in terms of throughput metric is observed. Plot the Cost ($) and Job types are small, medium, large. In existing scenario, the cost values are lower by using Map Reduce and DHT algorithm. The small job value of existing scenario is 5 $.In proposed system, the small job value is higher by using the pointer gossip content addressable network and trade of planner algorithm is 2.5$. The medium job value of existing scenario is 10$. In proposed system, the medium job value is higher by using the pointer gossip content addressable network and trade of planner algorithm is 7$.The large job value of existing scenario is 17.5$. In proposed system, the large job value is higher by using the pointer gossip content addressable network and trade of planner algorithm is 15$.From the result, concluded that proposed system is superior in performance.

From the experimental results, The existing system achieves 74% by considering the parameters(throughput and cloud cost) and proposed system achieves 86.5%.Finally it is concluded that the proposed system is superior than existing system by means of 12.5% in performance.

## *8.* CONCLUSION

The proposed work, resource utilization is highly optimized through making sure that every computing resource is distributed efficiently by using Pointer Gossip Content Addressable Network algorithm. The solution is formulated which can optimize the task execution performance based on its assigned resources under the user budget by using Trade of Planner algorithm. Resources contention problem is minimized. From the experimental results, The existing system achieves 74% by considering the parameters(throughput and cloud cost) and proposed system achieves 86.5%.Finally it is concluded that the proposed system is superior than existing system by means of 12.5% in performance. Research study is the complete in-depth analysis on a specific area. The research will have impact on the future and is an on-going activity that never ends. The research work can be enhanced by considering the Trade of Planner on multiple clouds. Unmoving, there are many practical and challenging issues for current self-cloud environments.

## REFERENCES

1. Amit Agarwal, Saloni Jain," Efficient Optimal Algorithm of Task Scheduling in Cloud Computing Environment", International Journal of Computer Trends and Technology (IJCTT),volume 9 number 7, Mar 2014.
2. Hien Nguyen Van, Frederic Dang Tran, Jean-Marc Menaud," SLA-aware virtual resource management for cloud infrastructures", 9th IEEE International Conference on Computer and Information Technology, Oct 2009.
3. Saranya.S, Saranya.N," An Efficient Resource Allocation for Improving Resource Utilization in Self Organizing Clouds", International Journal of Innovative Research in Computer and Communication Engineering, Volume.2, Special Issue 1, March 2014.
4. Sergey Blagodurov, Daniel Gmach, Martin Arlitt, Yuan Chen, Chris Hyser, Alexandra Fedorova. Maximizing Server Utilization while Meeting Critical SLAs via Weight-Based Collocation Management., IEEE ,2013
5. Sheng Di, Cho-Li Wang, "Dynamic Optimization of Multiattribute Resource Allocation in Self-Organizing Clouds", IEEE Transactions on Parallel & Distributed Systems, vol.24, no. 3, pp. 464-478, March 2013.