An efficient resource management in cloud computing

Bashir Yusuf Bichi
Kano University of Science & Technology
3 PUBLICATIONS **0** CITATIONS
SEE PROFILE

Anas Muaz Kademi
Yasar University
2 PUBLICATIONS **0** CITATIONS
SEE PROFILE

Tuncay Ercan
Yasar University
83 PUBLICATIONS **543** CITATIONS
SEE PROFILE

An Efficient Resource Management in Cloud Computing

Bashir Yusuf Bichi[*], TuncayErcan[*],AnasMu'azuKademi[*]

[#]*Yasar University, Department of Computer Engineering*
Izmir,Turkey

bbichi2009@yahoo.com, tuncay.ercan@.yasar.edu.tr
anasmkademi@gmail.com

***Abstract*— Ascloud computing is gaining more recognition to the public, request for services to a given task within the virtual environment of the cloud also increases.This paper proposed a max-min algorithm liked technique with the aim ofdevelopinga new framework that tends to balance the load that may be experienced due to the high demand of resources by a set of task within the virtual environmentof the cloud computing ecosystem.**

***Keywords*— Load balancing, Max-min algorithm, Makespan, Min-min algorithm, Task Scheduling, Resource Allocation.**

I. Introduction

The fast development in the area of computing gives users of computer system the opportunity to have access as well as exploit the vast resources that are contained within the system. Cloud Computing is seen as a new technology that adopt the aspect of distributed computing and internet. The concept of cloud is to allow a client or a customer to have access to computing resources through the use of web services in a more efficient manner. Resources in the field of information and communicationtechnology are more or less the fundamental elements that include some part of computer systems, computer networks, software application and so on. Managing these resources involve controlling and limiting access to the pool of resources that are been shared. This brings the concept of an agreement between the resource services providers and the clients otherwise known as Service Level Agreement (SLA), The idea behind the SLA is to restrict access to a given resources.

Cloud resource management may involve some policies such as admission control, Quality of Service (QoS) which involves specification from SLA. Other policy which is the main topic regarding this research is Load Balancing which involves balancing the work load evenly among the cloud servers.The cloud services are provided to customers through the use of virtual hardware, the services provided to the client such as IaaS, SaaS, and PaaS as shown in Fig. 1can be scale up or down depending on the client's level of usage and the SLA adopted .

Revolutionary increase of users and demand for various services parallel with the need for efficient resources usage reveal that load balancing should be done correctly and efficiently. The aimed of which is to optimize resourceutilization, maximize throughput, minimize response time, and avoiding overload. Many load balancing algorithmshave been designed so far, and the need for much more efficient algorithm that, with other things, also fairly allocate the loads across the system is high. Moreover, each developed algorithm has its own draw back and mostly performs better in one application than in others [12].

Load balancing, the absence of which negatively rises some issues (in performance, availability), is one of the primary challenges in cloud computing [12] [14]. For improvement upon the available solutions, of which min-max relatively performs well [13], a modified and extensively improved algorithm is formed.

The other part of this paper is organized as follows: Section 2 discussed on some previous studies related to task scheduling algorithms. In Section 3 and 4 we introduce some concept with regard to the technique for task scheduling and resource allocation in cloud computing, section 4 discuss on the issues pertaining load balancing, mathematical formulation, and the proposed algorithm.In section 5, mathematical simulation and results arediscussed, and lastly concluding remarks are given in Section 7.

II. Related Work

As research in cloud environment is increasing, task scheduling needs to be more scalable to the user demands. [1][2] Uses a technique known as improved max-min algorithm and enhance max-min algorithm respectively with the aim of distributing the load among the available resources. [1] is a modification of [2] in which both uses the max-min task scheduling algorithm. This paper employs the technique in [1] to propose another algorithm that will help in balancing load across the virtual resources and to allow for scalability when handing task with the aim of improving the performance of the system. Load balancing over resources in the cloud environment is used toachieveminimum load when using resources, different methods are use to achieved such balance as stated in [7, 8, and 9]. Based on these methods, we take interest in max-min algorithm and shows how load can be balanced across different resources in the cloud environment.

III. Cloud Computing and Task Scheduling

As the aspect of parallel and distributed computing involves Cloud computingwhich is a collection of computers that are interconnected and virtualized as one computing resources, cloud client get access to the resources through the SLA [11].

As mention previously cloud computing offers software, platform, and infrastructure as a service respectively. The software as a service includes providing software such as Mail (e.g. Gmail, Yahoo mail), social network sites, Google drive, and so on, to the customers or clients. The infrastructure as a service deals with VM, storage, network, load balancer and so on as a service to the client and lastly the platform as a service deals with database like sql, oracle, web services, runtime (e.g. java) and so on as a service to the client. The clients get access to these services through various devices as shown in the figure below [3] [4].

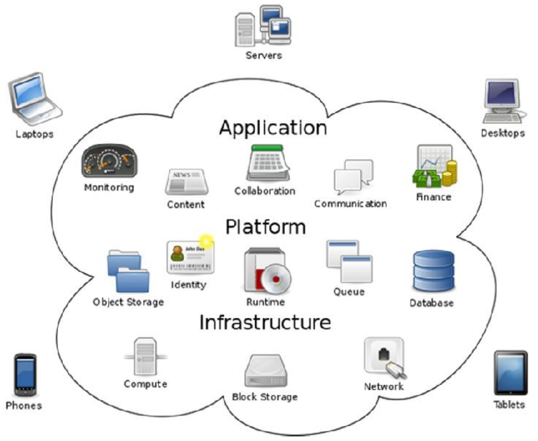

Fig.2.ServicetypesinCloudComputing [http://en.m.wikipedia.org/wiki/Cloud_computing].

Task scheduling is a well known concept as it is a vital aspect in cloud computing. It allows for scheduling virtual resources over the cloud to keep a balance load across the resources [7] as indicated in the figureFig. 2 below.

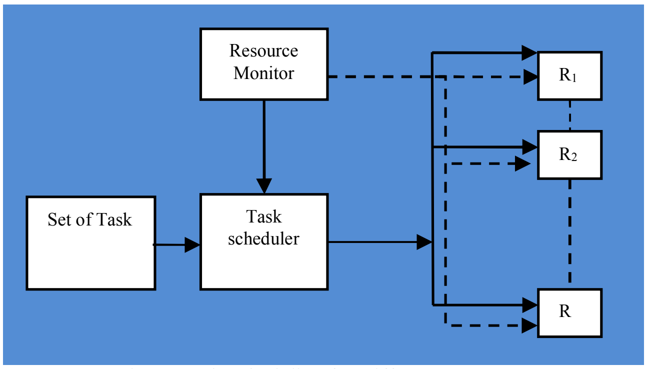

Fig. 2 -Task Scheduling for Different Resources

IV. Virtual Resources and Allocation

As it is shown in Fig. 2 above, the users send task to the cloud environment with different requirement to the cloud service providers. The requirement can be tasks with different set of data size and processing power, the task scheduler will then match the tasks with available resources (virtual resources) that are available.

Resources in cloud computing cover all useful entities which can be use through the cloud platform. These resources include storage, memory, network bandwidth, and virtual machine [3]. The resources can be virtualized and provisioned from the existing physical resources in the cloud environment. The parameters that are virtualized include; the CPU, memory, disk etc. The provisioning can be done by mapping these virtualized resources to their corresponding physical ones. Resource allocation in cloud computing is all about assigning available resources to a needing cloud application. Dynamic resource management is seen as a very active research area in the field of cloud computing. The cloud computing resources costs vary depending upon the type of configuration for using such resources. Therefore an efficient use of these resources is considered as a prime interest for both the customer/client and the cloud provider. Resource allocation in cloud computing takes place in two levels [5];firstly load is balanced within the physical machine whenever an application is uploaded and secondly, request are assigned to a specific application if there are multiple request for the resource.

Resource allocation exhibits some benefits irrespective of the organization size or business market. It also have some limitations, below are some set of advantages and limitations of resource allocation [6]; among the advantages is that users shares their resources whenever there is scarcity of resources. The limitation may arise in the aspect of securitywhich is a major challenge and another setback is that migration issue may arise whenever a client decides to switch service providers.

V. Load Balancing and Task Scheduling Algorithm

Load balancing is a technique used to distribute processing load (i.e. large processing load) to smaller processing nodes (i.e. resources) to enhance the overall performance within the system in a distributed environment as shown in fig.2 above. The idea of load balancing is to avoid loading up a resource during task scheduling so that all the resources will be allocated with a task evenly across a given virtual environment. Various load balancing algorithms exist as stated in [7] with the aim of distributing the task's load across resources. Some of these algorithms include;

- Min-Min Algorithm

This algorithm has all the relevant information needed in advance. The algorithm uses some parameters to obtain the information it needs. Some of these parameters are; ETC (Expected Time Compute), MET (Minimum Execution Time), MTC (Minimum Completion Time) etc. The Min-Min algorithm selects a task with minimum completion time and maps it with a node with a minimum completion time [8].

- Max-Min Algorithm:

Max-min algorithm chooses large task to be executed firstly before executing small once [10]. This algorithm works almost the same way as the Min-Min algorithm except in Max-Min the task with maximum value is selected from the set of execution time of tasks and maps it to a node with minimum completion time. The ready time of the node is updated by adding the execution time of the task [7, 8].

As the cloud users sends task to the cloud environment with different requirement to the cloud service providers. The requirement can be tasks with different set of data size and processing power, the task scheduler will then match the tasks with available resources (virtual resources) that are available. Some mathematical relations are given in [9] to analyze resources scheduling in cloud computing which are employed and used in this paper are given below.

The set of VMs V with their respective processing power is given as;

$$V = \{v_j(c_j) | \, j = 1,2, \dots k\} \qquad (1)$$

The set of tasks is also given as

$$T = \{t_i(a_j, b_j) | \, i = 1,2, \dots y\} \qquad (2)$$

Where

c_j = processing speed (MIPS)

t_i = given task i

a_i =Data file size of a given task (Mb)

b_i = Processing power (MI) of a task t_i

With above equations (1) and (2) the expected execution time (EET) for a given task by a virtual resource can be obtained as;

$$EET = Size\ of\ task\ (MI)/Computing\ Power\ of\ resource\ (MIP \ (3)$$

Now with equation (3) above, another metric can be obtained, which is the completion time (CT) of task t_i by a given resource.

$$CT_{(i,j)} = EET_{(i,j)} + r_i \qquad (4)$$

Where r_i indicate the starting time of the execution of task t_i.

Using (4), another important metric can be obtained, which is called the makespan, define as a measure of the throughput of the heterogeneous computing system [7].

$$makespan = Max_{i\in w, j\in y)}(CT_{i,j}) \qquad (5)$$

This paper employs a known scheduling algorithm called an improved max-min algorithm from [1] and then based on this algorithm we propose another algorithm that will help in balancing load across the VMs' resources to improve the performance of the system.

- Improved Max-Min Algorithm

The Max-min algorithm allocated task $t_i$to resource v_j such that large tasks have higher priority. For instance for a given large task, the max-min algorithm execute smaller task concurrently while running large tasks. Therefore, the largest task determines the total makespan for other resources. The improved max-min algorithm is given below [2].

```
For all submitted tasks in Meta-task; $t_i$
   for all resources; $v_i$
               $C_{ij} = E_{ij} + t_j$
   Find task $t_k$ costs maximum execution time
   Assign task $t_k$ to resource $v_i$ which gives minimum
   completion time
Remove task $t_k$ from Meta-tasks set.
Update $t_j$ for selected$v_j$.
Update $c_{ij}$ for all j.
While Meta-task not Empty
   Find task $t_k$ costs maximum execution time.
   Assign task $t_k$ to resource $v_j$ which gives minimum
   completion time
   Remove Task $t_k$ form Meta-tasks set.
   Update $t_j$ for selected$v_j$.
   Update $c_{ij}$ for all j.
```

- Proposed Algorithm

The improved max-min algorithm is reliable and proved to be efficient in scheduling the set of tasks to the available resources. However to make effective and sufficient use of resource a proposed algorithm was introduced which is based on the improved max-min algorithm but small changes are made to make sure that all resources are used sufficiently and to minimize the use of these resources if few once can perform the task. The proposed algorithm is shown in the pseudo code below;

```
For all submitted tasks in Meta-task; Ti
   For all resources; Rj
            $C_{ij} = E_{ij} + t_j$
   Find task $T_k$ costs maximum execution time
   Assign task $T_k$ to its corresponding resources $R_j$
Remove task $T_k$ from Meta-tasks set.
Update $r_j$ for selected $R_j$.
Update $C_{ij}$ for all j.
Pivot= $T_k$;
For all updated task in Meta-task; Ti
   For all updated resources; $R_j$
   Find task $T_h$ costs maximum execution time
   Assign task $T_h$ to its corresponding resource $R_j$
Remove task $T_h$ from Meta-tasks set.
Update $r_j$ for selected $R_j$.
Update $C_{ij}$ for all j.
2pivot= $t_h$
While Meta-task not Empty
   Find task $T_g$ costs maximum execution time.
      If 2pivot+$t_g$ ≤Pivot then
         Assign task $T_g$ to previous resource $R_j$ which
         gives minimum completion time
   Remove Task $T_g$ form Meta-tasks set.
Update $r_j$ for Selected $R_j$.
   Update $C_{ij}$ for all j.
   Update 2pivot.
      Else
         Assign task $T_g$ to resource its
      corresponding resource $R_j$
      Remove Task $T_k$ form Meta-tasks set.
      Update $r_j$ for Selected $R_j$.
      Update $C_{ij}$ for all j.
```

In the algorithm the total makespan is made to be a pivot 1 value for the first step and another pivot 2 value is assigned during the second step of the execution. Then during the next execution step the second pivot value and the completion time of the current state are summed up together. If they are greater than the first pivot value, then a new resource is allocated to that task.

By given this criteria, the resources can be used in a balanced manner and fewer resources can be used, the remaining resources will not be involved to minimize the use of such resources. The aim of the above algorithm when compared to the improved Max-Min algorithm is to make effective used of the available resource during scheduling.

The flowchart for the above pseudo code is given in the figure (Fig. 3)

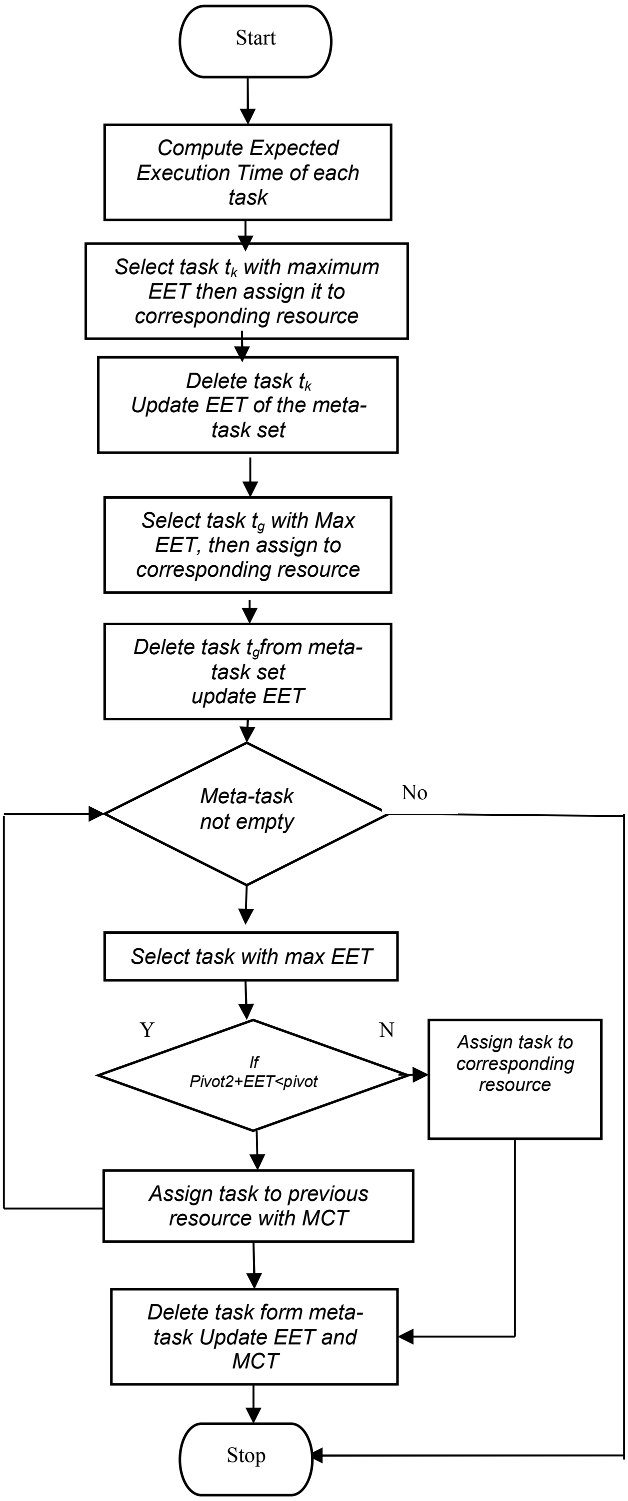
Start
Compute Expected Execution Time of each task
Select task t_k with maximum EET then assign it to corresponding resource
Delete task t_k Update EET of the meta-task set
Select task t_g with Max EET, then assign to corresponding resource
Delete task $t_g$from meta-task set update EET
Meta-task not empty
No
Select task with max EET
Y
N
If Pivot2+EET<pivot
Assign task to corresponding resource
Assign task to previous resource with MCT
Delete task form meta-task Update EET and MCT
Stop

Fig. 3: Flowchart of ProposedAlgorithm

VI. PROPOSED ALGORITHM RESULT AND ANALYSIS

Scenario: A theoretical analysis ofpredefined meta-task values and resources are used to carry out the scheduling process as given below. The tables underneath shows the meta-task values and the resources used.

TABLE I
TASKS VALUES

Task	Size of task (MI)	Data volume (Mb)
T1	522	200
T2	1128	500
T3	430	300
T4	340	410
T5	570	328

Table he table (table 2) below holds the processing speed and the bandwidth of the resources on a network system.

TABLE 2
RESOURCE PROCESSING SPEED AND BANDWIDTH

R	Processing speed (MIPS)	Bandwidth (MbPS)
R1	130	100
R2	266	120
R3	294	150

Given the above values, Matlab is employed to compute the expected execution time of each task and the results are tabulated and analyzed as given in Table 3 below;

TABLE 3
EXPECTED EXECUTION TIME OF TASK

	R1	R2	R3
T1	4.015	1.962	1.776
T2	8.677	4.241	3.837
T3	3.308	1.617	1.463
T4	2.615	1.278	1.156
T5	4.389	2.143	1.938

From the tables above i.e. Table 3: T_i with maximum execution time is selected and then is assigned to the corresponding resource R_i. Fig. 4 shows how the allocation was performed based on the max-min idea, the task are allocated to all the available resources within the scheduler, the processing time is measured in seconds.

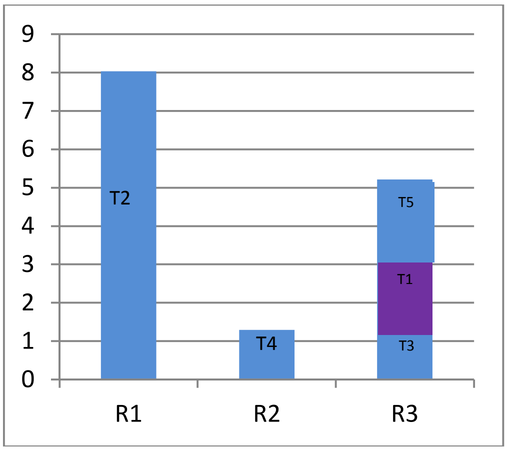

Fig. 4: Chart for Resource Allocation for Max-Min Algorithm

In contrast to theproposed max-min scheduling algorithm the figure fig. 4 below shows how the allocation is performed.

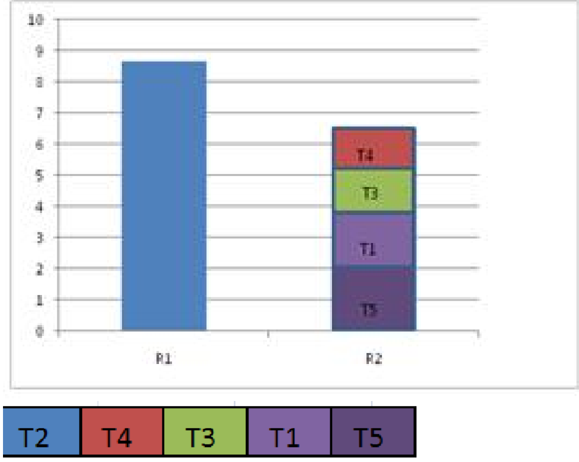

Fig. 5: Chart for Resource Allocation for proposed Max-Min Algorithm

From the chart (fig. 4), the largest task T1 (the blue colour) has a maximum makespan of 8.031 and it's scheduled to resource R1. The maximum makespan, is considered as the maximum throughput for other resources the summation of the remaining task i.e. T4 (red), T3(green), T1 (purple) and T5 (dark purple) are not up to the maximum throughput therefore they are assigned to the second resource R2. This makes it possible to balance different smaller tasks to run concurrently on different resources across the system and also to use the resources wisely when needed. Another important factor which is based on the on-demand characteristics of cloud computing is that, the number of resources used is also minimized and a resource can be put into use when there is a demand for that resource. Based on the results obtained, instead of assigning the load to the three resources, it's possible to assign the task to only two resources, thereby increasing the efficiency of the system, thus we can have many task running concurrently on some resources and other resources can be put in use only when the need arise.

VII. CONCLUSIONS

In conclusion Cloud Computing is an on-demand service, therefore, efficient on-demand allocation of VM is needed. In thispaper technique to handle on-demand allocation is analysed and it proved to be effective. Allocation of resources can be performed efficiently within a cloud environment by balancing the load across the various virtual machine resources, by employing an efficient technique for load balancing such as the max-min algorithm that was used in this paper.

The usage of max-min technique made it possible to handle resources in an efficient and balanced manner. Thus, for a better service to be experienced in a field of cloud computing, a proper and efficient allocation techniques need to be adopted.

REFERENCES

[1] UpendraBhoi, Purvi N. Ramanuj. Enhance Max-Min Task Scheduling Algorithm in Cloud Computing. International Journal of Application or Innovation Engineering & Management. 2013.

[2] O. M. Elzeki, M. Z. ReshadandM. A. Elsoud. Improved Max-Min Algorithm in Cloud Computing. *International Journal of Computer Applications (0975 – 8887) Volume 50 – No.12, July 2012*.

[3] Y Yuan, W-Cai Liu. Efficient resource management for cloud computing 2011.

[4] Ryan Knight, The new role of XML in cloud data integration Using XML to integrate Salesforce data with enterprise applications. June 2009.

[5] R Shelke, R Rajani.Dynamic resource allocation in Cloud Computing. 2013.

[6] Ronak Pate, Sanjay Patel, Survey on Resource Allocation Strategies in CloudComputing. *International Journal of Computer Applications (0975 – 8887) Volume 50 – No.12, July 2012*

[7] S. SwaroopMoharana, D. Rajadeepan. Analysis of Load Balancer in Cloud Computing. International Journal of Computer Science and Engineering Vol.2 2013.

[8] D. Manan Shah, A. Amit Kariyani, L. Dipak Agrawal. Allocation of Virtual Machines in Cloud Computing using Load Balancing Algorithm. International Journal of Computer Science and Information Technology & Security. Vol. 3 2013.

[9] Yichao Yang, Yanbo Zhou. Heuristic Scheduling Algorithms for Allocation of Virtualized Network and Computing Resources. Journal of Software Engineering and Application 2013.

[10] Pinal Salot , A survey of various scheduling algorithm in cloud computing environment, IJRET | FEB 2013

[11] . Patel, Pankesh, Ajith H. Ranabahu, and Amit P. Sheth. "Service level agreement in cloud computing." (2009).

[12] G. Gopinath, S. Vasudevan An in-depth analysis and study of Load balancingtechniques in the cloud computing environment.2ndInternational Symposium on Big Data and Cloud Computing. 2015

[13] NayandeepSran, Navdeep Kaur. Comparative Analysis of Existing Load Balancing Techniques in Cloud Computing.International Journal of Engineering Science Invention.Vol. 2 Issue 1 2013. PP.60-63

[14] Garima Joshi, S.K. VermaA Review on Load Balancing Approach in Cloud Computing.International Journal of Computer Applications (0975 – 8887) Volume 119 – No.20, June 2015.