International Journal of Computer Applications (0975 – 8887)
Volume 93 – No.8, May 2014

# Improve Scheduling Task based Task Grouping in Cloud Computing System

Lailah M.Mustafa
B.Sc. Computers and Systems Engineering Department , Alzher University, Egypt

Mohamed K. Elmahy
Maj. Gen.prof. Chairman of STCE, Egypt

Mohamed H.Haggag
Ass. Prof. Department of Computer Science, Helwan University, Egypt

## ABSTRACT
In recently years, the information communication technology (ICT) appeared new paradigm of utility computing called cloud computing. The consumer cloud is always important of high performance for cloud computing service and satisfy service agree level (SLA). In cloud computing, there is a need of further improvement in task scheduling algorithm to group of tasks, which will reduce the response time and enhance computing resource utilization. This grouping strategy considers the processing capacity, memory size and service type requirement of each task to realize the optimization for cloud computing environment. It also improves computation/communication ratio and utilization of available resources by grouping the user tasks before resource allocation. The experimental results were conducted in a simulation cloud computing environment by generator services and tasks request for consumer cloud. The results show that gives comparator between our strategies and improve activity based costing algorithm.

## General Terms
Cloud Computing System, Scheduling task.

## Keywords
Service Level Agreement (SLA), grid model, Cloud Information Service (CIS)

## 1. INTRODUCTION
Cloud computing is new paradigm of utility computing after cluster and grid computing. From the Google trend, it can be observed that the last five years, Cloud computing more interesting than other paradigms. the National Institute of Standards and Technology (NIST) has been defined of cloud computing is a type of parallel and distributed system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resource(s) based on service-level agreements established through negotiation between the service provider and consumer [1], so the basic characteristics of cloud computing are parallel, distribution, virtualization and dynamic resource. The cloud computing have three delivery services available to cloud consumer, they are Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) such as social Networks, application deployment, compute and storage [2].

Figure 1 depicts hierarchical view for cloud computing [3]. Data Centers is the foundation of cloud computing which provides the hardware the clouds run on. Data centers are usually built in less populated areas with cheaper energy rate and lower probability of natural disasters [3]. Top of data centers layer, IaaS layer virtualizes computing power, storage and network connectivity of the datacenters, and offers it as provisioned services to consumers. Users can scale up and down these computing resources on demand dynamically [3].

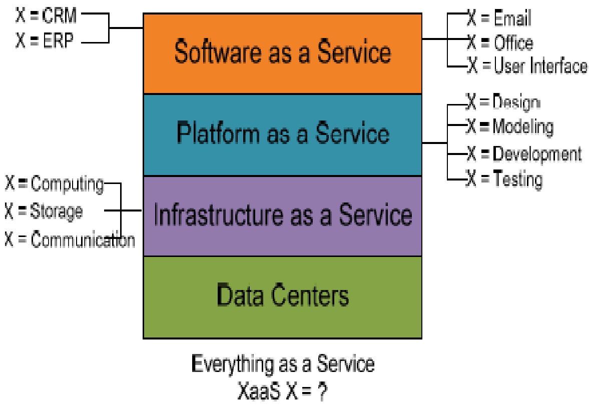


**Fig 1: Hierarchical View of Cloud Computing**

PaaS, often referred as cloud ware, provides a development platform with a set of services to assist application design, development, testing, deployment, monitoring, hosting on the cloud. It usually requires no software download or installation, and supports geographically distributed teams to work on projects collaboratively [3]. In SaaS, Software is presented to the end users as services on demand, usually in a browser. It saves the users from the troubles of software deployment and maintenance. The software is often shared by multiple tenants, automatically updated from the clouds, and no additional license needs to be purchased. Features can be requested on demand, and are rolled out more frequently [3].

The SLA is a contract negotiated and agreed between a customer and a service provider [3]. This is SLAs can cover terms regarding the quality of service for performance system so cloud consumers need SLAs to specify the technical performance requirements fulfilled by a cloud provider such as processing capacity for compute resource. In this case, the provider may specify its requirements on capability, flexibility and functionality in SLA in order to provide essential requirements SLA for cloud consumers [3].

In this work, a scheduling of task classification tasks based on service type. Then, it is using task grouping concept. The scheduler retrieves information of the processing capability and memory size for computing resource. Then, the scheduler selects the appropriate computing resource target to groups based on processing capability and memory size for resource in order to attain the reduce for response time of task execution. Thus, improves computation/communication ratio and utilization of available resources.

The CloudSim toolkit has been used to test the task grouping and scheduling in a simulation cloud computing environment. The toolkit, a java-based discrete-event cloud computing simulation package, supported both system and behavior modeling of cloud computing system components such as data center and virtual machines (VMs). Mapping of task to resource and resource management are also supported [5].

This paper is organized shows as follows: Section 2 discusses the related work, Section 3 architecture of task-grouping schedule, Section 4 the algorithm of task-grouping schedule, Section 5 the present experimental result and Section 6 gives conclusion and future work and lastly, the references.

## 2. RELATED WORK

In this section, some reprehensive research works on task grouping and scheduling in distributed computing system and cloud computing environment have been surveyed. Grouping-Based job scheduling model in grid computing [6], group jobs according to MIPS, memory size and bandwidth of the resource. This model reduces the processing time of jobs, utilize grid resources sufficiently, network delay to schedule and execute jobs on the grid due to this study presented and evaluation an extension from computational-communication to computational- communication-memory based grouping job scheduling strategy, but the algorithm doesn't parallel schedule resource. Scheduling framework for bandwidth-aware job grouping-based scheduling in grid computing [7], group jobs according to MIPS and bandwidth of resource. This model reduces processing time of jobs compare to a non bandwidth-aware job grouping scheduling framework. Dynamic job grouping-based scheduling for deploying application [8] group jobs according to MIPS of resource only. This model reduces processing time, communication time of jobs and cost. In the same way, improve activity based cost algorithm for cloud computing [9]. Before group tasks scheduling, tasks are sorted according to their priority and they are place in three different list based on three levels of priority. They are high, medium and low priority. This model according MIPS of resource only when the scheduler is grouped task. This algorithm improved the computation-communication ratio. It also is minimization of makespan and cost compare to activity based cost algorithm.

Moreover, this study focuses and evaluates dynamic grouping based task scheduling where classification task based on service type before there are grouping. The propose model minimization of processing time in cloud computing. This model is according to MIPS and memory size of computing resource, and MI and file size of tasks.

## 3. ARCHITECTURE OF TASK-GROUPING SCHEDULE

Figure 2 presents an overview Architecture of task-grouping schedule model which identifies the major blocks of model. There are task schedule, computational server, storage server, task group and selection resource, information collection, cloud information services, dispatcher, resource.

In architecture of task-grouping schedule model are using basic of grid model and task scheduling. The scheduler accept task for request user with SLA parameter such as service type, size of file and scheduler is classification task user's request based on service type. The services types are two major in cloud computing, there are storage service and computational service. The storage server accepts tasks for storage service where storage server provided data storage and it don't require mapping of services where storage of data doesn't need map resource with task request. The computational server accepts tasks for computational services where provides mapping between tasks and computing resource based SLA parameters. Begin the schedule task grouping by grid model which contain of five basic blocks are task grouping and selection resource, information collection, cloud information service (CIS), dispatcher and resource [7].

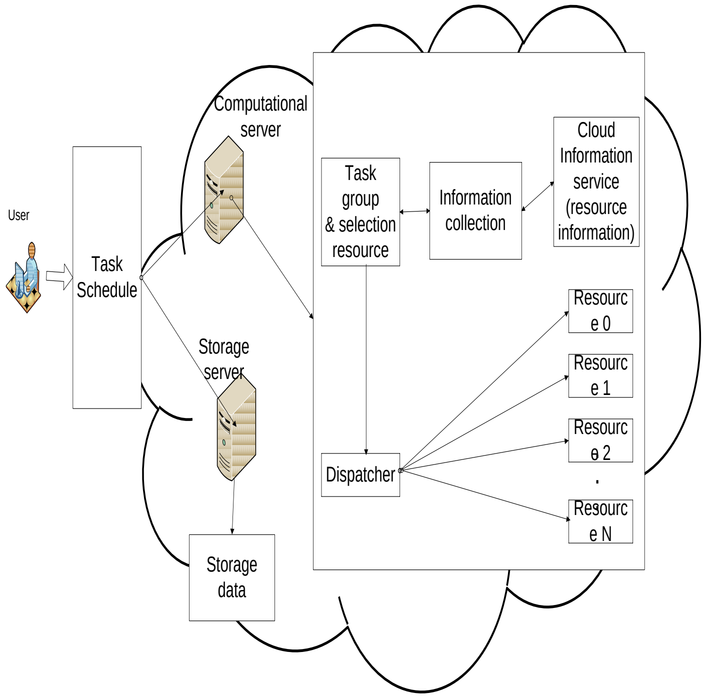


**Fig 2 Architecture of task-grouping schedule model**

The task group and selection resource functions are accepting tasks and group it to list and selection appropriate of computing resource by dispatcher based on information collection where it collected information for resource available from cloud information services. The cloud information services (CIS) is service in cloud computing which provides information about all the registration computing resource. It collects computing resource information such as operating system (windows, Linux), management policy (time share, space share), resource index and processing capability (Million Instructions per Second). In addition, it also provides information to user on the availability of the resources. Information collector collects information from cloud information service. It collects available of the computing resource and get characteristic resource such as processing capability (MIPS) and memory size to each available of resource through cloud information service. The task group and selection resource used for information collector to gather necessary information resource to perform task selection via required for information such as MIPS, memory size. the dispatcher functions as sender where sends grouped task t their appropriate resource based on the schedule model during the mapping of tasks with computing resource and it gathers the results of the processed tasks from the resource. The resource functions execute of group tasks and send to result for user.

Figure 3 presents an overview of block diagram scheduling and grouping task model, which identifies five basic block of this model. There are user request task, task scheduling, task grouping, schedule and execute.

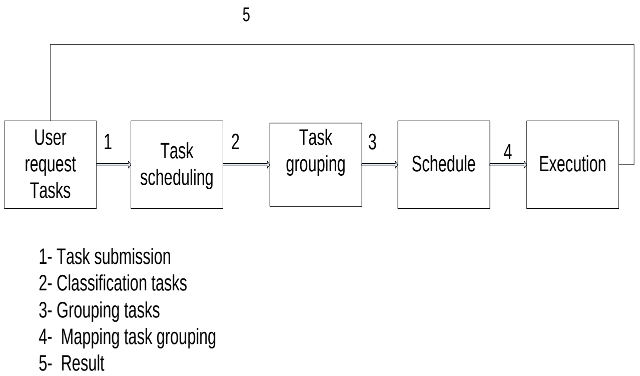


**Fig. 3 Block diagram of task schedule and groups model**

The user request task with SLA parameter sends to task scheduling. The outputs task schedule are submit task for required user or not and classification them then send to task grouping since grouped for tasks and mapping with computing resource based on schedule of group task then execute task and return result for user.

## 4. THE ALGORITHM OF TASK-GROUPING SCHEDULE

The algorithm is divided into two parts; there are task schedule and task grouping. The task schedule algorithm is classification of task depended on service type of task. The second part algorithm is grouping of task and mapping with computing resource.

### 4.1 Task Scheduling

The scheduler submits number of tasks for required of user with SLA parameters. There are service type and file size of tasks. Then tasks are sort based on file size each task ascending order Then classification of tasks based on service type where ask what is service type? If service type is storage service then storage data and send massage to user "the task storage in system". But the service type of tasks is computational service send them to computing resource.

1. Input: request user for tasks with SLA term
2. Task_list(ID,MI,FS) {
3. Sort tasklist based on min FS
4. For (each tasks for user){
5. Get service_type
6. If (service_type is computation) then
7. Insert task$_i$ into tasklist
8. Else
9. Insert task$_i$ into storage device
10. }
11. Send tasklist to task_grouping Algorithm
12. }

### 4.2 Task-Grouping Strategy

Task-grouping strategy is depended on processor capability expressed in the amount of million instructions per second (MIPS) and memory size of the available computing resource the size of grouped task depended on the processing length expressed in million instructions (MI).therefore, the following conditions must be satisfied :

Taskgroup_MI <= Resource_MI ………… (1)
Resource_MI = Resource_MIPS * GS
Taskgroup_FS <= Resource_MS …………. (2)

Where task_group_MI: Million instruction is task's required computational power, resource MIPS: million instruction per second is processing capability of resource, GS: granularity size is used to measure total number of tasks that can be complete within a specific time. It always defined by user, Reource_MI: MI of resource is multiply granularity size with MIPS of resource, task group MS: the total memory size of group tasks, Resource_MS is the storage memory available at resource. The condition one required computational power (taskgroup_MI) of grouped task is less than or equal processing capability of resource. The second condition total memory size of group task is less than or equal memory size of computing resource. These conditions are basic factor in task grouping strategy. If there are not execute this strategy repeated until find appropriate available resources.

Terms used in the algorithm
MI: Million instruction requirements of a user task.
FS: file size requirements of a user task.
MIPS: million instructions per second of resource
SM: Storage memory of resource
GS: granularity size (time in second) for task grouping.

1. Input: TaskList (ID$_j$,MI,FS) and ResourceList(ID$_i$,MIPS, SM)
2. Task_grouping {
3. Set file size of Task_group$_i$ (Fs(Task_group$_i$))equal zero;
4. Set MI ofTask_group equal zero;
5. Get Gs specified by user;
6. Set index for ResourceList is i from 0 to ResoueceList-1;
7. Sort ResourceList based on min MIPS
8. Get MIPS of Resource$_i$ ;
9. Get memory size of Resource$_i$;
10. MI(Resource$_i$) = MIPS(Resource$_i$)* GS;
11. For(j=0;j<=TaskList;j++){
12. MI(Task_group)= MI(Task_group)+MI(Tasklist$_j$);
13. Fs(Task_group)= Fs(Task_group)+FS(Tasklist$_j$);
14. If (MI(Task_group)<= MI(Resource$_i$) && Fs(Task_group)<= SM(Resource$_i$))
15. Send Task$_j$ to resource$_i$
16. Else {
17. I++;
18. Set file size of Task_groupi equal zero;
19. Get MIPS of Resource$_{i+1}$ ;
20. Get memory size of Resource$_{i+1}$;
21. MI(Resource$_{i+1}$) = MIPS(Resource $_{i+1}$)* GS;
22. MI(Task_group)= MI(Task_group)+MI(Tasklist$_{j+1}$);
23. Fs(Task_group)= Fs(Task_group)+FS(Tasklist $_{j+1}$);
24. If (MI(Task_group)<= MI(Resource$_{i+1}$) && Fs(Task_group)<= SM(Resource $_{i+1}$)){
25. Send Taskj to resource $_{i+1}$
26. }
27. }
28. }

## 5. EXPERIMENTAL RESULT

CloudSim3.0 has been used to create the simulation clod computing environment. The inputs to simulations are number of tasks, average MI of task 10, average file size of task 400MB (megabyte) and five second of granularity time. The algorithm simulated with six resources has been created to verify task grouping and scheduling, where each resource has different characteristics such as MIPS and memory size. In this simulation, the MIPS and memory size of resource in table 1.

**Table 1. Characteristics of resource**

| Resource | MIPS | Storage Memory (MB) |
|---|---|---|
| R1 | 120 | 5000 |
| R2 | 131 | 5000 |
| R3 | 153 | 5000 |
| R4 | 296 | 5000 |
| R5 | 126 | 5000 |
| R6 | 210 | 5000 |

Table 2 and Table 3 shows the simulation result of the proposed scheduling group task and scheduling compared to improve ABC for processing time and processing cost for various numbers of cloudlets namely 25, ,50 75, 100.

**Table 2. Simulation processing time for IABC and task grouping schedule algorithm**

| No. of cloudlet | processing Time Task-grouping schedule Algorithm | processing Time IABC Algorithm |
|---|---|---|
| 25 | 56.69 | 133.1 |
| 50 | 120.83 | 234.01 |
| 75 | 166.08 | 378.51 |
| 100 | 221.18 | 487.5 |

**Table 3. Simulation processing cost for IABC and task grouping schedule algorithm**

| No. of cloudlet | processing cost Task-grouping schedule Algorithm | processing Cost IABC Algorithm |
|---|---|---|
| 25 | 35.6 | 72.34 |
| 50 | 71.2 | 374.01 |
| 75 | 106.8 | 402.61 |
| 100 | 106.8 | 543.32 |

Figure 4 depicts the graph of the result collected from the simulations. It compares IABC scheduling algorithm and task grouping scheduling algorithm on the basic of processing time for tasks for the value in Table 2. From the fig. 4 The task grouping and scheduling has reduce the total processing time by 54% - 57% when compare IABC list scheduling depending on the number of cloudlets.

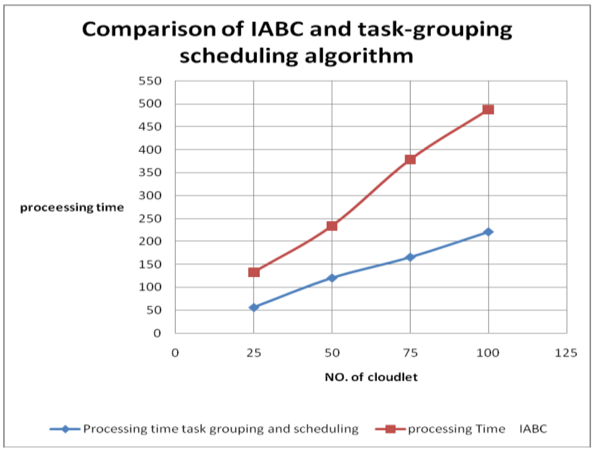


**Fig 4: Comparison of IABC and task-grouping scheduling algorithm for processing time**

Figure 5 depicts the graph of the result collected from the simulations. It compares IABC scheduling algorithm and task grouping scheduling algorithm on the basic of cost for processing of task for the value in Table 3.

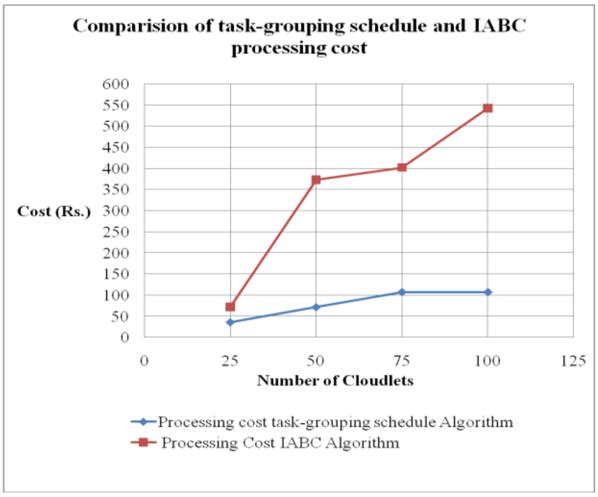


**Fig 5: Comparison of IABC and task-grouping scheduling algorithm for processing cost**

From the figure5 the task grouping and scheduling has reduce the total processing cost by 50.78% - 80.34% when compare IABC list scheduling depending on the number of cloudlets and number of the resource which used for implement of cloudlet.

## 6. CONCLUSION AND FUTURE WORK

This paper discuses task-grouping schedule algorithm in cloud computing environment. In cloud computing, many schedule algorithms are available to solve scheduling and mapping between tasks and resource problem. Task-grouping scheduling strategy aims minimum total tasks completion time. CloudSim3.0 is employed to simulation the tasks test algorithm and distributed task-grouping scheduling. The analysis conducted on the simulation algorithm has shown that the propose algorithm is able to perform task schedule using resource condition. The result has also good comparative result in better task-grouping schedule compare to an IABC algorithm. In our future work, we will include some more SLA parameter for task and resource.

## 7. REFERENCES

[1] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, “Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility, Future Generation Computer Systems”, Elsevier Science, Amsterdam, The Netherlands, 2009 (pp. 599-616).

[2] Fang Liu, Jin Tong, Jian Mao, Robert Bohn, John Messina, Lee Badger and Dawn Leaf, “ NIST Cloud Computing Reference Architecture” Special Publication 500-292 September 2011.

[3] Wei-Tek Tsai*, Xin Sun, Janaka Balasooriya, “Service-Oriented Cloud Computing Architecture”. 2010 Seventh International Conference on Information Technology, IEEE.

[4] Kaiqi Xiong, Harry Perros, “Service Performance and Analysis in Cloud Computing”, 2009 IEEE.

[5] R.N. Calheiros, R. Ranjan, A. Beloglazov, C.A.F. De Rose, R. Buyya, Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms, Software: Practice and Experience 41 (1) (2011) 23–50.

[6] V.Soni, R.Sharma & M.Mishra, “Grouping-Based Job Scheduling Model In Grid Computing ”, World Academy of Science, Engineering and Technology Publishing, Vol. 4(1),issue 5 , 2010, pp: 781-784.

[7] Ng Wai Keat, Ang Tan Fong, “Scheduling Framework For Bandwidth-Aware Job Grouping-Based Scheduling In Grid Computing”, Malaysian Journal of Computer Science, Vol. 19, No. 2, pp. 117-126, 2006.

[8] N. Muthuvelu, Junyan Liu, N.L.Soe, S.venugopal, A.Sulistio, and R.Buyya, “A dynamic job grouping-based scheduling for deploying applications with fine-grained tasks on global grids,” in Proc of Australasian workshop on grid computing, vol. 4, pp. 41–48, 2005.

[9] Mrs.S.Selvarani, Dr.G.Sudha Sadhasivam, “Improved Cost-Based Algorithm For Task Scheduling In Cloud Computing” IEEE, 2010.