Workflow Scheduling Approach Based on Makespan in Cloud Computing

Sonam Seth*, Nipur Singh
Department of Computer Scinece,
Kanya Gurukul Campus, Dehradun, Uttarakhand, India
*sethsonam@gmail.com, nipur@gkc.ac.in

Received: *29.11.2018*, ***Accepted:*** *26.12.2018*

Abstract

In recent computational trend cloud computing has grown up rapidly as utility. With the advancement of technology cloud computing has new challenge in computing of dynamic user requests. The resources are not so wide, so the tasks are not inter-dependent, in this scenario utilization of resources is not efficient and therefore makespan increases. Existing task scheduling methods do not give the efficient results so to overcome this challenge workflow scheduling method is required. Workflow scheduling manages the workflow execution. In this paper makespan based workflow scheduling method for cloud computing systems is proposed. Workflow scheduling is used to map and manage inter-dependent tasks on distributed resources. Simulation of proposed method is done on workflowsim.

Keywords- Cloud Computing, Makespan, Workflow management, Max-Min, Scheduling.

Introduction

Cloud computing is a widely used technology in computing paradigm where applications, data and IT services are provided over the Internet. The extension of cloud is on-demand service, self-service and pay-per-use nature (Singh *et al.*, 2013). The Cloud computing is able to provide various types of services like Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) etc. to end users. This is just because of new approaches in cloud computing. It gets difficulties from various different types of issues like security, performance, database management, virtual machine migration, workflow scheduling etc. Scheduling of workflow is the major issue in the workflow applications only.

The scientific analysis is computation and data intensive and natural to takes is for a long execution time. In other aspects, the data intensive play an important role in the web environment as well as in scientific applications (Guo *et al.*, 2012). It is found that the workflow scheduling is an interesting area of research in distributed systems. The most important research area in workflow scheduling can be the scientific research, health care, social networking, education, surveillances and business. These issues can contain a wide variety of large-scale applications. The execution of complex workflow in the distributed system requires both computation cost as well as communication costs (Tharsanee *et al.*, 2017).

Cloud service provider offers heterogeneous resources of different capabilities at different prices. The scheduling plan of workflow may result in various makespan, that may in turn in to more cost for each execution. Thus, Makespan is an very important and essential Quality of Service (QoS) parameter that requires to be considered for effective cloud workflow scheduling (Tharsanee *et al.*, 2017). Usually the Directed Acyclic Graphs (DAGs) represents the workflows. In many well-known situations, where the DAG is arbitrary, most problems related to time and/or latency minimization are NP-hard (Agrawal *et al.*, 2010).

The Workflow Management System

The workflow can be defined as the identification of interfaces within the structure that enables the products to interoperate at a different variation of the levels. The reference model of workflow is given below:

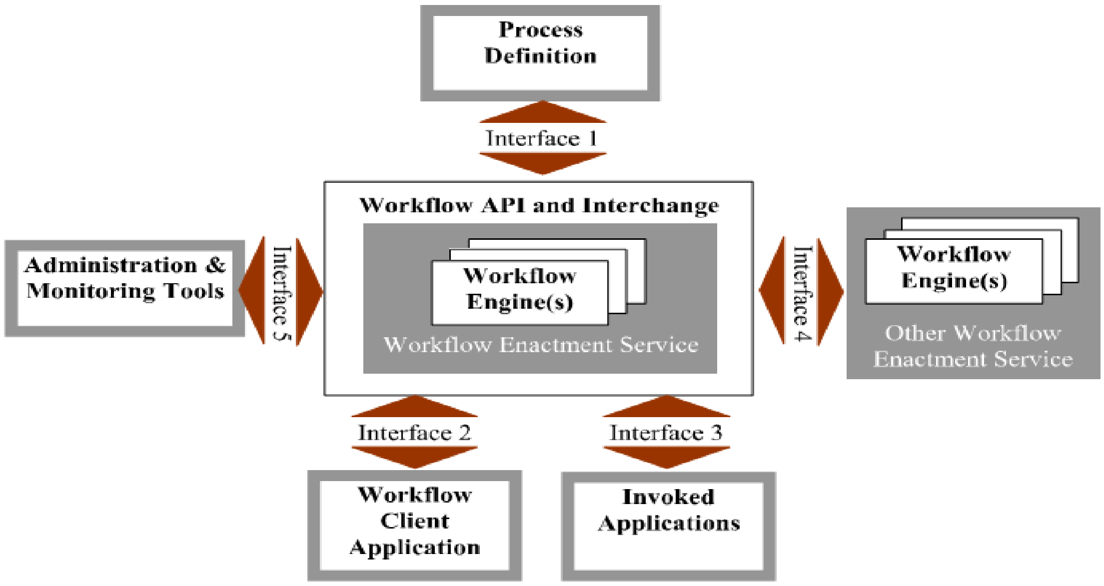

Figure 1: Workflow Reference Model (Agrawal *et al.*, 2010)

Workflow Engine: Workflow Engine is a software application that represents and acts like Data Center Broker. It manages Virtual Machine (VM) creation, management and submission of cloudlets to VM, and also destroys VM after its use.

Process Definition: This is the representation of process of workflow in the form which supports automated manipulation (Sharma *et al.,* 2017).

Workflow Interoperability: This shows the interfaces to support interoperability between various workflow systems. We can call is workflow parser that parses a DAX into task so that workflow-Sim can manage it.

Invoked Application: This application has the feature to support interaction with a variety of IT applications (Sharma *et al.,* 2017).

Workflow Client Application: This application has the interfaces to support interaction with the user (Sharma *et al.,* 2017).

Administration and Monitoring: This will provides the monitoring and observation of system and metric functions to organize the overall management of composite workflow application environments (Sharma *et al.,* 2017).

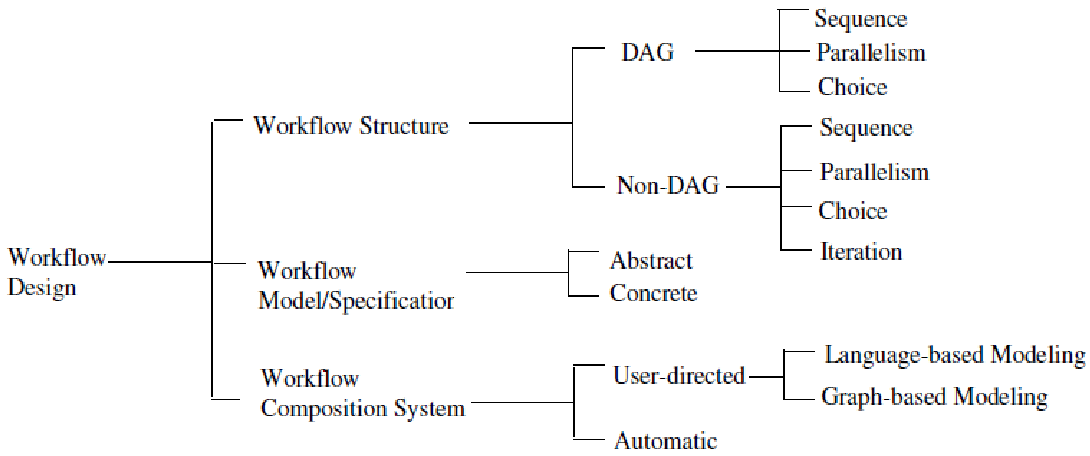

Figure 2: Workflow Design Taxonomy (Singh *et al.,* 2013)

Workflow Scheduling

Workflow applications are represented as Directed Acyclic Graphs (DAGs) with precedence constrained tasks which are inter-dependent on each other. The each tasks in the workflow will have some output and input files which is based on the specific application. The time taken for the input and output file transfer will also impact the overall execution time of the workflow. Therefore it is essential to consider the size of the input and output files of the task under execution and the bandwidth that the virtual machine offers for the data transfer (Tharsanee *et al.,* 2017). The management and mapping of task's execution workflow on shared resources is done with the help of workflow scheduling. We hereby concentrate on minimizing execution time of a workflow application.

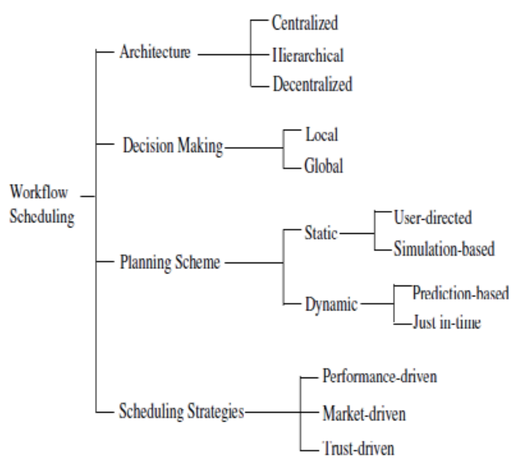

Figure 3: Taxonomy of Workflow Scheduling (Yu *et al.,* 2005)

Workflow scheduling is a challenging task in cloud computing, as the resources are central which fulfils the need of all type of jobs. Therefore, it is difficult to predict which resources will be available at the time of actual execution of the jobs. It is difficult to achieve maximum possible utilization of resources because of dependencies, and different load and resource requirements among different levels (Singh *et al.,* 2013).

Max-Min Workflow scheduling

This algorithm overcomes the drawback of the min-min (Sharma *et al.,* 2017) algorithm. The steps performed by Max-Min algorithm almost same as the Min-Min algorithm did. The main difference in steps reflects only in the second phase, where a task with maximum completion time is selected as in min-min and has been assigned to resource R that provides the minimum completion time. Therefore, we can call this algorithm as Max-Min algorithm. These steps are repeated until the metatask get empty or all the tasks are mapped.

Now, here we representing a workflow application as a DAG which is represented by G=(V,E), where V ={T1, ..., Tn} is the set of tasks, and E represents the data dependencies between these tasks, that is, fj,k =(Tj, Tk) -> E is the data produced by Tj and consumed by Tk. We have a set of virtual resources as VM = {1, ..., j}, a set of compute sites PE = {1, ..., r}, and a set of tasks T = {1, ..., n}.

The proposed MMW scheduling method is given below:

Algorithm 1: Max-Min Workflow Scheduling (MMW)
Input: Workflow in DAX
Output: Minimum completion time

Phase 1

Calculation of minimum completion time of each task

```
for i = 1 to n        //all task in metatask
for j = 1 to m        //all resources
CTij = ETij + RT //compute completion time
End j
End i
```

Phase 2

Find task with maximum completion time and assign to resource with minimum execution time

```
do
for each task in metatask list (ML) with maximum completion time and map on to resource
assign all task from metatask to resource with minimum completion time
remove Ti from metatask list
update resource Rj ready time RTj
update completion time of unassigned task in metatask list
while(ML==Null)
End
```

The main aim of max-min scheduling algorithm is to reduce the waiting time of large size jobs. In this algorithm, small size tasks are concurrently executed with large size tasks, hence reducing the makespan and gives better resource utilization.

Results and Discussion

The proposed workflow scheduling approach has been implemented on WorkflowSim by programmatically extending the core framework provided in the WorkflowSim. The basic configuration is given in following tables.

Table 1: VM Configuration

Parameter	Value
Size	10000
RAM	512
MIPS	1000
BW	1000
Number of CPU	1
Vmm	Xen

Table 2: Host Configuration

Parameter	Value
RAM	2048
Storage	1000000
BW	10000

Table 3: Workflow Makespan Result Analysis

Case Number	Number of Task	Number of VM	Makespan
1: Montage_25	25	5	56.95
2: Montage_50	50	5	129.42
3: Montage_100	100	5	256.44

After implementation of the MMW scheduling algorithm for different workflows their makespan has been calculated. Value of makespan for all cases has been calculated. As shown in table case 1 shows the makespan of workflow with 25 cloudlets and 5 VM is 56.95, in case 2 makespan of a workflow with 50 cloudlets and 5 VM is 129.42, similarly makespan of 100 cloudlets with 5 VM is 256.44 in case 3. Hence result table shows that as complexity of workflow increases in terms of task node in DAG their completion time is also increases.

Conclusion and Future Work

The main objective of the proposed work is to minimize the overall completion time of the workflow but no consideration was given on the resource utilization. It has been analyzed that proposed scheme is effective enough to optimally use the resources. The suggested method is implemented to simulate and extract results by using the WorkflowSim simulator. Due to static nature of resources resource allocation is not effectively achieved to minimize the makespan. There is need of dynamic policy to reduce makespan as size and complexity of workflow increases.

Acknowledgement

I owe my deep and sincere gratitude to Dr. Nipur Singh, Professor, Department of Computer Science, Kanya Gurukul Campus, Dehradun, India for her enlightened guidance, continuous encouragement, and esteemed supervision throughout the period of this research.

References

Agrawal, K., Benoit, A., Magnan, L., Robert, Y. 2010. Scheduling Algorithms for Linear Workflow Optimization, Parellel and Distributed Processing (IPDPS) *IEEE International Symposium*, Atlanta, GA, USA, pp.1-12.

Guo, L., Zhao, S., Shen, S., Jiang, C. 2012. Task Scheduling Optimization in Cloud Computing Based on Heuristic Algorithm", *Journal of Networks*, 7(3), 547-553.

Kaur, N. Aulakh, T.S. Cheema, R.S. 2011 Comparison of Workflow Scheduling Algorithms in Cloud Computing, *International Journal of Advanced Computer Science and Applications (IJACSA)*, 2(10), 81-86.

Sharma, N. Tyagi, S., Atri, S. 2017. A Comparative Analysis of Min-Min and Max-Min Algorithms based on the Makespan Parameter, *International Journal of Advanced Research in Computer Science*, 8(3),1038-1041.

Singh, L., Singh, S. 2013. A Survey of Workflow Scheduling Algorithms and Research Issues", *International Journal of Computer Applications*, 74(15), 21-28.

Singh, R. Pateria, P. K. 2013. Workflow Scheduling in Cloud Computing, *International Journal of Computer Applications*, 61(11),38-40.

Tharsanee, R..M. Mohanapriya, N. 2017. Minimum Makespan Cloud Workflow Scheduling Algorithm for Scientific Workflows, *International Journal of Innovative Research in Engineering Science and Technology*, 5(1), 89-92.

Yu, J., and Buyya, R. 2005. A taxonomy of scientific workflow systems for grid computing, *SIGMOD Record*, 34(3), 44-49.