ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

IARJSET

International Advanced Research Journal in Science, Engineering and Technology

National Conference on Emerging Trends in Engineering and Technology (NCETET'16)

Lourdes Matha College of Science & Technology, Thiruvananthapuram

Vol. 3, Special Issue 3, August 2016

Improved Scheduling of Scientific Workflows Using HDPSO

Merly Mathew[1], Jayalekshmi S[2]

PG Scholar, Department of CSE, LBSITW, Trivandrum, India [1]

Associate Professor, Department of CSE, LBSITW, Trivandrum, India [2]

Abstract: In cloud computing, computing infrastructure is viewed as cloud, where all the individuals and business firms keep their data and access it from anywhere in the world on demand. It is the latest distributed computer paradigm after grid, utilities where everything is based on pay- per use. Workflow and task scheduling are the recent research topics in cloud. Many scheduling policies have been defined to maximise the amount of work, but however, many of them are not optimal. In this paper a meta-heuristic optimization technique, HDPSO is used to minimize the execution cost simultaneously meeting the deadline.

Keywords: Cloud Computing, HDPSO, Scheduling, Workflow.

I. INTRODUCTION

Cloud computing which is developed from grid, distributed and parallel computing have tasks distributed on resource pool having resources like computers, storage devices, CPU. They are provided in an on- demand fashion through internet. It is document- centric and not PC-centric, where the document only matters and not the PC used to access document. The cloud has several deployment models like public, private, community and hybrid [1]. In public, the cloud infrastructure is open for public and exists on the premise of cloud provider. The private may be owned, operated by a person, organization and exist in on or off the premises. However in community, the cloud infrastructure is for exclusive use by a shared community of users. The hybrid cloud is composition of more than two cloud infrastructures (public, private, community).

The two entities in cloud computing are the infrastructure providers and service providers, where the infrastructure providers manage cloud platforms and lease resources based on their usage. The service providers rent resources from infrastructure providers to serve the end users. So the customers and cloud providers enter into an agreement called Service Level Agreement (SLA). It clarifies the roles, set charges and expectations and also provides mechanisms for resolving service named problems within a specified time period. It also covers performance, reliability conditions with respect to Quality of Service (QoS).

The three service models in cloud computing are Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). Using SaaS, a single application can be delivered to thousands of users. The users access the application via an API (Application Programming Interface). In PaaS and IaaS, the development environment and infrastructure is offered as a service respectively.

Scheduling is the process of finding out the appropriate resources to allocate to the tasks or jobs. It is done effectively by considering QoS constraints such as budget, deadline and high throughput. Virtualization is a concept used for providing the tasks a minimum completion time, better performance, resource utilization and quick response time. These virtual machines are scalable but scheduling them is a major problem in task allocation. Task scheduling is an important issue which greatly influences the performance of cloud computing environment. Scheduling can be independent or dependent based on the dependency of jobs [2]. In independent, which is also called task scheduling, each task execute independently. However, in dependent, which is called workflow scheduling, there is a dependency graph between the tasks.

In this paper, resource provisioning is based on a variant of meta-heuristic optimization technique, Particle Swarm Optimization (PSO) named as Hybrid Discrete Particle Swarm Optimization (HDPSO). Inspired on the social behavior of bird flocks, Kennedy and Ebehart [3] introduced the technique PSO. It is based on swarm of particles moving through space and communicating to find the optimal search direction. PSO has better performance than other algorithms. It is easier to implement since it has only few parameters to tune with. In this paper, a cost minimized deadline constrained scheduling technique is used in cloud environment considering the heterogeneity of virtual machines.

II. RELATED WORK

The workflow scheduling algorithms are of two types: – Heuristic and Meta-Heuristic. Heuristic algorithms are based on priority where the user can use his knowledge to allocate priority for cloud resources and workflow

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
National Conference on Emerging Trends in Engineering and Technology (NCETET'16)
Lourdes Matha College of Science & Technology, Thiruvananthapuram
Vol. 3, Special Issue 3, August 2016

applications. However, the Meta Heuristic algorithms do not need human interface. They provide a solution to workflow applications which are near optimal. The examples of meta heuristic algorithms are Ant Colony Optimization (ACO), Genetic Algorithms (GA) and Particle Swarm Optimization (PSO).

In 2009, W. N. Chen et al. proposed the Ant Colony Optimization (ACO) [4], based on how ants find a path between their colony and the source of food. The ants are generated and mapping is done with the path and the objective function to be evaluated. The user can specify the QoS parameters while submitting the workflow application, preferring and optimizing them. ACO finds a schedule that meets all user imposed QoS constraints like deadline, budget and reliability. In ACO, the ants keep record of each and every node that they visit and record that data for future decision making. As a result they deposit pheromones during their movement for other ants to select the next nodes. Each ant works independently and represents a virtual machine looking for a host to get allocated.

In 2006, J. Yu et al. [5] suggested Genetic Algorithms, (GAs) applying the principle of evolution. It generates a high quality solution which is derived from a large search space in polynomial time. Any solution in the search space of the problem is represented by an individual (chromosomes). It maintains a population of individuals that evolves over generations. The quality of an individual in the population is determined by a fitness function. The fitness value indicates how good the individual is compared to others in the population. In GA, an initial population is created consisting of random solutions. New offsprings are then generated by applying genetic operators like selection, crossover and mutation. Fitness of each individual in the population is evaluated and repeated.

In 2010, S. Pandey et al. proposed a Particle Swarm Optimization (PSO) heuristic for scheduling workflow applications in cloud [6]. This algorithm was developed by Dr. Eberhart and Dr. Kennedy in 1995. It considers both computation cost and data transmission cost and workloads are distributed with minimal cost. It mainly considers resource utilization and time as the main parameters. The particle in PSO is generally the workflow and its tasks. The dimension is the number of tasks in the workflow and the moving range of particle is the number of resources in the resource pool. The fitness function will be the total execution cost of the schedule. In this PSO based algorithm, a particle is represented by its position and velocity. Each particle has a best position, pbest and a global best solution, gbest. The particles fitness value will be compared with pbest. If the current value is better than pbest, update pbest to that current value and location. Similarly compare the particles fitness value with gbest and if current value is better, then update gbest to that current value and location. This PSO algorithm is simple and effective for applications with low computational cost like data mining, pattern recognition, environmental engineering etc.

In 2010, Z.Wu et al. proposed the Revised Discrete Particle Swarm Optimization (RDPSO) [7], which schedule applications among cloud services considering both computation cost and data transmission cost. It achieves better performance on makespan and cost optimization. The PSO algorithms usually give a better performance as it considers the dependencies between cost and tasks. In RDPSO, a set based concept is introduced into PSO, where each task is mapped onto a set of services. Also due to the discrete property of scheduling, the gbest will only have a few values to select from.

In 2014, M.A. Rodriguez et al developed a scheduling algorithm based on the meta-heuristic optimization technique, Particle Swarm Optimization (PSO) [8]. It aims to minimize the overall workflow execution cost while meeting deadline constraints on an IaaS cloud environment. Here the IaaS cloud features like pay-as-you-go model, heterogeneity, elasticity and dynamicity of resources are considered. Usually it performs better than the current algorithms considering cost and deadline as the main parameters.

III. PROPOSED APPROACH

A workflow is represented by a DAG (Directed Acyclic Graph), G = (T, E) where T represents the tasks T= {t_1, t_2, ..., t_n} and E represents the data dependencies among the tasks. In workflow scheduling, a large task is divided into different subtasks, where each are allocated to resources to achieve a predefined objective. A sample workflow is as in Figure 1. Each node represents the tasks and the directed edges represent the data transfer time between the tasks.

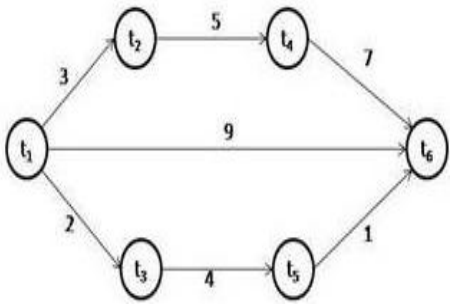

Figure 1 A sample workflow

A. Problem Definition

In this, the total execution cost (TEC) may include both execution cost (EC) and transfer cost (TC).

EC = costpersec * actual CPU time

TC = costperBW * filesize

TEC = EC + TC

Here ensuring should be done that Total Execution Time (TET) does not exceed the deadline, δ as the main aim is to minimize the execution cost meeting the deadline. This is represented as Equation.

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
National Conference on Emerging Trends in Engineering and Technology (NCETET'16)
Lourdes Matha College of Science & Technology, Thiruvananthapuram
Vol. 3, Special Issue 3, August 2016

Minimize TEC
subject to TET $\leq \delta$

B. Resource Provisioning
Particle Swarm Optimization is an evolutionary technique based on behavior of animal flocks (e.g. fish or bird). A particle represents an individual moving through search space and is represented by velocity at a particular point. The velocity is determined by the best position the particle is in so far (pbest) and the best position in which any particle is in (gbest). The fitness function describes the quality of particles position.

Each particle is represented by its position and velocity. Particles keep track of its best position (pbest) and global best solution (gbest) and change values towards the pbest and gbest values. The algorithm iterate until the stopping criterion, which is commonly either maximum number of iterations or predefined fitness value. The pseudo code for the PSO algorithm is shown as in Algorithm 1. In each iteration, the particle updates its position and velocity according to the equations respectively.

$$\vec{x_i}(t+1) = \vec{x_i}(t) + \vec{v_i}(t)$$

$$\vec{v_i}(t+1) = \omega . \vec{v_i}(t) + c_1 r_1 (\vec{x_i^*}(t) - \vec{x_i}(t)) + c_2 r_2 (\vec{x^*}(t) - \vec{x_i}(t))$$

where:
ω = inertia,
c_i = acceleration coefficients, i=1,2
r_i = random number, i=1,2 and $r_i \in [0,1]$
$\vec{x_i^*}$ = best position of particle i
$\vec{x^*}$ = position of best particle in population
$\vec{x_i}$ = current position of particle i
Parameter c_1 is called cognitive parameter as it defines the previous best position and c_2 is called social parameter as it is relative to other neighbors.

Algorithm 1 Particle Swarm Optimization
1: Set the dimension of particle as d
2: Initialize the particle's population with random position and velocities.
3: for each particle, calculate its fitness value do
4: Compare the particle's fitness value with the particle's pbest. If the current value is better than pbest, then set pbest to the current value and location.
5: Compare the particle's fitness value with the global best gbest. If the particle's current value is better than gbest, then set gbest to the current value and location.
6: Update the position and velocity according to the equations.
7: end for
8: Repeat from step 3 until the stopping criterion is met.

On defining the meaning and dimension of particle, the particle represents workflow and its tasks and hence the dimension of particle defines the number of tasks in the workflow. In the proposed approach, HDPSO [9] is used instead of PSO for generating the schedule. HDPSO is the hybrid combination of Min-Min and DPSO. The pseudo code for HDPSO is as in Algorithm 2.

Algorithm 2 Hybrid Discrete Particle Swarm Optimization
1: Generate initial population using Min-Min.
2: Apply fitness function and evaluate each particle in initial population.
3: Find out best position of each particle and global best position of particles, pbest and gbest respectively for the initial population.
4: Update the position and velocity according to the equations.
5: Repeat until the stopping criterion is met i.e, maximum number of iterations.

In this, the fitness function is the makespan, i.e, completion time of last task. The execution time of each machine is added and the maximum value represents the makespan.
Makespan = Max $\{F_i\}$, where F_i is finish time of last task.

Both PSO and HDPSO are then evaluated under homogeneous and heterogeneous environments of virtual machines. In homogeneous, there is only one virtual machine and in heterogeneous, there are more than one virtual machine.

IV.EXPERIMENTAL EVALUATION

A. Implementation Details
The evaluation is done in a simulated environment using Workflow Simulator [10] with Java NetBeans IDE 8.1. Java is a high-level object-oriented programming language which is platform independent and simplified to eliminate features that cause common programming errors. NetBeans is a commonly used Integrated Development Environment (IDE) for Java.

B. Results and Analysis
For performance analysis, the workflows mainly Montage, CyberShake and Sipht are considered. The fitness function generally measure how good or bad the position is. The fitness function usually varies from problem to problem. The effect of fitness value with increase in iterations is as in Figure 2.

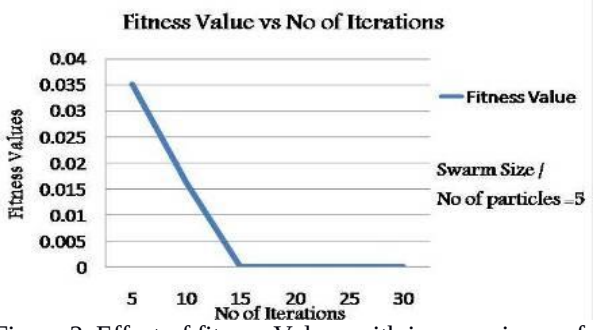

Figure 2. Effect of fitness Values with increase in no of iterations

IARJSET
ISSN (Online) 2393-8021
ISSN (Print) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
National Conference on Emerging Trends in Engineering and Technology (NCETET'16)
Lourdes Matha College of Science & Technology, Thiruvananthapuram
Vol. 3, Special Issue 3, August 2016

Here the population size is taken as 5. The figure shows that as the number of iterations increases from 5 to 30, the fitness value reduces.

The algorithms PSO and HDPSO are considered in both homogeneous and heterogeneous environment of virtual machines. In homogeneous environment, there is only one virtual machine, however in heterogeneous environment, 5 virtual machines are considered. Different parameters like deadline, execution cost are also considered.

Cost Evaluation

The effect of execution cost on the workflows namely CyberShake, Montage and Sipht is as in Figure 3. From the Figure 3, it is clear that HDPSO has reduced execution cost in both homogeneous and heterogeneous environment.

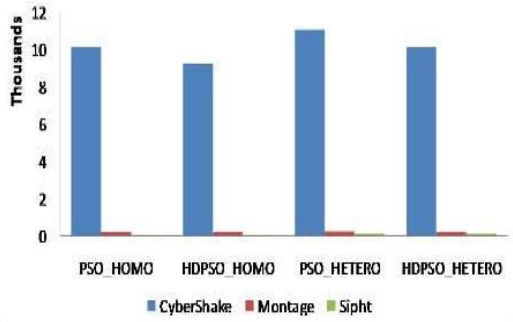

Figure 3 Effect of execution cost

Deadline Constraint Evaluation

The evaluation is conducted using four different deadlines. The value of deadline lies between the slowest and fastest runtimes. The equation used for calculating the deadline is as below.

$Deadline_i$ = time(fastest) + k *(time(slowest)-time(fastest))

where :

k = 0.2, 0.4, 0.6, 0.8

i= 1, 2, 3, 4

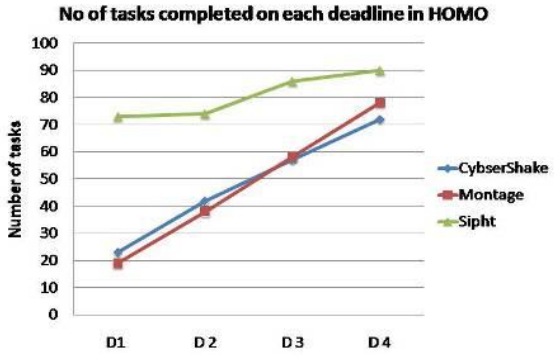

Figure 4. No of tasks completed for HOMO

In homogeneous environment, the different workflows namely CyberShake, Montage and Sipht complete different number of tasks in different deadline intervals. The no of tasks completed in each deadline interval by these workflows is shown in Figure 4. From the figure, it is clear that Sipht completes more number of tasks in each deadline. In intervals D1 and D2 CyberShake completes more number of tasks than Montage, however in D3, both of them completes about same number of tasks. Again in D4, Montage complete more number of tasks than CyberShake.

In heterogeneous environment, number of tasks completed by each of the workflows in different deadline intervals varies. The Figure 5 and 6 plots the number of tasks completed by the considered workflows in each interval of PSO HETERO and HDPSO HETERO. The Figure 5 shows that in each of the deadline intervals, the number of tasks completed by Sipht, Montage and CyberShake is more in order. From the Figure 6, Montage has lowest number of tasks completed in each deadline interval. In interval D1, Sipht completes more tasks than CyberShake while in D2, the reverse takes place. Both these workflows complete about the same number of tasks in intervals D3 and D4.

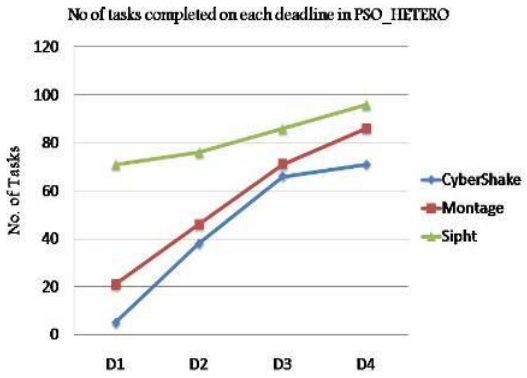

Figure 5. No of tasks completed for PSO_HETERO

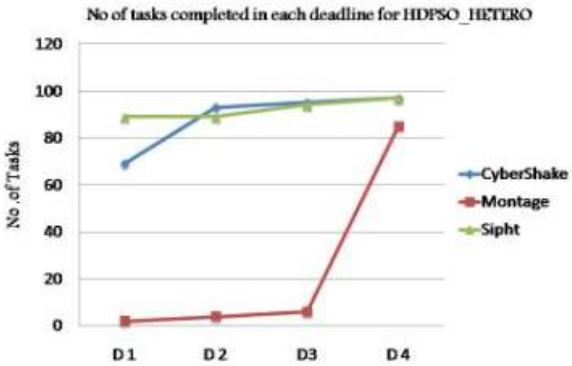

Figure 6. No of tasks completed for HDPSO_HETERO

V. CONCLUSION

Cloud computing has execution cost as an important factor as it is based on pay per use. In this paper, a combined Resource provisioning and Scheduling (RS) method is used for executing the scientific workflows. It also considers various principles like pay-as you go model, elasticity and heterogeneity of the resources. Out of all workflow scheduling algorithms, PSO is used because it has faster convergence, fewer parameters to tune and

IARJSET

ISSN (Online) 2393-8021
ISSN (Print) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

National Conference on Emerging Trends in Engineering and Technology (NCETET'16)

Lourdes Matha College of Science & Technology, Thiruvananthapuram

Vol. 3, Special Issue 3, August 2016

easier to implement. HDPSO is a hybrid of DPSO and Min-Min to overcome the local search capability of PSO. The experimental results show that the use of HDPSO instead of meta-heuristic optimization technique, PSO in resource provisioning helps to minimize the execution cost while meeting the deadline. Also HDPSO has lesser makespan than PSO in heterogeneous environment of virtual machines which makes HDPSO to have better performance than PSO. In future, any other method for further reducing the execution cost can be done. Also it can be used for deploying application in real life environments and can be implemented in real cloud as this is done in a simulated environment.

REFERENCES

[1] P. Mell, T. Grance, "The NIST definition of cloud computing — recommendations of the National Institute of Standards and Technology" Special Publication 800-145, NIST, Gaithersburg, 2011.

[2] Anterpreet Kaur, "A Review of Workflow Scheduling in Cloud Computing Environment", International Journal of Computer Science Engineering (IJCSE), Vol. 4, No.02, pp. 47-50, March 2015.

[3] J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proc. 6th IEEE Int. Conf. Neural Netw., 1995, pp. 1942–1948.

[4] Wei Neng Chena and Jun Zhang," An Ant Colony Optimization Approach to a Grid Workflow Scheduling Problem with Various QoS Requirements," IEEE Transactions on System, Man and Cybernetics, Applications and Reviews, Vol 39, No 1, pp.29-43, 2009.

[5] J. Yu and R Buyya," A budget constrained scheduling of workflow applications on utility grids using genetic algorithms," Proc. 1st Workshop Workflows Support Large-Scale Sci., pp. 01-10 , 2006.

[6] S.Pandey ,L.Wu, S.M Guru, R.Buyya, " A Particle Swarm Optimization based Heuristic for Scheduling Workflow Applications in Cloud Computing Environments", 24th IEEE International Conference on Advanced Information Networking and Applications, pp. 400-407,2010.

[7] Z. Wu, Z. Ni, L. Gu, "A Revised Discrete Particle Swarm Optimization for Cloud Workflow Scheduling," Computational Intelligence and Security (CIS), pp. 184-188, 2010.

[8] Maria Alejandra Rodriguez and Rajkumar Buyya," Deadline based Resource Provisioning and Scheduling Algorithm for Scientific Workflows on Clouds," IEEE Transactions on Cloud Computing, Vol. 2, No. 2, pp.222-235, 2014.

[9] Purnima Devi, Mala kalra, "Workflow Scheduling using Hybrid Discrete Particle Swarm Optimization (HDPSO) in Cloud Computing Environment", International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE) , Vol 3, Issue 12, pp.12301-12307, December 2015.

[10] W.Chen, E.Deelman, "WorkflowSim: A Toolkit for Simulating Scientific Workflows in Distributed Environments", IEEE 8th International Conference on E-Sciences (e-Sciences), pp.1-8, Oct 2012.