

Analysis on Cost and Performance Optimization in Cloud Scheduling

V. Gayathri

Department of CSE
Angel College of Engineering and
Technology, Tirupur,
Tamil Nadu

S. Selvi

Department of CSE
Angel College of Engineering and
Technology, Tirupur,
Tamil Nadu

Dr. B. Kalaavathi

Department of CSE
K.S.R. Institute for Engineering
Technology, Tiruchengode,
Tamil Nadu

Abstract— Cloud computing is a technology that makes use of the internet for providing software or other IT services on demand to the users. Since cloud computing is a *pay as you go* pricing scheme, the cost and performance optimization become the most important issue during scheduling in cloud. Various optimization techniques have been proposed and implemented to find an effective solution for these issues. This paper presents an analysis on cost and performance optimization techniques along with their features. This paper concludes with a comparison of these techniques which helps to find improved optimization algorithms for the readers.

Keywords— Cloud Computing, Scheduling, Optimization, Workflows, Resource capacity, Resource provisioning, Deadline, Virtualization, Virtual Machine, Auto-scaling, Consolidation, Mutation.

I. INTRODUCTION

Cloud computing is a technology that make use of the Internet to obtain software or other IT services on demand. This on-demand computing power and storage capacity makes cloud as an efficient computing platform. Cloud computing provides the shared resources needed by the users at any given time by pay as they go pricing scheme and keeping cost to the user down. Cloud computing is a technology and business model as well as where the providers are providing the software, hardware, platform, or storage providers, deliver their offerings over the Internet. The data application of providers are maintained by the central remote servers by using internet. The applications are used without installation and the personal files which can be located at different location form user can be access through internet. Cloud computing is not a new technology but it leverages the advantages of existing and increases the efficiency. When compared to grid computing which coordinates the networked resources as the distributed computing paradigm, the cloud computing satisfies this distributed resources objectives like computing power, software, storage services, and platforms by virtualization technologies which helps user to complete the jobs in less time and cost. When compared to utility computing which provides on demand resources and charge based on the usage, the cloud computing perceived this utility based scheme by increasing the resource utilization and decrease the operating cost. Virtualization is the key feature that acts as the foundation for providing the resources to users on demand. The commonly used virtualized server called virtual machine

(VM) helps for assigning and reassigning the resources on demand. These services can be provides by cloud from anywhere and at any time through internet. The cloud computing technologies and virtualization growth requires the virtual machines for number of jobs. Since cloud computing becomes the emerging and attractive technology, many user deploy their application in the cloud or extend their home clusters during high demand results in workflow management system. The workflow management system should balance both performance and cost. Different types of scheduling optimization approaches have been developed to reduce the cost and increase the performance. In this paper the cost based task scheduling and dynamically optimized resource allocation strategies has been surveyed.

II. EXISTING SCHEDULING ALGORITHM ON PERFORMANCE AND COST OPTIMIZATION

A. Adaptive Dual Objective Scheduling Algorithm(ADOS)

The basic objective of this effort is to minimize the completion time of the task. However it may causes the redundant usage of resources, which results in the novel scheduling algorithm for the solution. This algorithm gives better result in completion time and also in resource usages. Due to dynamic changes in the resource, the performance fluctuation may occur which can be managed by rescheduling. The operation starts with the initial random schedule by branch and bound technique and a genetic operator which consists of point and swap mutation. The makespan and resource usage has given equal importance including the rescheduling which increases the feasibility of the ADOS algorithm[1]. The scheduling problem is addressed as scheduling of mutually dependent jobs with workflow applications onto a set of heterogeneous hosts. This scheduling process mainly focuses on finding appropriate job that matches to reduces the makespan or schedule length. If the matches are better than the original then it is selected and produced good quality schedule on small resource usage. The loosened schedule selection scheme has been included to produce an effective solution which reduces the makespan significantly. The algorithm randomly selects the mutation method to essentially select different schedules on probability 0.5. The mutation occurs only during the rescheduling process. The mutated schedules are used as current schedule which continues best in the next iteration. The dual objective approach has been executed by achieving static heuristic

scheduling and dynamic rescheduling technique. The exhaustive performance has been evaluated which results in providing quality schedules against performance fluctuation.

B. Partition balanced time scheduling algorithm(PBTS)

The important challenge in integrating workflow with resource provisioning is to find the appropriate amount of resources needed to execute the task and reduce the cost from perspective of users and increase the resource utilization from perspective of resource provider. The algorithm evaluated the minimum number of computing hosts for executing the workflow within the user specified finish time. The resource capacity is the amount of resource request by workflow to provisioning system. When resource capacity is high the make span reduces but results in low resource utilization and high cost. When resource capacity is low, the execution time increases. The first heuristic algorithm called Balanced Time Scheduling (BTS) which evaluate minimum resources needed to execute workflow within specified completion time. Since BTS is static for complete workflow execution, then the extended polynomial algorithm called Partition Balanced Time Scheduling (PBTS)[2] which estimates the best number of resources part time charge unit. This process reduces the gross cost during the application lifetime. PBTS is the scalable approach which can handle workflow with date-parallel tasks and MPI like: Parallel tasks whose sub-tasks are executed concurrently on various resources .PBTS utilizes the elasticity of resources and gives the execution plan with lower cost. PBTS is dynamic which adjust the resource capacity to lower cost and completed within specified time.

C. Auto scaling Mechanism

The auto scaling mechanism[3] faces many challenges regarding performance and cost. Additional virtual machine needs can be acquired any time but it take time to use those VMs. As a result the cloud providers and Third Party Auditor provides schedule-based and rule-based mechanisms. These mechanisms auto starts with auto scaling which scales up and scales down the VMs. These are simple but not conveying enough to address requirements of users. The approach has the computing elements as VM and their cost. The performance requirements are workflow and deadline. The aim of this approach is to complete the task with minimum cost.

The mechanism contains two plans:

The Scheduling plan is the decision which finds the instance type for each task at a particular time 't'. The Scaling plan decides the number of instances of each instance type at same point 't'. These scheduling and scaling plans are updated regularly since the VMs are billed on hourly basis.

1) Preprocessing

The auto scaling mechanism starts with preprocessing technique which consists of task bundling and deadline assignment.

a) Task Bundling

Task bundling assumes the tasks that prefer same type of instances as single task and allows running on same instance.

b) Deadline Assignment

The deadline assignment is based on fast execution time of task and low cost service for each task. The process also concentrates on the job makespan and the cost to complete the task. The rent has been determined based on the cost-efficiency of each machine.

$$Rank=(maskspan_{before} - maskspan_{after})/cost_{after}-cost_{before}$$

2) Dynamic-Scaling-Consolidation scheduling

a) Scaling

Load vectors (LV) is calculated for each task. The VM running time t_m for execution interval $[T_0, T_1]$ after deadline assignment.

$$LV_m=[t_m / [T_1-T_0]]$$

The loads vector for each task for each instance type are calculated and added to get 'm' load vectors. The number of instances should be greater equal to load vector at any time to complete within the execution interval.

b) Instance Consolidation

The tasks are executed on the same and most cost-efficient instance and they are fully utilized. Sometimes non cost efficient instances may be used to consolidate the partial instance hours. A smart decision based on scheduling is taken to consolidate the tasks at the same instances.

c) Dynamic Scheduling

After the determination of each instance type, the earliest deadline first algorithm is used to schedule the tasks. While scheduling, the deadline miss can be found in time. The instances are acquired by auto scaling method and complete the work within the deadline.

D. Dynamic Provisioning Dynamic Scheduling (DPDS)

DPDS is online algorithm[4] that provision resources and scheduler tasks at runtime. It has two procedures:

1) Provisioning Procedure

DPDS starts with fixed number of resources based on available time & budget.

$$N_{vm} = [b / (d * p)]$$

In this formula 'b' denotes the dollars, 'd' denotes the deadline, 'p' denotes the hourly price and 'N_{vm}' denotes the number of virtual machines. N_{vm} then periodically computes resources based on utilization of above or below given threshold. The VM complete their hour billing cycle is detected by provisioning interval & termination delay.

2) Scheduling Procedure

DPDS uses the dynamic priority-based scheduling procedure. Ready tasks enter the priority queue based on priority. When idle VMs are available then next task of priority queue is submitted. Low priority tasks are deferred when high -priority tasks are available.

3) Workflow-aware DPDS

Extends DPDS by admission procedure, when workflow enters this procedure estimates whether there is enough budget remaining to admit new workflow. If not, then workflow is rejected.

E. Transformation based optimization framework(ToF)

The cloud computing is the pay-as-you-go model which gives more important to metrics like cost and performance. This becomes the great challenge in the workflow and the performance system. Due to interconnected factors in the workflow, the performance and cost optimization became the important metrics. Requirements are differ based on the users which focus only on cost and compromise with performance. Some of them force on performance and compromise with budget. All these techniques lacks on optimization techniques on cost and performance. Then the transformation based optimization framework[5] has been processed to optimize the performance and cost. The directed acyclic graph(DAG) model has been used to denote the workflow.

1) Workflow

Workflow structures are generally represented as DAG is $G(V,E)$. V denotes the set of vertices is the tasks. E denotes the set of edges in the data dependencies between the tasks.

2) Initial Assignment

Initially a task assigned to the instance type per execution in each workflow. Various heuristics based methods are has been used to assign the instances to task which also forms as DAG called instance assignment graph.

3) Transformation operation

The transformation operations results in structural changes of the assignment of DAG. The transformation operations are classified as main schemes and auxiliary schemes. The main scheme aims to reduce the cost. The auxiliary schemes aim to change the form of workflow which is suitable for main scheme to reduce cost. The six basic workflow transformation operations are Merge, Demote, Split, Promote, Move and co-scheduling. The merge and demote operation comes under main scheme. The Split, Promote, Move and co-scheduling comes under the auxiliary scheme.

a) Merge operation

The merge operation performs when two vertices are assigned to the instances of same type. The vertices are assigned to one after another. The instance node of the instance DAG are combined to form the super node and maintain the hierarchical relationship and structural dependencies among the nodes in DAG.

b) Demote operation

The demote operation performs the execution of single vertex by assigning it to the cheaper instance which causes the longer execution time. The dependencies of the demote

vertex also delayed by the demoted vertex also delayed by the demoted vertex.

c) Move operation

The moving operation is used for moving one task after the end of another task to reduce the task. The dependencies of the moved vertex were also delayed by the moved vertex.

d) Split operation

The split operation is performed when more urgent task need to run on the same type instance by pausing the current task for a particular time. The suspended technique can be resumed by the checkpoint technique after the completion of the urgent task.

e) Promote operation

The promote operation is performed during the execution of the task to a better or costlier instance for decreasing the execution time. The promote operation are mainly performed to satisfy the deadlines. The promote operation continues with the merge operation to utilize the instances.

f) Co-scheduling operation

The co-scheduling operation is performed when multiple tasks running at the same time. The multiple tasks which have similar start time and end time with similar leftover time for deadline can be run at the same instance type.

4) Optimization Sequence

a) ToF Planner

This ToF planner helps to find the sequence of operation to be performed for cost and performance optimization. The planner has three designs. First the planner runs periodically. Second the main schemes and auxiliary schemes are performed alternatively and cost model avoids the unwanted transformation operations. Third the rule is defined to achieve the performance and cost optimization goal.

b) The Optimization process of ToF for workflows

All workflows are allowed in the form of queue. Each workflow is initially assigned with instance type. The selected main scheme and auxiliary scheme operations are performed for largest cost reduction without affecting the deadline of the workflow. If the time constraints are not violated then the cost has been estimated by cost model. If these transformation operation has no cost reduction then the initial assignment are executed to optimize the workflow.

F. Max-Min Task-Scheduling Algorithm

The main aim of task scheduling algorithm mainly in cloud computing is the allocation of task to the particular node. The allocation of task to the appropriate node is the challengeable task which can be improved by Max-Min Task Scheduling algorithm[6]. This scheduling algorithm maintains the task status table and virtual machine status table which help to identify the workload of virtual machines. This table also helps to find the task completion time.

When the tasks enter the scheduler, the scheduler finds the longest execution time tasks in the task status table then select the virtual machine in the virtual machine status table and finds the shortest completion time. This result in the allocation of tasks to the total number of tasks and their execution time in the virtual machine are updated in the virtual machine status table. This process is repeated until every task is allocated to the concerned virtual machine.

G. Cost with Finish Time-Based Algorithm(CwFT)

The cloud computing scheduling framework is mainly built on the basis of computing resources on cloud and computing components of our local systems. The Cost with Finish Time-based scheduling algorithm[7] focuses on increasing the performance of process and to reduce the monetary cost for resource provided by the cloud. Usually in distributed environments, the optimization techniques focuses on makespan, but the CwFT focuses on both makespan and monetary cost. The algorithm consists of two phase.

1) Task prioritizing phase

The priority level of all tasks is assigned in this phase. The priority of the task is estimated on the basis of length of critical path which includes the computation time. Finally, the sorted list of all tasks in descending order is determined. The goal of this list is to provide the topological order of the tasks. This list avoids the precedence constraints.

2) Node selection phase

In the phase the sorted list from the task prioritizing phase are scheduled to the appropriate nodes to optimize cost and performance. The appropriate nodes are selected on the basis of the Earliest Execution Starting Time (EST), Earliest Execution Finish Time (EFT) and Data Transfer Time (DTT). Based on the parameters, the best nodes are allocated to complete the tasks.

H. Multi Queue Job Scheduling (MQS)

The efficient job scheduling algorithm results in satisfaction of client and better resource utilization. The Multi-Queue Job Scheduling Algorithm[8] mainly focuses on reducing the cost and increasing the performance. The MQS algorithm classifies the jobs into three categories like small, medium and long based on the burst time of the job in cloud environment. The submitted jobs by the client are sorted in the ascending order based on the burst time. This categorization is done to increase the customer satisfaction and reduce the saturation by the dynamic allocation jobs to the appropriate nodes or systems. The best suitable jobs are allocated by this method without any decrease in the performance. The queue manager is responsible for resource utilization. The scheduling process and resource allocation are based on the three different queues named small, medium and long based on the time of jobs.

The first 40% of jobs are sorted in small queue then next 40% of jobs are sorted in medium queue and the remaining 20% of jobs are sorted in long queue. This algorithm gives importance to all jobs even though the client needs and expectations are different. The allocation of resource to jobs is dynamic which reduces time and space.

I. Genetic Algorithm with Particle Swarm Optimization (GAPSO)

The scheduling is done by both Genetic algorithm and particle swarm optimization. The Genetic algorithm helps to obtain the good quality solution. The particle swarm optimization is the approach used for solving the optimization problem. The GAPSO algorithm[9] is the hybrid technique which involves optimizing the cost on clouds. To increase the performance, the HEFT algorithm is also used in the initial population. The randomized solution of GA and PSO is obtained by initial population with cost, deadline and fitness of each solution. The selection and genetic operations are performed which results in new population. The new population, new velocity of each individual are obtained base on specified parameters. Fitness value has been evaluated for each solution and sorted based on fitness value. The worst solutions are placed by the best solution for each population. The process is repeated to get the optimal solution.

J. Optimized Resource Filling(ORF)

The scheduling algorithm mainly helps to reduce the idle time of the system and utilize the high resource usage. The optimized resource filling[10] utilize the unused space and attains the maximum resource usage with minimum starvation. The user submitted jobs are arranged in the ascending order. The small and medium jobs are grouped as the smadium in queue and remaining jobs in long queue. The job id is used to match the resources to the job. This induces the prioritization of the jobs by the scheduler. Now the resource manager updates the resources to the job. The resources are allocated on the basis of round robin method. During the arrival of new jobs the resources and queue jobs are dynamically changed by the queue manager. The jobs declare the resources required and their types during the arrival in queue. The proposed algorithm results in high throughput with minimum waiting time and low turnaround time.

K. Dynamically Optimized Cost-based Task Scheduling Algorithm

The main objective of this scheduling[11] approach is to combine the solution at user level and provider level to get the optimal solution. The cost based task scheduling aims to reduce the cost to the user. The dynamically optimized resource allocation strategy aims to benefit the service provider. The user tasks are grouped to utilize the available resources before resource allocation. In this algorithm the prioritization is based on the task profit. The execution done on descending order for the task profit. The high profit tasks are executed on the low cost machines to reduce the cost of the user. This algorithm benefits both the user and provider.

L. Adaptive workflow scheduling with iterative ordinal optimization.

This workflow scheduling model with iterative ordinal optimization helps to generate the optimal workflow schedules with the dynamic workload in virtual clusters. The aim of each iteration is to provide the bi-objective scheduling. The IOO method is mainly used for the fast dynamic

multitask workload scheduling[12]. The throughput has been decreased with reduced memory demand. The IOO method provides the efficient and effective workload scheduling based on iterative model compared to other OO method. The

IOO method reduces the execution time and manages the workload dynamically. The IOO method able to process even more successive iteration which is fast enough to adapt to dynamic workload variations.

III. SUMMARY

TABLE I. COMPARISON OF VARIOUS SCHEDULING ALGORITHM BASED ON COST AND PERFORMANCE OPTIMIZATION

S no	Author/year	Scheduling Algorithm	Parameters	Advantages	Disadvantages	Environment
1.	Y.C. Lee, R. Subrata, and A.Y. Zomaya (2009)	Adaptive Dual Objective Scheduling Algorithm(ADOS)	Execution Time, Makespan, Performance	1. Less completion time and more resource usage. 2. Manage continuous workload.	1. Complexities in rescheduling. 2. Under/over provisioning problems.	Grid
2.	E.K. Byun, Y.S. Kee, J.S. Kim, and S. Maeng (2011)	Partition Balanced Time Scheduling Algorithm(PBTS)	Gross Cost, Execution time, Makespan	1. Satisfies deadline. 2. Adapt to fluctuation.	1. Unnecessary repeated task execution	Cloud
3.	M. Mao and M. Humphrey(2010)	Auto Scaling Mechanism	Time, Cost, Performance, Makespan	1. Cost Efficient 2. Easy adaptable	1. Prefer same type of instance 2. Complex data dependencies	Cloud
4.	M. Malawski, G. Juve, E. Deelman, and J. Nabrzyski (2012)	Dynamic Provisioning Dynamic Scheduling (DPDS)	Time, Budget, price of VM	1. User prioritized workflow is maximized. 2. No provisioning delays.	1. Tasks may executed above deadline.	Cloud
5.	Amelie Chi Zhou and Bingsheng He (2014)	Transformation based Optimization Framework(ToF)	Monetary cost, Performance, Makespan	1. Reduce cost and increase performance	1. Planner Design is difficult	Cloud
6.	Xiaofang Li, Yingchi Mao, Yanbin Zhuang.(2014)	Max-Min Task-Scheduling Algorithm	Makespan, Performance, Execution time	1. Enhances the resources utilization 2. Better allocation of tasks	1. New task interrupt the running task 2. Efficiency of the system reduced.	Cloud
7.	Nguyen Doan Man, Eui-Nam Huh.(2013)	Cost with Finish Time-Based Algorithm(CwFT)	Monetary cost, Computation time, Makespan	1. Balance between performance and cost 2. Selection of best processing nodes	1. Increases complexity	Cloud
8.	AV.Karthick, Dr.E.Ramaraj, R.Ganapathy Subramanian.(2014)	Multi Queue Job Scheduling (MQS)	Burst time, cost, Performance.	1. Reduces starvation 2. Optimal resource usage	1. Unpredictable events may cause.	Cloud
9.	Tanyaporn Tirapat, Xiaorong Li, Tiranee Achalakul.(2013)	Genetic Algorithm with Particle Swarm Optimization(GAPSO)	Makespan, Deadline, Cost	1. Reduces cost 2. Optimal solution	1. Produce latency 2. Take more time	Cloud

S no	Author/year	Scheduling Algorithm	Parameters	Advantages	Disadvantages	Environment
10	AV.Karthick, E.Ramaraj, R.Kannan.(2013)	Optimized Resource Filling(ORF)	Waiting Time, Makespan,, Space.	1. Increase throughput. 2. Reduces starvation.	1. Large jobs may be delayed.	Cloud
11.	Yogita Chawla, Mansi Bhonsle. (2013)	Dynamically Optimized Cost based Task Scheduling Algorithm	Cost, Makespan, Overhead time.	1. Reduces cost. 2. Improves resources utilization	1. Grouping tasks are complex.	Cloud
12.	Fan Zhang, Junwei Cao, Kai Hwang, Keqin Li, Samee U. Khan. (2014)	Adaptive Workflow Scheduling with Iterative Ordinal Optimization	Memory, Throughput, Time.	1. High throughput. 2. Adapt to workload variation.	1. Not optimal	Cloud

IV. CONCLUSIONS

The cost and performance optimization become the most important issue during scheduling in cloud computing. In scheduling cost and performance optimization become the main issue. In this paper various workflow management techniques have been surveyed. The cost and performance optimization techniques have explained. Finally the algorithms are compared by their features. Various parameters needed for the optimization can be included in the future for better result on these techniques.

REFERENCES

- 1) Y.C. Lee, R. Subrata, and A.Y. Zomaya, "On the performance of a Dual Objective optimization Model for Workflow Application for Grid platforms", IEEE Trans. Parallel and Distributed Systems, vol. 20, no. 9, pp. 1273-1284, Sept. 2009.
- 2) E.-K. Byun, Y.-S. Kee, J.-S. Kim, and S. Maeng, "Cost Optimized Provisioning of Elastic Resources for Application Workflows", Future Generation Computer Systems, vol. 27, no. 8, pp. 1011-10261, 2011.
- 3) M. Mao and M. Humphrey, "Auto-Scaling to Minimize Cost and Meet Application Deadlines in Cloud Workflows", Proc. Int'l Conf. Grid Computing (GRID), pp. 41-48, 2010.
- 4) M. Malawski, G. Juve, E. Deelman, and J. Nabrzyski, "Cost and Deadline-Constrained Provisioning for Scientific Workflow Ensembles in IaaS Clouds," Proc. Int'l Conf. for High Performance Computing, Networking, Storage and Analysis (SC), pp. 22:1-22:11, 2012.
- 5) Amelie Chi Zhou and Bingsheng He, "Transformation-Based Monetary cost Optimization for Workflows in the cloud," IEEE transactions on cloud computing, vol. 2, no. 1, 2014.
- 6) Xiaofang Li, Yingchi Mao, Yanbin Zhuang, "An Improved Max-Min Task-Scheduling Algorithm for Elastic Cloud", International Symposium on Computer, Consumer and Control, 2014.
- 7) Nguyen Doan Man, Eui-Nam Huh, "Cost and Efficiency-based Scheduling on a General Framework Combining between Cloud Computing and Local Thick Clients", 2013.
- 8) AV.Karthick, Dr.E.Ramaraj, R.Ganapathy Subramanian, "An Efficient Multi Queue Job Scheduling for Cloud Computing", World Congress on Computing and Communication Technologies, 2014.
- 9) Tanyaporn Tirapat, Xiaorong Li, Tiranee Achalakul, "Cost Optimization for Scientific Workflow Execution on Cloud Computing", IEEE International Conference on Parallel and Distributed Systems, 2013.
- 10) AV.Karthick,E.Ramaraj,R.Kannan,"Optimized Resource Filling Technique for Job Scheduling in Cloud Environment", International Conference on Computing and information Technology, 2013.
- 11) Yogita Chawla, Mansi Bhonsle, "Dynamically optimized cost based task scheduling in Cloud Computing", International journal on emerging trends and technology, Volume 2, Issue 3, 2013.
- 12) Fan Zhang, Junwei Cao, Kai Hwang, Keqin Li, Samee U. Khan, "Adaptive Workflow Scheduling on Cloud Computing Platforms with Iterative Ordinal Optimization", 2014.