ISSN (e): 2250 – 3005 || *Volume, 05* || *Issue, 03* || *March – 2015* ||
International Journal of Computational Engineering Research (IJCER)

Cost Optimization in Multi Cloud Platforms using Priority Assignment

Anoop Abraham Eapen[1], Vineetha V Nair[2]

1 Student, Department Of Computer Science & Engineering, Mangalam College of Engineering, Ettumanoor, Kottayam , India

2 Assistant Professor, Department Of Computer Science & Engineering, Mangalam College of Engineering, Ettumanoor, Kottayam , India

ABSTRACT

Cloud computing is one of the emerging technologies today. The cost and pricing schemes that are often associated with the cloud services offered can be misleading to most users. Cost optimization is very important as far as cloud services are concerned. In this paper, a priority assignment function is introduced, which assigns priority to the data, based on which the data storage and encryption is performed, depending on the priority assignment which in turn reduces the unnecessary cost and time complexity that is associated with data storage.

KEYWORDS: *Virtual Machine, Virtual Machine Monitor, Directed Acyclic Graphs, Transformation Oriented Framework etc…*

I. INTRODUCTION

Cloud computing is one of the emerging technologies today. Different categories of people use the services that are offered by the cloud today ranging from individuals to large organizations. Costs that are often associated with the cloud services can be very high. Hence it is very important that the operations that are associated with the cloud needs to be optimized. One of the performance challenges facing traditional cloud computing environments is rooted in their architecture. First-generation cloud platforms were housed in a relatively small number of relatively large, high-capacity data centers. Even when these large data centers are housed in the world's largest networks, the average distance between the data center and the end user can be more than 1500 miles. Optimizing the cloud for maximum performance involves a more distributed cloud, with capacity located in many locations, closer to end users. This reduces the number of network 'hops' that every request, every piece of data, every bit of content must make to reach the end user. The shorter the distance, the fewer network hops, the greater the speed, and the better the user experience. The cloud services offered often removes the problems associated with actual physical hardware manipulation. The cloud also provides task execution environments and facilities by allocating virtual machines to the tasks. The virtual machines are designed so as to support different execution needs. They are generally allocated depending upon the type of tasks that are to be executed by the user who is requesting to perform the tasks. The cloud service providers make use of the pay as you go pricing scheme, where the user will be billed on the basis of a particular time slot that is allotted to him. This scheme is often advantageous to many users as the get to access resources of much larger capacity, for each time interval. The data storage part is also important, as the data that is to be stored into the cloud should be stored in such a way that the overall cost that is associated with the data storage is also reduced.

II. RELATED WORK

There had been many approaches towards cost optimization in cloud systems. There are different operations that were performed on tasks, as well as on files that are stored on to the cloud so as to improve the cost factor. Earlier papers discuss operations such as merge, promote, demote, split, move and co scheduling operations on tasks so as to improve the VM utilization as well as to reduce the overall costs. Different tools that are associated with workflow management such as Pegasus[12], [1]workflow execution environments such as DagMan and Condor are discussed in detail in previous studies. Some studies discuss cost optimization in cloud systems by optimally under clocking the VM’s so as to decrease the overall power consumption by using mechanisms such as DVFS[8]. Grid workflow execution strategies[5] based on algorithms such as the ADOS algorithm have also been discussed in previous studies.

III. EXISTING SYSTEM

Before specifying our existing system, we need to have an understanding regarding the following concepts:

3.1 VM:

The virtual machine instance is the actual platform that is designed to carry out the workflow execution. It may be different for different tasks depending upon the time and resource constraints that are imposed by the particular task.

3.2 VMM:

There are different virtual machines that are used for the execution of the different tasks. They have to be maintained and usually are present as a pool of VM's. The Virtual Machine Monitor is responsible for the management of the virtual machines[1]. The Virtual Machine Monitor keeps track of the various virtual machines, to which task they are allocated, when they will be freed etc.

3.3 DAG:

Directed Acyclic Graph is generated by breaking down the jobs[1]. They represent how the workflow proceeds. The various operations that are specified are also performed on workflows. Our existing system performs the specified workflow optimization operations which is performed on workflows for optimizing them. Each task or job is broken down into corresponding DAG's. The DAG's specify the operations that are performed in a job.

The existing system introduces ToF, in which 2 schemes based on which workflow optimization takes place. It also introduces a planner, which governs how the following operations are to be performed so as to obtain the most optimal result. The existing system, when compared to other systems, reduces the overall cost of workflow execution. However, there is no hint of how the data post processing, will be stored in the cloud databases. There will be different types of information that are present in the cloud. So, providing the same level of security to all the data is not a good or as optimal method of storage in the cloud. As shown in the figure, the workflows are put in a FIFO queue and on its basis the optimization is carried out. The optimizer repeatedly checks, whether any possible optimizations can be employed on the DAG's. Two operations - Merge and Demote are the main schemes and Move, Promote, Co-Scheduling, Split, etc. are the auxiliary schemes. The transformation model is responsible for performing the various transformations, while the time of action is specified by the planner. The auxiliary schemes support the execution of the main schemes. The transformation model is responsible for performing the various transformations, while the time of action is specified by the planner.

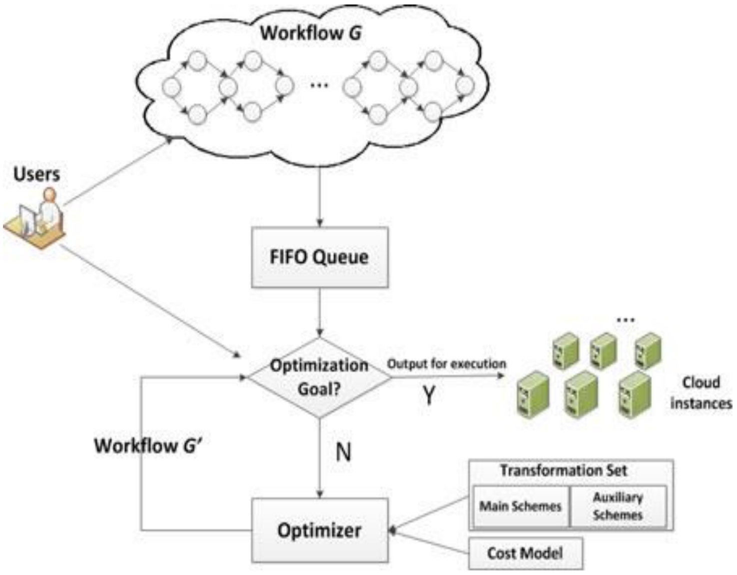


Figure 3.1 ToF Overview

3.1 ToF Optimization Algorithm

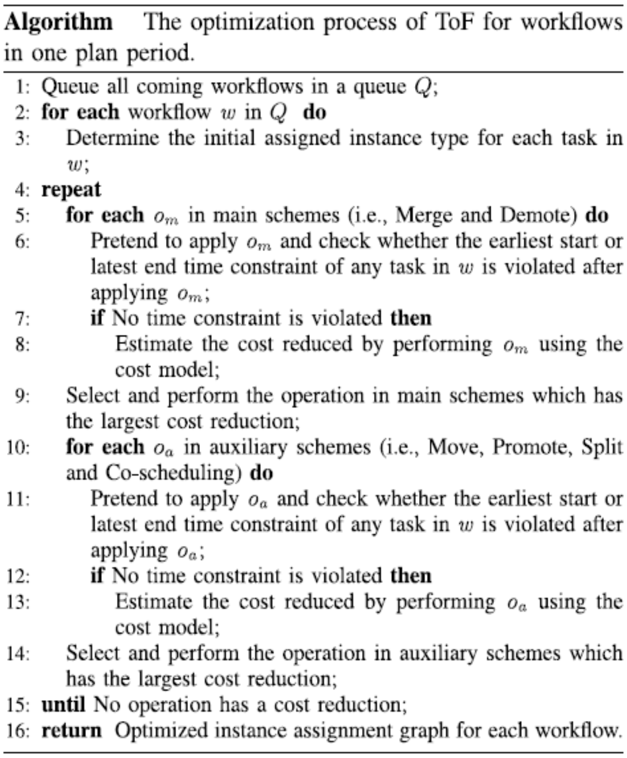
ToF optimization algorithm is used in the existing system for performing the various operations on the tasks. The algorithm operates on the tasks, and initially it pretends to apply the Merge and Demote[1] operations on the task. It checks whether by the application of any of these operations, the time assignment of any task will skip the deadline. If so, then the operations cannot be performed. Else, if it is possible to apply these operations, then the operation that will bring about the most optimization will be applied.

The main objective of the application of the ToF optimization framework is to maximize the optimization potential, i,e, by reducing the overall cost and time needed for the execution of that particular task. The other operations, i.e, auxiliary operations are also applied, if the main operations can be performed; i.e, the time constraints are not violated. The operations are done until no Figure 3.2 illustrates the ToF Algorithm.

3.2 Planner

Planner is the next important component that is associated with the optimization framework. Although the operations are needed for performing the cost and time reduction, i.e, optimization, the operations have to be applied systematically to the various task that arrive to be executed. The planner has the following properties:

1. The evaluation of the searching space that is associated with the conduct of the operation is a tedious task. Besides using the main as well as the auxiliary schemes alternatively, so as to reduce the overhead that is associated with searching unnecessary tasks. The planner also makes use of the cost model, so as to prune off the unnecessary operations, that does not yield any favourable results.
2. The planner that is used within this scheme is rule based. i.e, it works on the basis of a set of rules, consisting of certain conditions and actions. The conditions denote the situations to be met for the actions to be performed.
3. The planner is ran periodically, so as to make the system dynamic and work in real time[1]. i.e, the tasks should be allocated periodically to new VM instances. The pay as you go pricing scheme of the cloud structure means that the service that is offered can be used for a particular time period only.



Algorithm The optimization process of ToF for workflows in one plan period.

```
Queue all coming workflows in a queue $Q$;
for each workflow $w$ in $Q$ do
    Determine the initial assigned instance type for each task in $w$;
repeat
    for each $o_m$ in main schemes (i.e., Merge and Demote) do
        Pretend to apply $o_m$ and check whether the earliest start or latest end time constraint of any task in $w$ is violated after applying $o_m$;
        if No time constraint is violated then
            Estimate the cost reduced by performing $o_m$ using the cost model;
    Select and perform the operation in main schemes which has the largest cost reduction;
    for each $o_a$ in auxiliary schemes (i.e., Move, Promote, Split and Co-scheduling) do
        Pretend to apply $o_a$ and check whether the earliest start or latest end time constraint of any task in $w$ is violated after applying $o_a$;
        if No time constraint is violated then
            Estimate the cost reduced by performing $o_a$ using the cost model;
    Select and perform the operation in auxiliary schemes which has the largest cost reduction;
until No operation has a cost reduction;
return Optimized instance assignment graph for each workflow.
```

Figure 3.2 ToF Algorithm

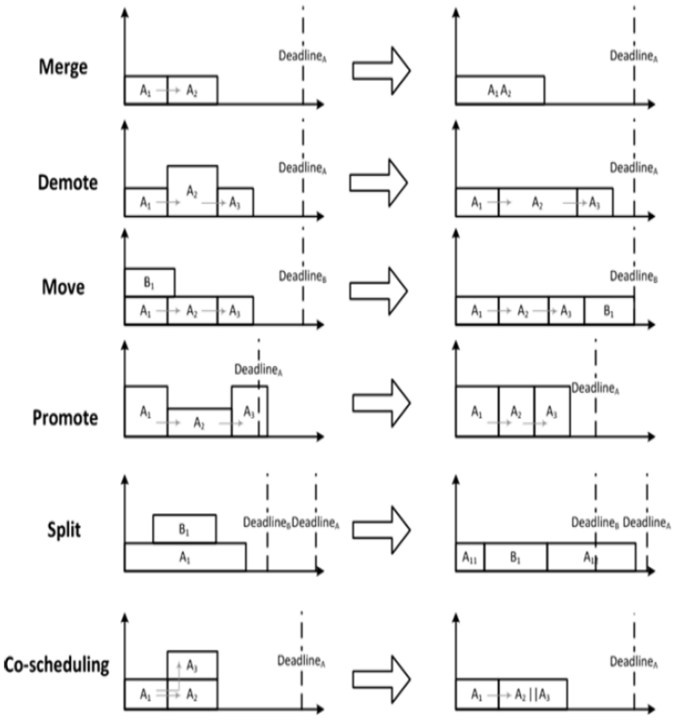
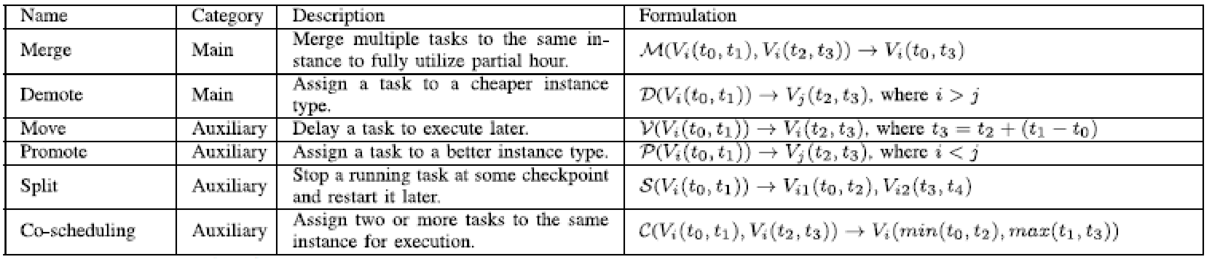


Figure 3.3 Instance time chart based on transformation operations

Figure 3.3: ToF operations



Name	Category	Description	Formulation
Merge	Main	Merge multiple tasks to the same instance to fully utilize partial hour.	$\mathcal{M}(V_i(t_0,t_1),V_i(t_2,t_3)) \rightarrow V_i(t_0,t_3)$
Demote	Main	Assign a task to a cheaper instance type.	$\mathcal{D}(V_i(t_0,t_1)) \rightarrow V_j(t_2,t_3)$, where $i > j$
Move	Auxiliary	Delay a task to execute later.	$\mathcal{V}(V_i(t_0,t_1)) \rightarrow V_i(t_2,t_3)$, where $t_3 = t_2 + (t_1 - t_0)$
Promote	Auxiliary	Assign a task to a better instance type.	$\mathcal{P}(V_i(t_0,t_1)) \rightarrow V_j(t_2,t_3)$, where $i < j$
Split	Auxiliary	Stop a running task at some checkpoint and restart it later.	$\mathcal{S}(V_i(t_0,t_1)) \rightarrow V_{i1}(t_0,t_2),V_{i2}(t_3,t_4)$
Co-scheduling	Auxiliary	Assign two or more tasks to the same instance for execution.	$\mathcal{C}(V_i(t_0,t_1),V_i(t_2,t_3)) \rightarrow V_i(min(t_0,t_2),max(t_1,t_3))$

IV . PROPOSED SYSTEM

In the proposed system, a mechanism is employed that ensures only the relevant data that is stored within the cloud databases are secured with relevant algorithms. A priority assignment feature is also proposed, which helps to assign priority to the information that has to be stored in the cloud. The system is explained with the concept of a multi cloud implementation, where the data is stored in different cloud databases. The encryption is done on the basis of the priority that is assigned to each data, and those data which are not assigned any priority will be considered as general data, and will be stored in a separate database. This removes a portion of the burden that is associated with encryption and decryption and helps to reduce the cost further, as far as any cloud service provider is concerned. It also helps to reduce unnecessary precautions that are taken to protect the data that is stored within the cloud. There will be significant cost reduction in large data storage structures, or in places where huge amounts of data are produced from the execution of tasks, that need to be stored within the cloud.

4.1 Priority Assignment

The priority assignment can be made to tasks by the the user. The execution as well as the storage of the result will depend upon the priority that is assigned. Based on the priority, it is decided, whether to provide encryption before the data is stored within the secured cloud storage, or to store it without encryption separately. Hence the overhead of encryption and decryption for unnecessary data as well as the cost associated with it is reduced. This improves the overall performance of large task execution clusters, where a lot of output data may be generated as the result. The priority assignment truly becomes a boon in such scenarios, where the bulk of information can be chosen to store on priority basis.

V. EXPERIMENTATION & RESULTS

The simulation is carried out using two systems, on which one of them runs the VM application and the other runs the client side. Both systems are powered by intel dual core processors with 1GB of RAM. The systems are interconnected via LAN. The admin system illustrates the admin, which is responsible for performing the various operations on the tasks that are provided by the user. The VM is hosted on the system, which is responsible for performing the various operations on the tasks that are provided by the user. The system is simulated with the help of one user as well as one arithmetic operation that is given to the virtual machine. The cost is evaluated by setting predefined cost parameters that is set based on the evaluation criteria of similar other systems, such as the ToF. The overall cost i.e, the cost of the operations as well as the data storage combined, is slightly lower for this scheme as compared to the earlier scheme, where additional modules are used for storing the computation results. Additionally, our scheme also deals with the data storage service, post processing. The comparative study is done, by considering the ToF.

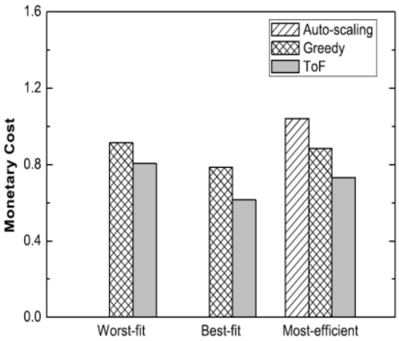


Figure 5.1 Comparison of ToF with other schemes

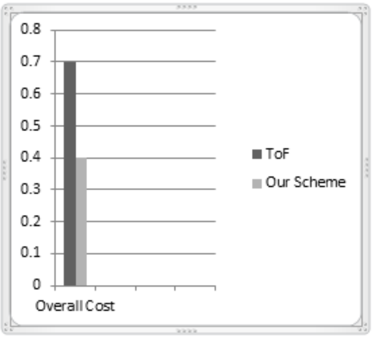


Figure 5.2 Comparison of ToF with our scheme

VI. CONCLUSION

The cost optimization for workflows is one of the most important activities, that is being carried out by different cloud vendors to reduce their overall execution time as well as the total monetary cost. Our system includes the advantages that are proposed by the ToF scheme, as well as the reduction in unnecessary cost incurred due to the provision of security to all the data, that is provided. The storage as well as the retrieval part is also simplified due to the removal of encryption to non important data. The system focuses primarily on cost reduction. Less focus has been given on reducing the computation time as well the batch processing of similar tasks, which can be enhanced, to further improve this scheme.

REFERENCES

[1] Amelie Chi Zhou and Bingsheng He, Transformation based monetary cost optimizations for workflows in the cloud, IEEE Transactions on cloud computing, March 2014

[2] M. Mao and M. Humphrey, "Auto-Scaling to Minimize Cost and Meet Application Deadlines in Cloud Workflows," Proc. Int'l Conf. High Performance Computing, Networking, Storage and Analysis, 2011.

[3] M. Mao, J. Li, and M. Humphrey, "Cloud Auto-Scaling with Deadline and Budget Constraints," IEEE/ACM 11th Int'l Conf. Grid Computing (GRID),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 ,2012.

[5] Y. Gong, B. He, and J. Zhong, "Network Performance Aware MPI Collective Communication Operations in the Cloud," IEEE Trans. Parallel and Distributed Syst., vol. 99, 2013, doi: 10.1109/TPDS. 2013.96.

[6] M. Mao and M. Humphrey, "Scaling and Scheduling to Maximize Application Performance within Budget Constraints in Cloud Workflows," Proc. IEEE 27th Int'l Symp. Parallel and Distributed Processing (IPDPS), 2013.

[7] H.M. Fard, R. Prodan, and T. Fahringer, "A Truthful Dynamic Workflow Scheduling Mechanism for Commercial Multicloud Environments," IEEE Trans. Parallel and Distributed Systems, vol. 24, no. 6, pp. 1203-1212, June 2013.

[8] Q. Zhu, J. Zhu, and G. Agrawal, "Power-Aware Consolidation of Scientific Workflows in Virtualized Environments," Proc. Int'l Conf. for High Performance Computing, Networking, Storage and Analysis (SC), pp. 1-12, 2010.

[9] H. Wang, Q. Jing, R. Chen, B. He, Z. Qian, and L. Zhou, "Distributed Systems Meet Economics: Pricing in the Cloud," Proc. Second USENIX Conf. Hot Topics in Cloud Computing (HotCloud), pp. 6-6, 2010.

[10] J.-S. V€ockler, G. Juve, E. Deelman, M. Rynge, and B. Berriman, "Experiences Using Cloud Computing for a Scientific Workflow Application," Proc. Second Int'l Workshop Scientific Cloud Computing (ScienceCloud, pp. 15-24, 2011

[11] H. Kllapi, E. Sitaridi, M.M. Tsangaris, and Y. Ioannidis, "Schedule Optimization for Data Processing Flows on the Cloud," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2011.

[12] E. Deelman, G. Singh, M.-H. Su, J. Blythe, Y. Gil, C. Kesselman, G. Mehta, K. Vahi, G.B. Berriman, J. Good, A. Laity, J.C. Jacob, and D.S. Katz, "Pegasus: A Framework for Mapping Complex Scientific Workflows onto Distributed Systems," Scientific Programming, vol. 13, no. 3, pp. 219-237, 2005.