Indian Journal of Science and Technology, Vol 9(45), DOI: 10.17485/ijst/2016/v9i45/106491, December 2016
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

A Security Aware Resource Allocation Model for Cloud Based Healthcare Workflows

P. Blessed Prince*, D. Agnes Ruphavathani and S. P. Jeno Lovesum

School of Computer Science and Technology, Karunya University, Coimbatore - 641114, Tamil Nadu, India;
blessedprince@gmail.com, agnesruphavathani@gmail.com, jenolovesum@gmail.com

Abstract

Objective: Cloud Computing is the new model in distributed computing that provides Software as a service (SaaS), Infrastructure as a service (IaaS), and Platform as a service (PaaS). Researchers utilize the services of cloud for running large scale data and other computation intensive applications like healthcare workflows. The major issue arises while allocating the cloud resources to the task in the workflows, in this paper an efficient scheduling method for work flow has been proposed. **Analysis:** The applications are modeled as workflows which include dependent tasks that are represented as a Directed Acyclic Graph (DAG) considering every node as a job. The known algorithms cannot solve the problem of resource allocation, and are categorized as NP hard problem. **Findings:** The proposed hybrid workflow scheduling algorithm which is the combination of Heterogeneous Earliest Finish Time (HEFT) and Security Aware and Budget Aware (SABA) algorithm helps to schedule the workflow efficiently with reduced makespan and required security services.

Keywords: Allocation, Cloud, Healthcare, NPhard, Scheduling, Security

1. Introduction

Cloud Computing is the emerging trend in distributed computing that provides Software as a service (SaaS), Infrastructure as a service (IaaS), and Platform as a service (PaaS).[7]Using cloud computing, we overcome the disadvantages of storage constraint, losing the data, hardware malfunction, data availability, etc. Scientist and researchers utilize the services of cloud for running large scale data and other computation intensive applications like healthcare workflows. [8]Workflow is defined as the set of tasks that is performed in series or in parallel. These workflow applications can be implemented in commercial public cloud like Amazon (Highly cost efficient) which denotes that it is Infrastructure as a Service (IaaS). Even though many papers discuss the scheduling of tasks problem, scheduling workflows efficiently with the security requirements is an area that needs to be explored further.

[9]Scheduling is how the tasks are assigned to run on available resources. The cloud user submits their request and the user request are formed as tasks. The tasks will be scheduled to the corresponding services and these tasks may require different security services such as confidentiality, integrity and authentication with various security levels denoted by the user. Scheduling these types of workflows with success rate is a challenging task.

[10]Resource allocation mechanism is done to identify and locate which VM will be best suitable for the cloudlets i.e., resource type may be small, medium, large, etc., which vary by cost and other. QoS (Quality of Service) parameters such as execution time, make span, computing time, storage, etc. are considered in to measure the success rate of the workflow. [11]In general, efficient workflow scheduling will include the best resource allocation that is done considering the deadline constraint and the security factors without compromising the QOS requirements such as makespan, cost, etc.,

This paper is organized as follows Section II describes the related work, Section III explains the proposed algorithm followed by the results and discussions in Section IV and Section V concludes the work and highlights the future work.

*Author for correspondence

2. Related Works

The workflow scheduling problems are categorized as NP hard problem since finding of the optimal solution increases with the size of the problem. In such cases HEFT is the heuristic algorithm that is widely used for scheduling the workflows[12]. Billing model Aware Task Clustering algorithm is proposed which reduces the makespan of the workflow and provides efficient scheduling[3]. Further in order to execute the workflows within the given deadline constraint[1] proposed a partial critical path algorithm that minimizes the cost of the execution. When considering the workflows with security services[2] discusses the properties that are required for proper resource allocation with security parameters. Security Aware and Budget Aware algorithm proposed by[4] considers both the security parameters as well as the cost factors. Job scheduling algorithm based on Berger model offered by[6] classifies the task according to its QOS requirements and performs the resource allocation[16] proposed a deadline based resource provisioning and scheduling algorithm based on Meta heuristic optimization technique and particle swarm optimization which aims to minimize the overall workflow execution cost while meeting the deadline constraints.

[12]Existing research gives solutions to minimize the makespan, cost and so on but does not focuses more on the resource allocation with the required security services. The security requirements certainly has the impact on the resources such as memory, cost, etc., The proposed algorithm highlights those issue and provides solution for the resource allocation problem while considering the security factors as shown in fig.3.2.

3. Proposed Algorithm

3.1 Workflow Modeling

[13]Resource allocation algorithm for the workflow is proposed by combining the task clustering mechanism, Heterogeneous Earliest Finish Time (HEFT) algorithm with SABA algorithm. The workflow is usually set of inter-dependent tasks that are represented as Directed Acyclic Graph (DAG). [14]The DAG workflow will be then considered as the XML files which are processed by workflow management tool such as Pegasus to generate a DAX file.

[17]The workflow is represented as a Directed Acyclic Graph (DAG). A DAG, G = (V; E), where V is the set of v nodes and each node V represents computational task. E is the set of e communication edges between tasks; each $e_{(i,j)}$ represents the task dependency constraint such that task n_i should complete its execution before task n_j can be started. [18]Example of the workflow in DAG representation is given in Fig 3.1 where S, Q are start and end nodes. Here, the nodes A,B,C and D,E,F executes in parallel as shown in fig 3.1

3.2 Cloud Resource Allocation

The proposed framework of the workflow scheduling comprises of the following modules.

[15]Workflow planner creates a series of task and submits to the clustering engine which is responsible for the merging of jobs into tasks according to the randomized task clustering mechanism. The workflow engine does the

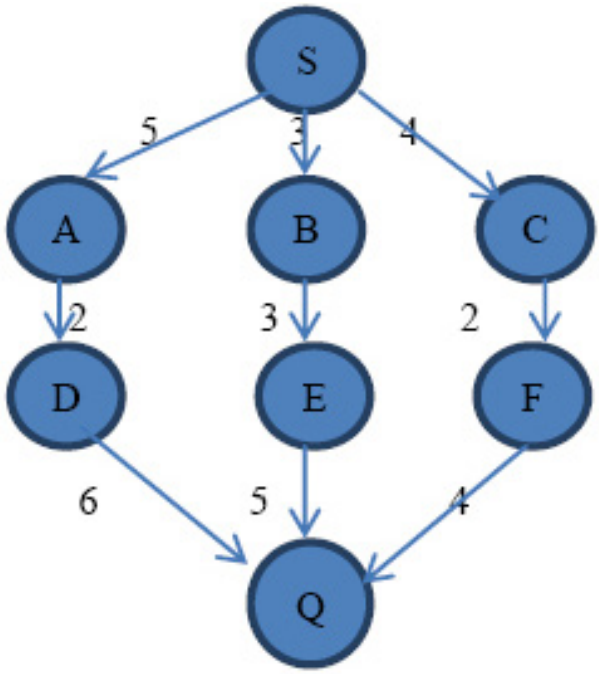

Figure 3.1. Example Workflow- DAG Representation.

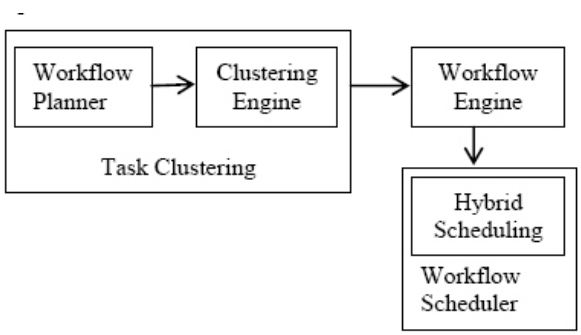

Figure 3.2. Framework of Security Aware Cloud Resource Scheduling.

work of releasing of jobs according to its dependencies and workflow scheduler does the VM assignment which is based on the proposed algorithm.

Thus the algorithm which schedule the jobs according to the user request to the servers in the cloud environment and tries to reduce the idle time in making the optimal utilization of the server. Hence better optimal solutions can be obtained with the specified requirements.

The proposed hybrid workflow scheduling algorithm is as follows

Input: Let T be the set of all tasks in the workflow G.
Let R be the set of all the available resource types.
Deadline D, Budget B and Security level S.

Output: Obtain Schedule S such that it meets Deadline D and Budget B when deployed on cloud platform.

Step1: Distribute the Given D,B and S across each level.

```
For( i=0;i<=Depth(G);i++)
Deadline_level[i] = distribute_deadline (G,I,D);
Budget_level[i] = distribute_budget (G,I,B);
Security_level[i] = distribute_security (G, I, S);
End Loop
Integer I, makespan=0, cost=0, Level=0;
```

Step 2: Identify the order of execution of dependent tasks.

```
Prioritization (G)
```

Step 3: Resource Allocation

```
For i=0 to i<=levels of the DAG G do
Resource_selection (G(T[i],R)
End procedure
```

Prioritization Algorithm of step 2 is as follows

1. Assign a weight to each node as the average computation cost across all machines Assign a weight across each edge as the average computation cost across combination of all machines.
2. Use priority ranking to compute a rank value for each node.
3. Sort nodes in descending order of their rank values.
4. Scan nodes in descending order of their rank values

```
if current node has a dependence with a node in G
then
add node to G
keep scanning until there are no more nodes
```

Let us consider that the task in the workflow requires various security services. The priority ranking can be computed by adding the computation cost of that node, I/O cost and the security cost by traversing the graph upward from the exit node as given in equation (1).

$$PR = Comp\_Cost + I/O\_Cost + Sec\_Cost \quad (1)$$

The cost of the task t_i that executes on VM V_j with price p can be computed using the equation (2)

$$Cost\ (t_i,v_j) = (exe\_time_{i,j} + I/O\ time_{i,j} + sec\_cost_{i,j}) * p \quad (2)$$

The makespan of the workflow G can be defined as the total completion time of the tasks and can be computed using the equation (3)

$$Makespan\ (G) = max\ (\ eft\ \{Q\}\ ) \quad (3)$$

Here Q is the exit node of the workflow. In every computation of the task the security cost factor is also included so that the impact of the security services can be achieved.

4. Results and Discussions

Simulation is considered as the method of evaluation of the workflows and the implementation is done using Workflow Sim 1.0 which has the functionalities to simulate the workflows in cloud environment. The input that is given to conduct the experiment is the biological/healthcare workflows, Epigenomics which consists of 24, 48, 100 nodes DAG. The results that are projected is the 24 nodes workflow parameters. The general structure of Epigenomics workflow is given in fig 4.1.

The performance metrics that are considered for the resource allocation algorithm is execution time, make span and deadline miss rate. The implemented hybrid workflow scheduling algorithm are compared with the HEFT and Data aware scheduling algorithm and found that the proposed algorithm yields the better results.

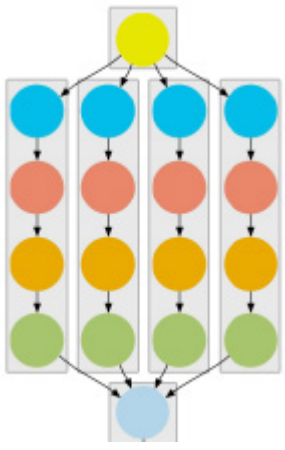

Figure 4.1. Epigenomics- healthcare/biological workflow structure.

4.1 Execution Time

Execution time is the time taken for executing the jobs in that workflow. Figure 4.2 shows the comparison of the execution time of the proposed Hybrid, HEFT and Data Aware Scheduling algorithms for jobs of the workflow which consists of 24 nodes.

From the graphs it is identified that hybrid scheduling results in minimized execution time and is improvement of 7% is made from that of the traditional algorithms.

Table 4.1.Comparison of Execution Time

HEFT Scheduling	Hybrid Scheduling	Data Aware Scheduling
33917	15242	76813
13037	18310	45069
16941	12430	11260
10875	28943	11761
18673	19512	21826
24574	14814	46458
15320	25914	65391
20484	33334	62680
19636	7763	34188
11365	23593	74258
12207	14917	19390
27849	14600	71218
29339	16097	52655
11786	11929	33220
22207	19900	22957
17932	17665	25895
33286	6388	65845
25924	27535	96895

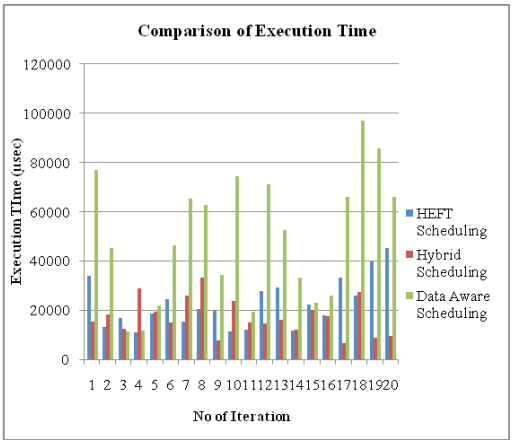

Figure 4.2. Comparison of Makespan.

4.2 Makespan

Makespan is the total completion time that is the time required to execute the entire workflow with its delay included. Figure 4.3 shows the comparison of the proposed algorithm with the HEFT and the Data Aware Scheduling.

4.3 Deadline

The deadline of the jobs is compared with the actual execution time to determine whether the jobs have executed within the given deadline. Figure 4.3 is the

Table 4.2

HEFT Scheduling	Hybrid Scheduling	Data Aware Scheduling
60474	39922	118697
23150	31215	73516
24336	33792	134914
28373	57632	45548
30734	43009	75937
35112	36468	90616
32477	55042	137520
56461	47155	58436
39006	23647	198905
49937	31369	259912
41015	21365	65903
50342	40536	91040
29513	35328	65178
30651	25046	161119
25498	21056	51664
24577	35159	67854
53719	20742	56895
33151	30154	54789

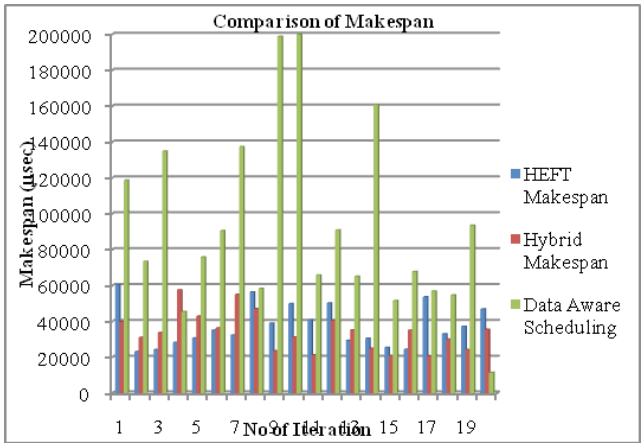

Figure 4.3. Comparison of Makespan.

Table 4.3. Comparison of Deadline

Job Id	Deadline	Actual Execution time - Hybrid Scheduling	Actual Execution time – Data Aware Scheduling
1	7.89	7.01	8.15
2	0.05	0.05	0.07
3	8.23	7.89	8.01
4	11.89	13	14.7
5	5.91	5.27	5.85
6	31.4	29.92	29.98
7	5.02	2.43	5.56
8	3.54	3.58	4.12
9	17.3	18.226	17.56
10	8.88	5.16	8.01

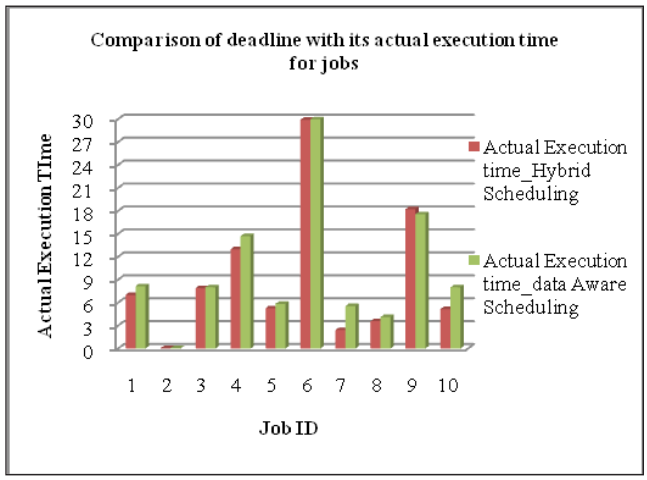

Figure 4.4. Comparison of deadline with its actual execution time for jobs.

comparison of deadline with its actual execution for jobs in the workflow.

From the graph below it is identified that the hybrid scheduling reduces makespan of the workflows which results in better performance of about 6%.

From the graph it is confirmed that the proposed hybrid scheduling algorithm executes the jobs within the given deadline

5. Conclusion

Scientist and researchers utilize the services of cloud for running large scale data and other computation intensive applications like healthcare workflows. These applications can be implemented in commercial public cloud like Amazon (Highly cost efficient).Scheduling of workflow is how the tasks are assigned to run on available resources. The workflows are represented as a Directed Acyclic Graph (DAG) considering every node as a job. The known algorithms cannot solve the problem of resource allocation, and are categorized as NP hard problem. The proposed hybrid workflow scheduling algorithm which is the combination of the task clustering mechanism and Heterogeneous Earliest Finish Time.

(HEFT) algorithm helps to schedule the workflow efficiently by reducing the execution time, makespan and also within the deadline constraint and yields the better results and improvement of in execution time and makespan respectively.

6. References

1. Samanthula BK, Elmehdwi Y, Howser G, Madria S. A secure data sharing and query processing framework via federation of cloud computing. Information Systems. 2013.
2. Fabian B, Ermakova T, Junghanns P. Collaborative and secure sharing of healthcare data in multi-clouds. Information Systems. 2014.
3. Meli CL, Khalil I, Tari Z. Load-sensitive dynamic workflow re-orchestration and optimisation for faster patient healthcare. Computer Methods and Programs in Biomedicine. 2014; II3(I-14):1–14.
4. Tsai C-W, Huang W-C, Chiang M-H, Chiang M-C, Yang C-S. A Hyper-Heuristic Scheduling Algorithm for Cloud. IEEE Transactions on Cloud Computing. 2013.
5. Thilakanathan D, Chen S, Nepal S, Calvo R, Alem L. A platform for secure monitoring and sharing of generic health data in the Cloud. Future Generation Computer Systems. 2013; 35:102–13.
6. Abdelkader DM, Omara F. Dynamic task scheduling algorithm with load balancing for heterogeneous computing system. Egyptian Informatics Journal. 2012.
7. Hsieh F-S, Lin J-B. Development of context-aware workflow systems based on Petri Net Markup Language. Computer Standards and Interfaces. 2014; 36:672–85.
8. Russello G, Dong C, Dulay N. Personalising Situated Workflow Systems for Pervasive Healthcare Applications. 2008.
9. Gogulan R, Kavitha A, Karthick Kumar A. A Multiple Pheromone Algorithm for Cloud Scheduling with Various QOS Requirements. IJCSI. 2012.
10. Al-Souly HA. Lightweight Symmetric Encryption Algorithm for Secure Database. International Journal of Advanced Computer Science and Applications. 2013; 53–62.
11. Jayadivya SK, Bhanu MS. QoS Based Scheduling of Workflows in Cloud Computing. IJCSEE. 2012.

12. Wan J, Zou C, Ullah S, Lai C-F, Cheng NC, Zhou M, Huazhong, Wang X. Cloud – Enabled Wireless Body Area Networks for Pervasive Healthcare. IEEE Networks. 2012; 1–7.
13. Yanga J-J, Li J-Q, Niu Y. A hybrid solution for privacy preserving medical data sharing in the Cloud environment. Future Generation Computer Systems. 2014.
14. He L, Huang C, Duan K, Li K, Chen H, Sun J, Jarvis SA. Modeling and analyzing the impact of authorization on workflow executions. Future Generation Computer Systems. 2012; 28:1177–93.
15. Rodriguez MA, Buyya R. Deadline based Resource Provisioning and Scheduling Algorithm for Scientific Workflows on Clouds. IEEE Transactions on Cloud Computing. 2013.
16. PaulRani A, GomathyNayagam M. Multiobjective Qos optimazation based on multiple workflow scheduling in cloud environment. IJIRCCE. 2013.
17. Senkul P, Toroslu IH. Architecture for workflow scheduling under resource allocation constraints. Information Systems. 2004; 30:399–4