

THE POSSIBILITY OF RESOURCE SCHEDULING FOR A COLLABORATIVE CLOUD COMPUTING USING FAILURE WORKFLOW CONTROL IN IAAS CLOUD

KAVITHA GANESAN¹ AND ANGELIN PEACE PREETHI²

¹Department of Electronics and Communication Engineering
Government College of Technology, Salem, India
Email : kavitha@gcesalem.edu.in

²Karpagam College of Engineering, Anna University, Chennai
Email : 3jsngl@gmail.com

ABSTRACT. Cloud computing is a pledge that the implementation of low-cost workflow processes has become a prototype computing infrastructure that you supply resources. Despite the need for scientific workflows and access to community-level resources, they usually require a collaborative cloud environment that creates multiple data centers. Combining Artificial Bee Colony algorithm (ABC), and Hybrid Genetic Algorithm (HGA) geographically dispersed data center inputs and intermediate data sets can cause unbearable waiting periods that interrupt the execution of the addresses. Large-scale data-intensive scientific workflows and cloud workflow execution cost. In the proposed system, Failure-Aware Workflow Control Resource Scheduling (FAWCRS) is understanding the Importance and Simulation of FAWCRS Content Natures to Monitor the Importance of Diversity Analyze test results on the impact of tracking diversity. Failure traces and defect detection and reduction of average completion time to accurately achieve predictable accuracy, lost time, and an incorrect number of rearrangements. It effectively improves the availability, reliability, and quality of intermediate data in the Scientific Workflow. Solve this system planning problem on infrastructure on a Service (IaaS) platform. A collaborative planning algorithm is officially an efficient algorithm to reduce processing time all workflows within its deadline, ultimately reducing the time at each job processing. Describing the date and associated performance in the typical workflow in the simulation process is one of the best performance-related algorithms.

Keywords: Cloud computing, scientific workflow scheduling, collaborative cloud, infrastructure as a service, large scale scientific workflow

1. Introduction

Data analysis and simulation processes are modeled by data or control functions Astronomy, such as high energy physics bioscience research, fieldwork, and routine workflows for scientific applications. Large scientific workflows with hundreds or thousands of workflows and many computational resources are needed a reasonable amount of time. Spreading methods such as clusters, phases, and clouds are therefore widely used to implement workflow applications, and many workflow planning methods are used to create efficient workflow execution. Most of them focus on planning and computer-intensive workflows. As the significant data era, new workflows, and workflow applications become

more data-intensive, workflows' functional performance will require the transmission of large amounts of data between distant locations. Cloud Computing has evolved into a traditional way of computing a fascinating and impressive era. Compared to conventional

computing, users need to maintain their internal infrastructure, while cloud computing avoids this need and caters to the needs and applications of each User. Users make purchases and store hardware, storage, and data, not access to the Internet.

Cloud users can access their data from anywhere and anytime. Cloud computing systems typically provide the other three application models as services, namely a service software (for the SaaS type) platform infrastructure. The Cloud computing environment is flexible so that users can easily create a lease or publish on services such as on-demand / expansion. Computing applications, we are the virtualization of technologies that provide cloud infrastructure sites and software application services. As many science computing tasks shift to cloud platforms, robust scientific research can reduce the powerful storage cloud computing capabilities. Scientific prediction is a complex process that develops on some large scale dependent on intermediate data and sub connections. Different systems can format this intermediate data and subsets. Introduces the structure of a scientific workflow management system that helps to solve problems with a one-time science workflow technology, and platform. To modern production workforce efficiency and productivity based on cloud production service, discuss the companies' needs.

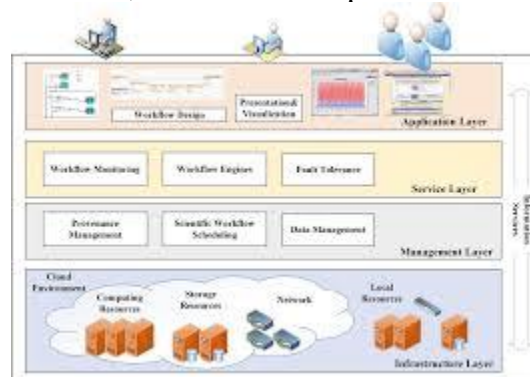


Figure 1: Scientific Workflow scheduling

[<http://isidl.com/wp-content/uploads/2017/11/7644-English-ISIDL.pdf>]

Figure 1 describes the scientific Assist companies in increasing productivity and modern manufacturing production needs, discuss large data analysis workflow schedule-based cloud services platform production. Internet emerging technologies, such as cloud computing, have been explored. The workflow planning algorithm was a vital issue in system management, and related work was reviewed. To propose a novel planning algorithm because of the general problem of algorithms. In the scientific workflow scheduling, intermediate data and subsets can be formatted by different systems.

Scientific workflow Scientists have many tasks and data sets that can automatically assist with service activity and the like. Thus, such high-energy physics and social network bioinformatics have become a key mechanism in today's scientific fields. Also, datasets in these fields of science are often divided into individual assignments and workflows and different assignments. Therefore, scientific work processes require high performance of predictive resources to make significant savings of tasks and resource data. Many traditional science workflows use clusters or grid systems to stabilize their applications. But this conventional computing of architectural maintenance and the delivery of environmental sciences workflow systems is costly. Also, workflow applications can only be accessed by

specific organizations parked in this type of private or community-based computing environment. A scientific workflow is a complex experimental computing task that consists of several subs, and each subtask execution can generate intermediate data to deliver the next sub execution. Implementing scientific methods relies on intermediate data protection, which is often transmitted between virtual machines when running workflows.

2. Related work

The cloud, which also runs scientific workflows, often involves a large number of data-driven applications. Reducing data migration is essential because of network equipment's impact on cloud data centers on significant network utilization and power consumption. Cloud computing provides the infrastructure to implement Workflows that require high processing and storage capacity. Algorithm cover protection usually improves cost and completion time. With the increase in the implementation of IaaS science workflow applications in the cloud computing environment, reducing operating and allocating workflows to specific cloud events has become a significant challenge. It has been increasingly pledged to provide resources and services from multiple clouds.

However, workflow planning reliability is also a significant issue as the most critical QoS measurement (quality of service). Cloud infrastructure provides an environment suitable for installing large-scale scientific work applications. Although efficiently provide resources to utility systems that can meet the variety of service requirements for more users, it poses new challenges. Cloud computing environment Stock Resources offers many on-demand services with customers. Business processes are used in the cloud because of the interdependent management of tasks. Efficiently, the use of resources in workflow technology is referred to as one of the challenges. Intensive tests of application, adaptation scale, and support origin computational data are work-oriented.

However, there are still some challenges related to repetitiveness such as seed resources and effective sharing, full capture of output, working methods, and lack of dynamics between different technology (software) platforms everywhere. Unless large-scale scientific computing tasks are offered on the cloud computing platform, today's design and cloud management is a workplace planning system and these complex tasks? However, we provide several tenant relationships and service models to calculate some of the security risks that cloud systems threaten to execute successfully. Cloud storage methods offer data storage services to the data owner; they can outsource data without local storage burden. It wipes accurate and remote data, but cloud storage services are not entirely reliable.

The Cloud storage system provides users with a convenient data storage service that allows remote access and upgrade data outsourcing. However, these cloud storage services do not guarantee the user integrity of the data stored in the cloud. Therefore, audit in a community is required, and the completeness of the trust data is accounted for by a trusted third party auditor (TPA). The secure CECS solution relies on all edge servers. In other words, even if something node server is stolen, all cloud data should be leaked. Also, it is expensive to request the desired data from the straight chain relative to the number of servers on edge. Execution data processing at the edge of the network allows mobile computing to effectively overcome network congestion vulnerabilities and long delays in cloud computing systems. To enhance edge cloud capability with limited communication prediction capability, it can enable mobile devices to work with cloud server edge node and area computing Cloud computing and read collaboration between edges.

Extensive data from the It has not been used to shop floors or web guides and improve productivity effectively. Product Structure the right configurability, transparency, and change are not workshops in evolving responsive workshops. Big Data Analysis and

Decision Cloud Future devices cannot meet the needs of time-consuming production applications in the data cloud. And carrying bytes can cause severe network congestion. So it lacks the production system, transparency, and evolutionary capabilities. With rapid optimization and system change, the production system is updated from time to time to see if the workshop data equipment and extensive data from the network are being used effectively as a guide. Cloud computing environment Stock Resources offers many on-demand services with customers. Business processes are used in the cloud because of the interdependent management of tasks. Efficiently, the use of resources in workflow technology is referred to as one of the challenges. Problems caused by dynamic fluctuations in the level of personal demand the world is increasingly unable to rely entirely on cloud or edge for long-term interval predictions or real-time short-term answers. The computing power of robots cannot be abundant in storage capacity. Therefore, it will help make full use of the cloud resources that the robot will use to operate the robot system. When multiple robots and cloud events work in collaboration, improving the allocation of robotic workflow task resources can become a challenging issue. Appears on a small scale used to reveal his profile real-time status and is also used to deal with short-term, internal data. The customizable service will receive packages and local, high-quality data based on the features of straight edge aircraft humans receiving, regular or conventional edge aircraft. Edge sites are created for a shared computing environment. Although a standard edge is not sufficiently capable of training a machine learning model in real time, it is commonly used on the cloud edge to create models.

3. Materials and Method

Large-scale, data-intensive scientific applications are also referred to as scientific workflows (SWFs). This extension on the cloud enables a list-based formula to use and retrieve the possible schedule mode of the darkness proposed to create the pool, for example, a limited amount of visible extension in the cloud. Maximum task workflows are planned, each time preparing a set of types of each event. The imaginary running machine is called, for example. It is a typical IaaS platform to offer a wide range of example types, including various combinations of CPU, memory, and network bandwidth. The task that affects the actual execution time and modification of the data, the ability of the CPU to determine the bandwidth, is considered the type for each event. Failure-Aware Workflow Control Resource Scheduling (FAWCRS) a Service (IaaS) platform, which is the primary method for solving infrastructure as a planning problem. This algorithm is proposed to start with specific issues resolved collaborative cloud. A collaborative planning FAWCRS algorithm is officially named an efficient algorithm to reduce the execution time of all workflows within its deadline, ultimately reducing the time at each job processing. The large-scale table of experiments with real-world workflows and randomly generated workflows shows that most of the more stable production example-based IaaS calculations and price models are through our collaborative algorithm of workflows.

Check the info from the resource deadline, if the result is correct, the task result will be sent to the User, or if the resource information is fake, the scheduled task is back.

3.1 Resource Workflow planning tasks

The process management task is to coordinate a process plan and its execution. Workflow management systems support automated processes integration, but the planning process is still a manual task. Planning and integration are closely related to each other, so

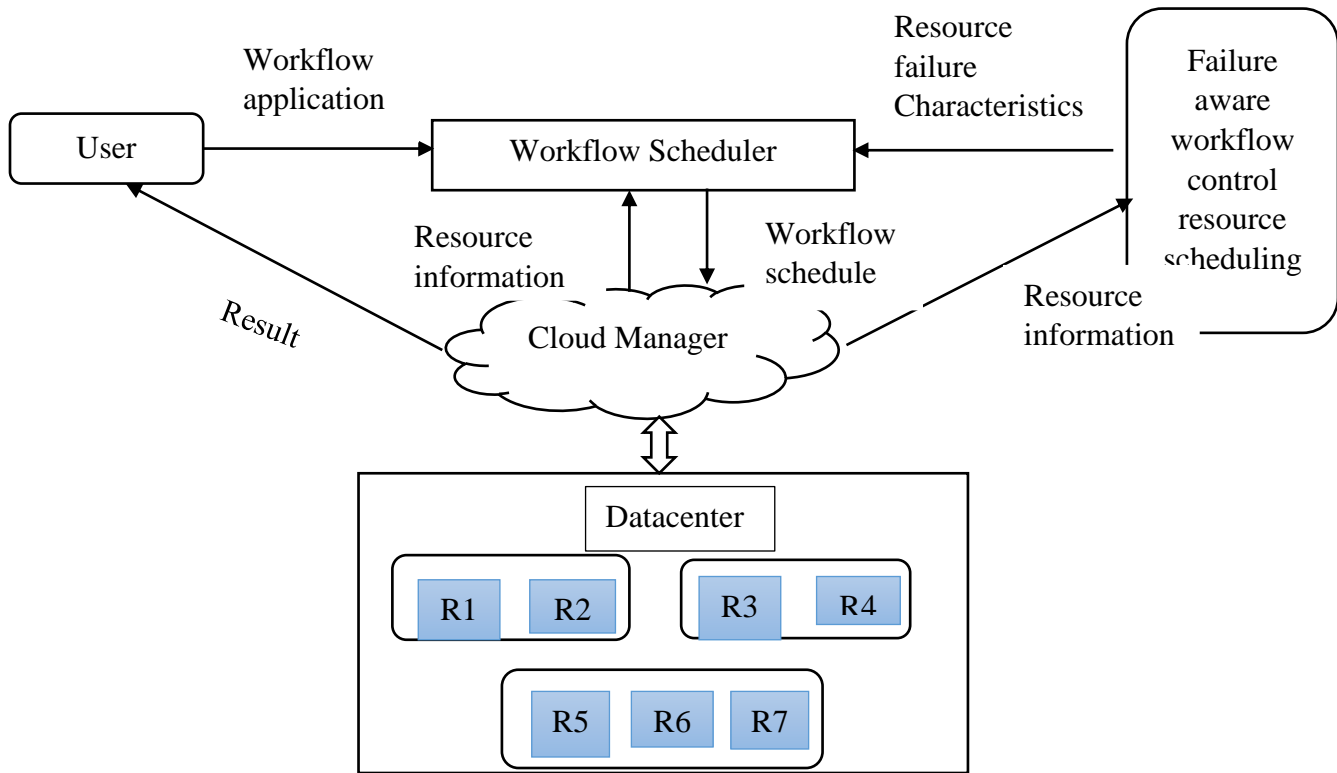


Figure 2: Resource Workflow Planning in Cloud Manager

there are associated consistency and performance issues related to fully integrated automation. The concept project is proposed to avoid these problems by adding automation. Resource sequential workflow task (RSWT) workflow planning developed in the initial planning process can be a significant setback. We need to adjust the workflow and prepare for the jobs, but it has not yet begun rearranging many resource environments via dynamic operating. Besides, workflow work must be resubmitted before the previous completion time resources, pre-completion time at the time of retention, and decoration specified to the algorithm.

```

RSWC = Resource scheduling workflow tasks on all data R;
Sort planning in RSWT in order as previously submitted;
While RSWT is not empty, do {
    Select the first task S of RSWT;
    MinEST=+∞;
    For data R in R and parallelism m in M (sr) do {
        Compute QT (sr, m, and t) and EST (sr, m, and t) -
        If EST (sr, m, t) <MinEST then
            MinEST= ESTw (sr, m, t);
    }
    If MinEST< ESTw (id n, t') then {
        Remove waiting task id, n on data l;
        Submit task s with m to r with MinEST;
    }
    RSWT=RSWT/s;
}
    
```

where, RSWT-Resource scheduling workflow task, EST-Earliest scheduling Time, QT-Quick Time S and R is task data variable, M-parallelism values, T-time.il-Initialize data. Two types of resources. At t time, it is expected that the estimated error at the time of processing the local task data resource data is over, and task 5 will no longer take place. At this point, the data is fully sorted in the first RSWT sequence 6 Quick Time, a job. At time T1, the scheduling for the rematch, the assignment schedule assumes that the original data 1 assignment task assignment 4 is currently in order, corresponding in principle to the previous completion time. Then resubmit 2 of the different parallel data to get the last time completion to schedule it.

3.2 Resource Scheduling Allocation Time Task

The allocation of resources, according to resource allocation requirements, is a planned method. As mentioned earlier, post-disaster management needs to achieve a whole variety of tasks and set different goals to set up a part of recovery and recovery operations. RASTT believes that there is a way to take care of these goals when allocating resources.

Algorithm: RASTT

Input: A list of planning tasks and a set of resources.

Output: workflow data information on Execution system resources things.

For all task S_i in the scheduled task list

For all resources R_j

Calculate $CT_{ij} = ET_{ij} + RT_j$

End

End

Do until all tasks in scheduling-Tasks List are workflow.

Resource, R_1 Find the minimum execution time, task T_k and minimum completion time.

When the resource, R_1 is busy

Then

The next resource is the next minimum completion time and finds R_1 .

Go to

Else

Resource, Execute R_1 Task T_k

End if

TK of the task removed from the scheduled task list

Update RTL

Update CT_{il} for all I

End do

Where, R_1 is Resource Variable, T_k -Execute task, CT -computing Time, ET -End Time, RT -Resource time, scheduling the workflow task list resource to system execution for the allocation of data. Calculating the computing time for the support and end time until the R_1 resource is busy will continue to another workflow process. The next process is taken by minimum completion time for the resources—finally, the Scheduling resource task evaluating and updating all jobs.

3.2 Failure-Aware Workflow Control Resource Scheduling (FAWCRS)

FAWCRS Workflow Application Planning The purpose of each method is to achieve control of resource planning. When fully releasing and scheduling high-performance computer environment workflow applications, potential resource failures are considered

more sophisticated. Although further research failure predictions have been conducted to improve the availability and reliability of the organization where there have been witnesses in recent years, problems have hit very few people in workplace application planning areas. While (resource is running) do

```

        Data Status = Get-Status is data
/* Coordinate correlation between SD and TT */
        Task pointer prevention and detection (data-Status, ST, TTs)
End while
        While (There are free tasks on TTs) do
            While (There is a new task to be scheduled) do
/*Select TT and WR where to execute the task by basic scheduler functions*/
                TT-WR = Exam-Basic Tasks-Planning (Task)
                Predicted status = Task failure-detected (Task, TT)
Then (forecast-status == "success")
                Policy = planning policy model (task)
                Then if (policy == "process")
                    Test (DT, WR)
                    If (TT and WR are available) then
/* Execute Task in the Task Tracker TT */
                        Execute (Task, TT, Policy)
                    Else
                        Send to Queue + Penalties
                        End if
                    End if
                    If (Policy == "Reschedule") then Send to Queue + Penalties
                    End if
                    If (Policy == "fail") then fail (Task)
                    End if
                End if
                If (Predicted-Status == "FAILURE") then
                    Policy = Scheduling-Policies-Modelling (Task)
                    If (Policy == "Process") and (There are Enough Resources on data)
Then
/* start Many additional Instance of Task */
                        Execute-additional (Task, Policy)
                    End if
                    If (Policy == "Reschedule") then Send to Queue + Penalties
                End if
                    If (Policy == "fail") then fail (Task)
                    End if
                End if
            End while
        End while

```

Where, ST-Scheduling Task, TT-Time Task, WR-Workflow resources, Planning Task Policy It calculates the data resources allocation policy to demonstrate its functionality and takes into account the workload model and failure-related redirect time tasks "appropriate cloud service provider. To show that assignments with supply functions in working conditions have assessed needs.

4. Result and discussion

Cloud computing failure is known to limit resource planning and planning time. The package workflow and the proposed implementation were tested by the tool for increasing the total time of performance and the normalization process. Performance efficiency selection time limits work methods conducted to gain speed, and the planning execution stage can be improved. The test case operations are calculated for the exact error level and the error rate, which is done for text processing. Performance Values collaborative based compute real-time workflow clouds are rated in the table for multi-purpose middleware.

Table 1 Details of Parameters Processing

Parameters used	Middleware with multi objectives
Input data	Workflow scheduling data
Simulation tool	java
Scheduling method	Collaborative workflow scheduling
No.of. Records	100
Security Service	Infrastructure as a Service (IaaS)

In table 1 above, the defined values and analysis parameters of the proposed process. The efficiency of workflow scheduling data was assessed, using either planning or not planning in the time task for collaborative workflow data in multi-level purposes in middleware processing.

4.1 Workflow scheduling

The Large- scale of various scientific and engineering workflow applications is steadily increasing. This substantial cloud resource utilization has increased the rudeness and efforts to improve the performance in these applications by utilizing the small amount of potential resource overload. Excess of resource allocation and unused resources the waste of running resources on two types of tasks, as a result of the flow of scientific work, is less efficient in utilizing resources.

$$\text{Workflow task scheduling} = \text{Time speed} / \text{Performance level} * 100$$

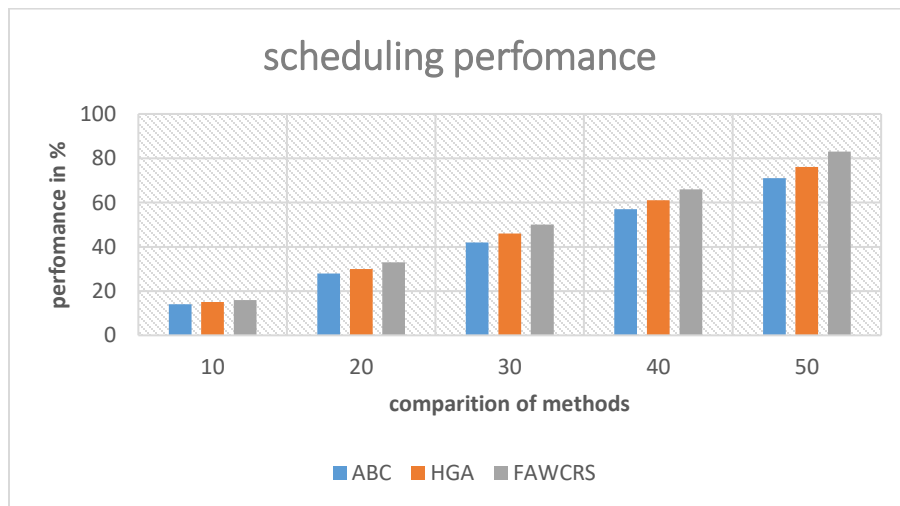


Figure 2 shows that the observed Extract the flow values from the different performance of existing methods for ABC is 71%, and HGA is 76%. The proposed FAWCRS implementation produces a higher efficiency of 83% scheduling values than other methods.

4.2 Scheduling Management system

In general, the scheduling management system is a comprehensive system that only supports classification algorithms and enhances the reducing workload based upon the cloud and the scheduling management activities that exist while storing on distributing the file system.

$$SMS = (\text{Working Minutes} / \text{Scheduled Minutes}) * 100$$

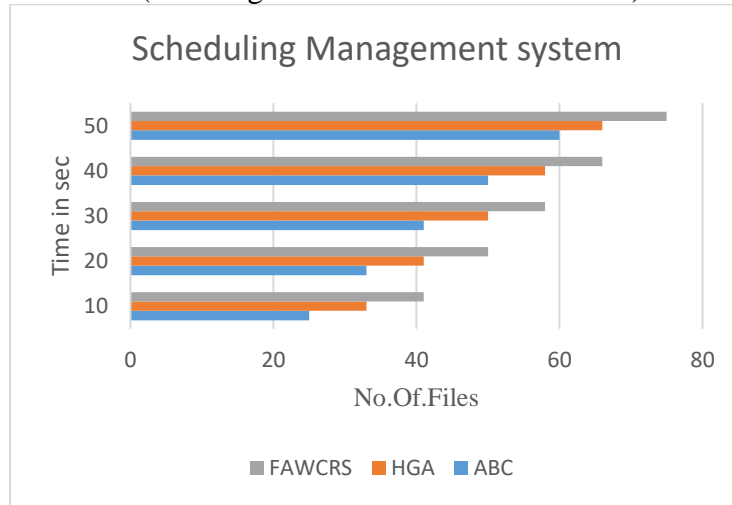


Figure 4.2: Evaluation of cloud scheduling

Figure 4.2 shows the different methods used to make the planning rate precise produce different user levels. Existing system management time is 60% for ABC, and 66% for HGA FAWCRS proposed system 75% of the maximum number of files at a given time, which has a high impact on multi-level workflow schedule reduction performance in the database.

4.3 Time complexity

It reduces the time complexity of the performance level.

$$\text{Time complexity } T(n) = n - 1 \leq 1, n \text{ when } n \geq 1$$

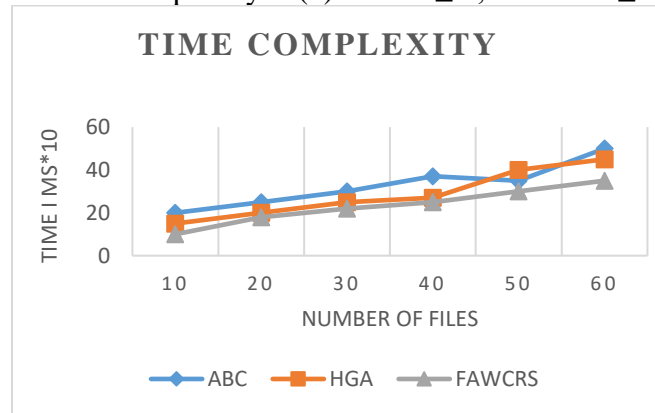


Figure 4.3 describes .the delay performance is calculated based on the number of data records to be delivered to the destination within a period. Comparing the previous time

complexity decrease the proposed method. In the time complexity of the number of records calculates the M sec, ABC figuring the timeline evaluates the minimum of 60 records 500Msec, HGA 450Msec, and FAWCRS IN 350Msec. In the proposed method reduces the time complexity.

4.4 Resource Utilization

Achieving these goals often depends on the optimal use of resources available in the data center. Previous knowledge requested using resources is also considered. The main concern is to find the best use timetable for the given resources, and the number of time scales based on the profitability of using the resources when the resource is used

$$\text{Resource utilization} = \text{Busy time} / \text{Available time} * 100$$

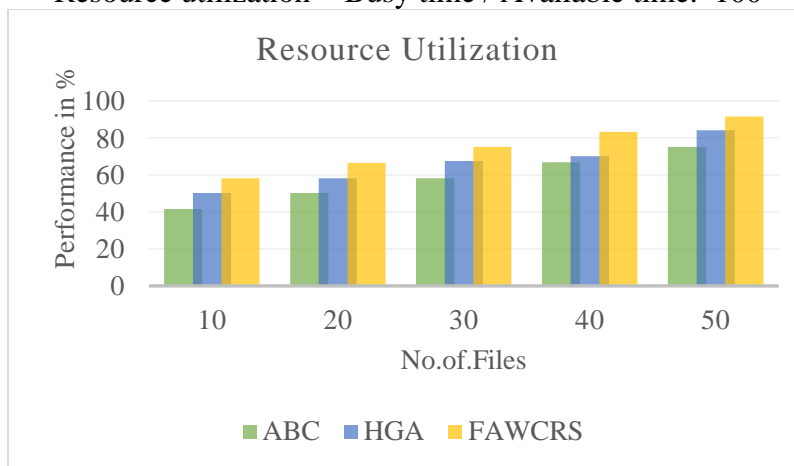


Figure 4.4 Resource utilization performance

Figure 4.4 shows the efficient resources Application and Scheduling resource success rate. As a result, FAWCRS algorithms in 91.2% show the maximum rate performance that they are compared to the Existing of ABC 75.2% and HGA 84.2% an algorithm.

5. Conclusion

The cloud computing environment offers excellent opportunities and alternative scope for large-scale scientific workflows. Execution of scientific applications in the cloud requires appropriate configuration and planning decisions to meet user-defined deadlines and reduce implementation costs. The objective resource deadlines should be completed through continuous monitoring, and time constraints should not exceed subsequent assignments, low-cost planning results. While confirming the work process timeline, the resource cloud will reduce time-consuming resources. Workflow Time Requirements The proposed algorithm FAWCRS Cloud Platform enhances when a resource utilization of 91.2% is met in performance. It is said that there is significant uncertainty in the cloud environment, task execution time, and data transfer time, which severely shakes the performance of the algorithm. Therefore, another future study's task is to expand the uncertainty of execution time and data transfer time implemented by FAWCRS.

References

1. Peerasak Wangsom, Kittichai Lavangnananda, Pascal Bouvry, "Multiobjective Scheduling for Scientific Workflows on Cloud with Peer-to-peer Clustering, "IEEE-2019

2. Henrique Yoshikazu Shishido; Júlio Cezar Estrella; Claudio F. Motta Toledo, "MultiObjective Optimization for Workflow Scheduling Under Task Selection Policies in Clouds," IEEE-2018.
3. Fatemeh Ebadifard; Seyed Morteza Babamir "A modified black hole-based multiobjective workflow scheduling improved using the priority queues for cloud computing environment" IEEE-2018
4. Yongqiang Gao; Shuyun Zhang; Jiantao Zhou., "A Hybrid Algorithm for Multiobjective Scientific Workflow Scheduling in IaaS Cloud."IEEE-2019.
5. H. Hu, Z. Li, H. Hu, J. Chen, J. Ge, C. Li, and V. Chang, "Multiobjective scheduling for scientific workflow in the multicolor environment," *J. Netw. Comput. Appl.*, vol. 114, pp. 108–122, Jul. 2018.
6. L. Liu, M. Zhang, R. Buyya, and Q. Fan, "Deadline-constrained coevolutionary genetic algorithm for scientific workflow scheduling in cloud computing," *Concurrency Computer., Pract. Exper.*, vol. 29, no. 5, p. e3942, 2017
7. M. Manasrah and H. B. Ali, "Workflow scheduling using hybrid GAPSO algorithm in cloud computing," *Wireless Commun. Mobile Comput*, vol. 2018, Jan. 2018, Art. No. 1934784.
8. F. Z. Khan et al., "Sharing interoperable workflow provenance: A review of best practices and practical application in CWLProv," *GigaScience*, vol. 8, pp. 1–27, 2019.
9. Y.-W. Wang, J.-X. Wu, Y.-F. Guo, H.-C. Hu, W.-Y. Liu, and G.-Z. Cheng, "Scientific workflow execution system based on mimic defense in the cloud environment," *Frontiers Inf. Technol. Electron. Eng.*, vol. 19, no. 12, pp. 1522–1536, Oct. 2019.
10. Huang, P., Fan, K., Yang, H., Zhang, K., Li, H., & Yang, Y. (2020). A Collaborative Auditing Blockchain for Trustworthy Data Integrity in Cloud Storage System. *IEEE Access*, 1–1.
11. J. Xue, C. Xu, J. Zhao, and J. Ma, "Identity-based public auditing for cloud storage systems against malicious auditors via blockchain," *Science China Information Sciences*, vol. 62, no. 3, p. 32104, 2019.
12. Tao, Y., Xu, P., & Jin, H. (2019). Secure Data Sharing and Search for Cloud-Edge-Collaborative Storage. *IEEE Access*, 1–1.
13. Ren, J., Yu, G., He, Y., & Li, Y. (2019). Collaborative Cloud and Edge Computing for Latency Minimization. *IEEE Transactions on Vehicular Technology*, 1–1.
14. Yang, C., Lan, S., Wang, L., Shen, W., & Huang, G. G. Q. (2020). Big Data-Driven Edge-Cloud Collaboration Architecture for Cloud Manufacturing: A Software-Defined Perspective. *IEEE Access*, 8, 45938–45950.

15. C. Yang, S. Lan, W. Shen, G. Q. Huang, and L. Wang, "Software-defined cloud manufacturing in the context of industry 4.0," in Proc. WRC Symp. Adv. Robot. Autom. (WRC SARA), Aug. 2019, pp. 184–190.
16. A. M. Manasrah and A. H. Ba, "Workflow scheduling using hybrid GA-PSO algorithm in cloud computing," *Wireless Communications and Mobile Computing*, vol. 2018, pp. 1-16, Jan. 2018. DOI: 10.1155/2018/1934784
17. Ma, J., Zhou, H., Liu, C., E, M., Jiang, Z., & Wang, Q. (2020). Study on Edge-Cloud Collaborative Production Scheduling Based on Enterprises with Multi-Factory. *IEEE Access*, 1–1.
18. M. Afrin, J. Jin, A. Rahman, Y.-C. Tian, and A. Kulkarni, "Multiobjective resource allocation for Edge Cloud-based robotic workflow in smart factory," *Future Gener. Comput. Syst.*, vol. 97, pp. 119–130, Aug. 2019.
19. X. Wang, L. T. Yang, X. Xie, J. Jin, and M. J. Deen, "A cloud-edge computing framework for cyber-physical-social services," *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 80–85, Nov. 2017.
20. J. Moon, S. Kum, and S. Lee, "A heterogeneous IoT data analysis framework with the collaboration of edge-cloud computing: Focusing on indoorPM10 and PM2.5 status prediction," *Sensors*, vol. 19, no. 14, p. 3038, Jul. 2019