Efficient task scheduling in Cloud Computing using IDEA based QoS Aware Task Scheduling

Nishan Singh[1], Dr Jatinder Singh Saini[2]
[1] *Baba Banda Singh Bahadur Engineering College*
[2] *Baba Banda Singh Bahadur Engineering College*
(E-mail: nishansinghkaler786@gmail.com[1], sainijatinder@gmail.com[2])

Abstract— The method by which an operating system (OS) allots resources inside a system to a variety of different activities is referred to as task scheduling. The system is responsible for managing prioritized job queues that are currently waiting for CPU time. The system should decide which work should be chosen from which queue and how much time should be allotted for the job. Job Scheduling in Cloud Computing is an essential activity in this industry to ensure that resources are used in the most effective manner possible. The CPU execution time and the total amount of memory (allocated memory) required by all tasks that are planned to be performed in the fog computing infrastructure are the two performance evaluation metrics that are utilized in fog computing to solve this issue. These are the CPU execution time and the total amount of memory (allocated memory). Both the make span and the reaction time for the job scheduling methods that were already in use fell short of expectations. Therefore, the IDEA-based EDA is used in the suggested methodology for the purpose of improving make span as well as reaction time. Because IDEA provides numerous options, it is possible to choose the best one in terms of reaction time, allowing for the projects with a shorter deadline range to be executed while still ensuring quality of service. It has been shown in the results section that the suggested strategy is a significant improvement over the approach that is currently being used.

Keywords—*Cloud Computing, Scheduling, IDEA*

I. INTRODUCTION

Cloud computing is one of the developing technologies that is used to manage data, platforms, and infrastructure these days since it provides a great deal of flexibility and features such as storage, sharing, and accessibility. Cloud computing is one of the rising technologies. In cloud computing, the performance of the capabilities it offers may be affected by a variety of factors, including the scheduling of work, energy consumption, and processing speed, among other things [1].

The concept of computing in the cloud is gaining tremendous traction and popularity as a result of its capacity to provide organisations with more adaptability and flexibility. The term "cloud computing" refers to a number of different types of clouds, depending on the adaptability of the resources being pooled together. [1]

Clouds may be broken down into four categories: public, private, network, and hybrid. It is the responsibility of the cloud provider to ensure that public clouds are available to the general public in a manner that does not need payment. Private clouds are tailored specifically to the needs of a company or an organisation, and their functionality may be limited by either the or-ganisation or an independent third party. In people group clouds, a few different associations work together to share the cloud's underlying infrastructure in order to assist a specific network with typical issues. The term "hybrid cloud" refers to a combination of public, private, and net-work clouds. [2][3].

A. *Task Scheduling*

Scheduling is a phrase that refers to the planning that goes into assigning work to different virtual machines, which may be used in cloud computing or general computing. Virtual Machines are installed on hosts, where they then proceed to carry out the duties that have been delegated to them.

When it comes to task scheduling, a scheduler is sometimes referred to as the "heart" of the process since they are responsible for all of the activities that are involved in allocating tasks. This is something that is used on a regular basis due to the fact that the tasks are to be allocated to Virtual Machines and the assets are to be shared.[2] A scheduler is responsible for completing all of these duties in accordance with certain planning, which is referred to as Task Scheduling Strategies. The ability to efficiently schedule tasks and manage resources enables service providers to maximize both income and resource utilization to their fullest extent. In actuality, the scheduling and distribution of resources are significant obstacles to overcome in terms of the performance of cloud computing's available resources. This is one of the primary reasons why scholars have been interested in the study of task scheduling in cloud computing. The process of scheduling tasks is known as task scheduling, and it involves organizing incoming requests (tasks) in a certain way in order to ensure that the available resources are used in an effective manner.

B. *Need of Task Scheduling*

The ability to efficiently schedule tasks and manage resources enables service providers to maximise both income and resource utilisation to their fullest extent. In actuality, the scheduling and distribution of resources are significant obstacles to overcome in terms of the performance of cloud computing's available resources. This is one of the main reasons why scholars have been interested in the study of task scheduling in cloud computing.[3] Scheduling tasks refers to the practise of organising incoming requests (tasks) in a certain way in order to ensure that the available resources are used in the most effective manner possible. Users of cloud computing services are required to make their requests available over the Internet since cloud computing is the technology that facilitates the delivery of services via this medium. It is possible for a lot of re-quests (tasks) to be created all at once due to the fact that each service has a number of users. Systems that do not use scheduling may include lengthier waiting times for jobs. Additionally, certain short-term tasks may be terminated as a result of the waiting period. When scheduling, the scheduler has to take into consideration a variety of restrictions, such as the type of the work, the size of the task, the amount of time it will take to complete the task, the availability of resources, the task queue, and the amount of demand that is being placed on the resources. One of the most fundamental challenges in cloud computing is the scheduling of tasks. The effective scheduling of tasks may lead to the more effective use of available re-sources. One of the most significant benefits of cloud computing is that it encourages efficient use of available resources [3]. Therefore, the scheduling of tasks and the distribution of re-sources are two sides of the same coin. Both have an effect on one another. Internet users don't need to be concerned about the hosting infrastructure in order to access material these days, since it can be accessed from anywhere and at any time. This kind of hosting infrastructure is made up of a variety of computers, each of which has its own unique set of capabilities. The service provider is in charge of maintaining and managing all of these machines. Computing in the cloud expands the possibilities of the aforementioned infrastructure, which is able to connect to the Internet. Users of cloud services generate revenue for cloud service providers by purchasing the providers' services.

C. Classification of Task Scheduling

Task scheduling algorithms can be classified as follows:

Immediate Scheduling: Whenever a work of this sort is submitted, it is instantly given to VM.

Batch Scheduling: After the tasks have been compiled into a batch, the batch is subsequently sent to the virtual machine.

Static Scheduling: When using this method of scheduling, the information about the global state is already known to the system. It does not take into account the run time events of VMs, but rather it distributes resources across VMs and workloads in a round robin fashion or in any other scheduling approach.

Dynamic Scheduling: In this scheduling, the status of the VM at run time is known, and this scheduling did not take into consideration any sort of previous knowledge about the system's global state. This strategy allots resources to VMs in accordance with their existing states.

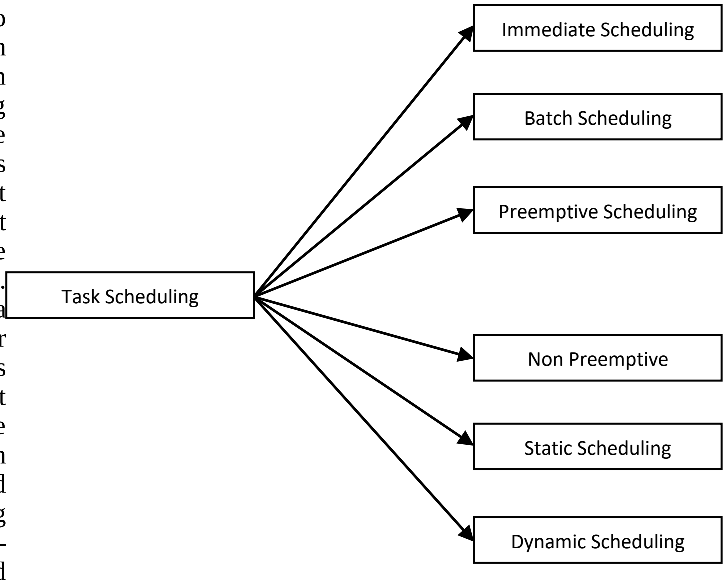


Figure 1: Types of Scheduling

Preemptive Scheduling: In this configuration, the task that is being performed temporarily halts the operation being carried out and may be replaced by another.

Non Preemptive Scheduling.

Non Preemptive Scheduling: This plan does not permit any kind of replacement or rescheduling of system components in any way.

D. Level of Task Scheduling

The process of job scheduling in cloud computing has progressed through the following three levels:

First level: This level consists of a collection of tasks that must be completed before the cloudlet may be sent for execution.

Second Level: At this level, the tasks are mapped to the VMs and resources that are available in order to achieve the highest possible levels of efficiency and make span.

Third Level: At this stage, the tasks are getting ready to be carried out, as seen in fig 2.

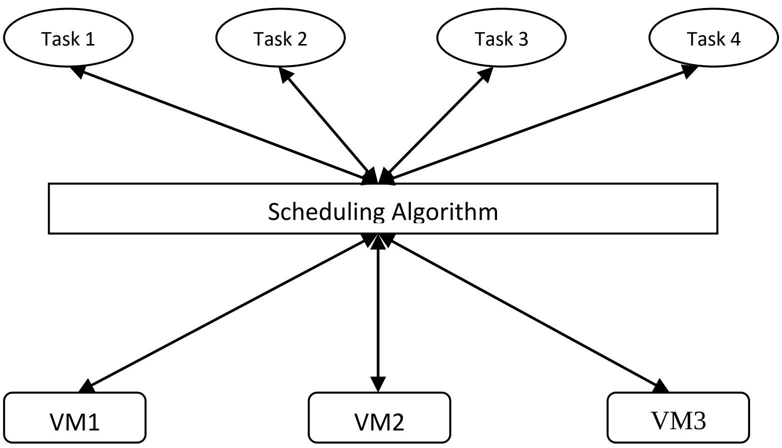


Figure 2: Third Level

Now third step is divided into following steps:

- The first thing that it will do is find all of the VMs that are currently running on the system and gather the status information that is supplied by the data centre broker.

The next step involves the appropriate VM carrying out the work.

II. Literature Survey

An EDA-GA based approach was described by **Pang, S., et al. [1]**, in which results are first creat-ed with the aid of EDA, which stands for Estimation of Distribution Algorithm, and then the re-sults of EDA are fed to the Genetic Algorithm. This process is repeated until the desired solution is found. The Genetic Algorithm is a bio-inspired algorithm in which the mutation and cross over rate are processed on the population formed by the EDA. Following this, a solution using the above described operations is developed, and this is referred to as the most optimum solu-tion.

Improved Particle Swarm Optimisation was a strategy that was suggested and developed by **Saleh, H., et al. [2]** for the allocation of all parameters in cloud architecture. Adjusting the size of the memory heap, which is responsible for increasing make and lower QoS metrics, is made easier with the aid of the IPSO approach. Therefore, an IPSO-based approach may control the heap size in RAM and perform better in all areas of Quality of Service (QoS).

The researchers lead by Sudheer, **M.S., and others [3]** created a method that makes use of par-ticle swarm optimisation. This technique assists the cloud in sharing bandwidth and memory in a decent manner with the tasks that are to be scheduled. The make span and several other quality of service metrics may be increased to some amount by distributing the memory and bandwidth of the resources among the task scheduler in an equitable manner.

[4] Devarasetty, P., and colleagues presented the assignment of system which was created uti-lising the knowledge that was obtained from previous experience that is broken down and de-pendent upon QoS factors. According to the collected data, the algorithms are selected based on the criteria of cost, amount of time, and level of complexity. Therefore, an algorithm is put into operation, which, in terms of both cost and amount of time, produces better outcomes.

Cost-Time Trade off Efficient Workflow Scheduling was the name given to the scheduling meth-od that was suggested by **Djamegni, C.T., et al. [5]**. According to the definition, this algorithm consists of four stages. Task Selection, Evaluation of the Implicitly Requested Instance Types Range (IRITR), Evaluation of the Spare Budget, and Virtual Machine Selection. In this activation function, the goal is to extract the most appropriate VMs Instance type in order to prevent overbidding and underbidding, both of which have the potential to lower costs and increase the effectiveness of VMs.

Koodziej, E. and colleagues [6] establish a unique model for cloud scheduling that is based upon the blockchain method. This approach is different from other contemporary ways in that it at-tempted to offload the implementation of the blockchain modules, which is something that oth-er contemporary techniques do not do. The technology known as block chain is used to provide protection for all of the transactions and access to the data stored in the cloud. As a result, the use of this method offers a system for work scheduling in the cloud that is both safe and effec-tive and is based on the make span.

The authors **Sen, P. et al. [7]** suggested a method for scheduling tasks in web administrations that use a multi-cloud architecture. This method focuses on how to get the most successful out-comes possible, as opposed to selecting a specific cloud's level of performance. In this method, a distributed cloud system is used for job scheduling, and the participants work together to ac-complish this. As a result, the workload is reduced, and the span is increased.

The authors **Kaur, R., and Kaur, A. [8]** presented a better method for scheduling assignments by analysing the many different job scheduling approaches available in cloud computing. The new approach leads to an improvement in the QoS parameters of virtual machines (VMs). This method guarantees improved outcomes since it eliminates the downsides that are associated with scheduling problems and the methods of task scheduling.

A method that is both efficient in terms of its use of energy and based on Q learning was sug-gested by **Zeng, J. and colleagues [9].** It may be broken down into two stages. During the initial phase of development, a centralised task dispatcher is used to construct an M/M/S queue. Dur-ing the second phase, the newly incoming jobs are distributed among the servers using a priori-tisation scheme that takes into account both the task's level of difficulty and the amount of time it will take to complete. After that, a policy that is always being updated is applied to the pro-cess of delegating jobs to virtual machines.

Chong, C.K. and colleagues [10] suggested an approach that tries to enhance the performance of findings by lowering the amount of noisy data that is present in the initial population. This study made use of the Kalman filter in order to lower the amount of noisy data, as well as the error rate, and then to increase the effectiveness of this approach. When noisy data in the pop-ulation is filtered out using the Kalman filter, the number of possible permutations is cut down, which in turn reduces the amount of time needed to carry out the operation.

An efficient priority and relative distance (EPRD) method with two phases has been proposed by **Zang, L., et al. [11]**. In the first step, a task priority queue was created, and then, in the second phase, a virtual machine (VM) would be mapped to the task based on its distance from the job. This strategy enhanced VM utilisation and scheduling efficiency, and it minimised the amount of time needed to schedule tasks for workflow applications that were bound by precedence. All of this was accomplished without breaking the end-to-end deadline.

Sanaj, M.S.; Prathap, P.M.Joe; [12] propose a chaotic squirrel search algorithm (CSSA) that is capable of doing scheduling of many tasks in IAAS in the most effective way possible. This tech-nique generates a task plan in an ongoing way, which makes the existing approach more effi-cient and hence more cost effective. This technique synthesises messy local search with Squirrel search algorithm (SSA) methods to allow explor-ing authority and supplement Squirrel search algorithm (SSA) algorithms.

The best-worst multi-criteria decision-making technique (BWM) and the compromise ranking method (VIKOR) were integrated in **Refieyan, E.; et al.'s [13]** proposal for a new adaptive meth-od, which was described as a combination of the two. The importance of the work is deter-mined by VI-KOR, who serves in the role of decision maker. This strategy is able to provide load balancing and optimise Quality of Service needs since it uses both of these approaches.

A method for load scheduling called the Hybrid Genetic-Gravitational Search Algorithm (HG-GSA) was suggested by **Chaudhary, D. and Kumar, B. [14]**. This method is used to lessen the overall amount of money spent on calculation. The hybrid crossover approach, which is based on the gravitational search algorithm, is what it uses to search for the most optimal location of a particle inside the search space. When determining the force with which the priority will be selected and the job that is to be sent to the VM, the best position is utilised as the input.

Aloboud, E., and Kurdi, H. [15] devised a scheduling method that took inspiration from the para-sitic behaviour of cuckoo birds. This algorithm, which is based on the behaviour of cuckoo birds, also adapts the scheme in which high priority activities are done first and resources are provid-ed to them, while low priority operations are allowed to wait. This approach enhances perfor-mance in terms of both the average percentage of CPU time that is being used and the average amount of time it takes to complete each kind of activity.

Min-Min and Max-Min are the names of the heuristic-based scheduling algorithms that were presented by **Sharma N.** and his colleagues in [16]. The cloud simulation simulator schedules a variety of tasks in order to compare the outcomes of these two algorithms by taking into ac-count the make span parameters. This comparison is carried out in order to determine which method is superior. The comparative research came to the conclusion that min-min should be used when there are a greater number of small-sized jobs than large-sized tasks to complete, but cloud computing should be used when there are a greater number of large-sized tasks.

The scheduling strategy that makes use of PSO was given by **Miglani, N.; et al. [17]**. This review is carried out based on the change that was made to the PSO technique by the manipulation of the parameters that are based on the QoS elements from the very beginning. The PSO tech-nique may be modified to get better outcomes in terms of task assignment, which in turn helps to minimise the amount of time needed for work completion.

III. METHODOLOGY

Estimation of Distribution Algorithm using Genetic Algorithm (EDA-GA) is an algorithm that was developed to check the multi-objective task scheduling issue. This problem may be solved by lowering the amount of time it takes to carry out the tasks that have been allocated. The steps listed below are to be used during this procedure.[1]In the first place, the EDA is put to use in order to provide some workable solutions that may be used in the job scheduling process. The result of the EDA will be sent on to the Genetic Algorithm once it has generated a new solution based on the solution that was picked by the EDA in order to determine which solution is the most optimum.[1]. The Genetic Algorithm The Genetic Algorithm is currently the most prominent algorithm being developed in the field of Artificial Intelligence. This algorithm derived its concept from Charles Darwin's theory of evolution, which allowed for the survival of the fittest. This concept genuinely allots the resources to the various activities according to the fitness function that each one has. In this configuration, tasks are not carried out in their entirety but rather in accordance with the specifications of the tasks.

Improved Differential Evolution Algorithm is an algorithm that uses the Kalman filter to minimise the noise in the population, which in turn reduces the error rate in results. As a consequence, the algorithm generates results in a shorter amount of time, and it is able to circumvent the limitations that are imposed by the Genetic Algorithm.[10] The EDA method will first produce a set of solutions using Estimation Distribution as part of the EDA-IDEA scheme that has been suggested. After this step, the collection of solutions, which was produced by EDA, is sent to IDEA, and IDEA, in turn, provides the solution that is the most optimum out of all of them.[10] Therefore, by using a hybrid system consisting of EDA and Improved Differential Algorithm, the issues that are present in the original method may be circumvented.

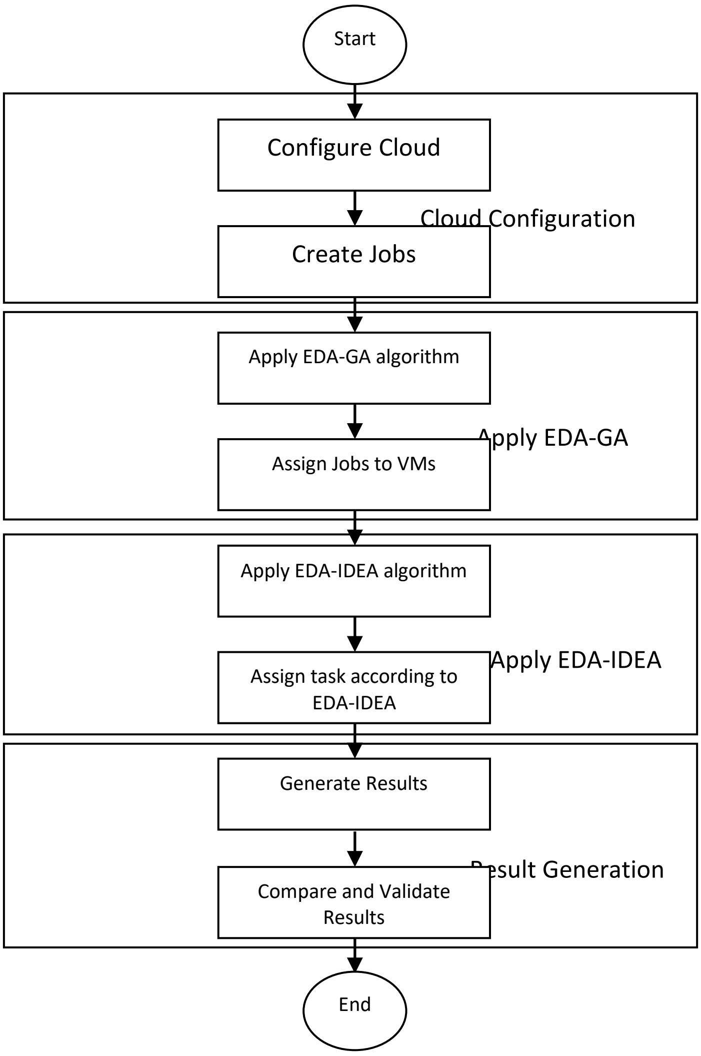
The stages involved in doing research are shown in flowchart form in Figure 3, which may be seen here. Figure 4 illustrates this point.1 comprehensive piece of study is broken up into four sections.

The first section will cover the configuration of the cloud.

In this section, the configuration of the Host machine, which is the actual machine, takes place. RAM, solid-state drives (SSD), bandwidth, and processors are the components that make up a host machine's configuration.

After the host system has been configured, virtual machines (VMs) are deployed on each ma-chine so that tasks may be carried out on the VMs. Virtual machines (VMs) may be configured by having resources allotted to them from the physical computer or the host machine. The total number of resources that may be allocated to Virtual Machines (VMs) cannot be more than the total number of resources that can be possessed by a Physical Machine.

The second phase of this process involves running the current method, which is an EDA-GA al-gorithm, on virtual machines. The EDA-GA method, which is described in more detail below, will be used to carry out the jobs that have been allocated to each VM.



Algorithm:

QOS AWARE TASK SCHEDULING-IDEA

Input: Tasks (T), Virtual Machines (VMs)

Output: Optimal Solution for Scheduling of tasks

Initialize model of probability

while i <= imax do

Sample Values of probability model for generation of population

Initialize population P

[1]. WHILE (l<max)

[2]. FOR (j = 0; j < NP; j++)

[3]. Select P[j1], P[j2], P[j3]

[4]. Mutation

[5]. Choose Initial Candidate I[i] = P[j1] + F *(P [j2] - P [j3]).

[6]. Crossover

[7]. Choose final candidate C[i] by crossing over the genes of P [j] and I1[j] as follows:

[8]. FOR (j1 = 0 ; j1 < NP ; j1++)

a. IF (U1(0, 1) < CR)

i. I[j][j1] = I1[j][j1]s

b. ELSE

i. I[j][j1] = P[j][j1]

[9]. END-FOR

[10]. Updating Population

a. I[j] = inv(inv(I[j]) + I)

b. I= P*H1'*inv(H1*P*H1'+R)

[11]. STEP 1.5: Evaluate I[j]

[12]. IF (I[j] is better than P [j])

[13]. P'[j] = I[j]

[14]. ELSE

[15]. P'[j] = P [j]

[16]. END-IF

[17]. END-FOR

[18]. P = P

[19]. Use excellent population for updation of probability model

[20]. i++

[21]. END-WHILE

[22]. END

[23]. K = Kalman gain value,

[24]. H = observation matrix,

[25]. H' = inverse of matrix H,

[26]. P = population

[27]. P' = population the next generation

[28]. I = candidate solution,

[29]. Using the above defined IDEA optimization algorithm the drawbacks of GA are overcome and GA is replaced with IDEA.

IV. Result and Comparison

Make Span: You may define a make span as an ideal duration for which a virtual machine is optimal when a task is run and the next one is yet to be allocated. This occurs when a new task has not yet been assigned. In cloud computing, the make span of a system should be kept to a minimum. If the make span is low, the total execution time of the system will be low as well. This indicates that the tasks have been completed by the system in a shorter amount of time.

Response Time: The term "response time" refers to the amount of time required for a Virtual Machine to fully carry out the instructions for a given job in its entirety. It is ideal for a system to have a response time that is as short as feasible. A system with a low reaction time has completed a work in a shorter amount of time, which in turn leads to an improvement in the system's overall performance.

Utilisation Above Capacity: The tasks that are allocated to a VM while its own quota is being carried out are one way to characterise what is known as "over utilisation" of that VM. Another definition of this term refers to the migrated tasks that are handed off to a virtual machine.

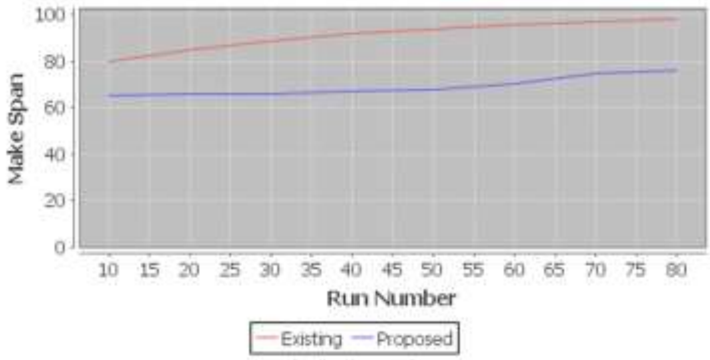


Fig 4: Cloudlet Size vs Make span

Fig. 4 is a depiction of a comparison study between the current technique, which is called EDA-GA, and the new approach, which is called EDA-IDEA. The graph depicts the value of the make span in comparison to the cloudlet size. The term "make span" refers to the period of time dur-ing which a virtual machine performs at its optimal level, which occurs after a job has been completed but before a new one is assigned.

In comparison to the EDA-GA based scheduling system, the make span of the EDA-IDEA based scheduling strategy is shown in Figure 5. with EDA-GA scheduling, the make span is around 98 milliseconds, but with EDA-IDEA scheduling, it is less than 80 milliseconds.

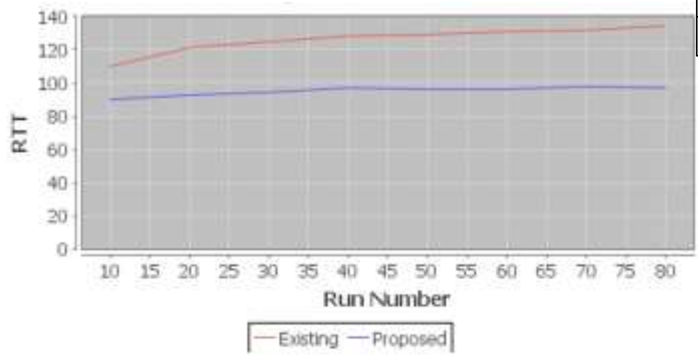


Fig 5: Response Time vs Jobs

Fig. 5 is a depiction of a comparison study between the current technique, which is called EDA-GA, and the new approach, which is called EDA-IDEA. The value of the reaction time is shown against the number of jobs that have been allocated in the graph. The term "make span" refers to the period of time during which a virtual machine performs at its optimal level, which occurs after a job has been completed but before a new one is assigned. Figure 5 illustrates the differ-ence in reaction time between the EDA-IDEA based scheduling method and the EDA-GA scheme. The response time for EDA-GA scheduling is around 137 milliseconds, however for EDA-IDEA it is less than 100 milliseconds.

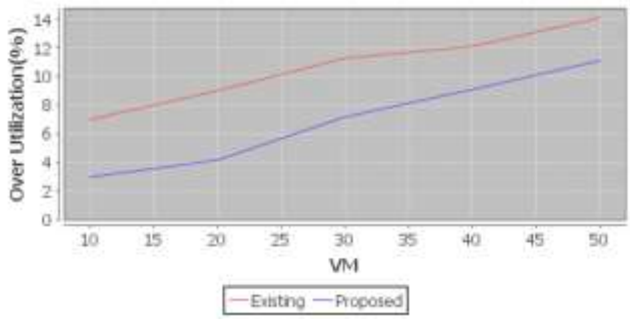


Fig 6: Over Utilization

Fig. 6 is a depiction of a comparison study between the current technique, which is called EDA-GA, and the new approach, which is called EDA-IDEA. The Over utilisation value is shown against the number of Jobs that have been allocated in the graph. The tasks that are allocated to a VM while its own quota is being carried out are one way to characterise what is known as "over utilisation" of that VM. It may also be understood to refer to the tasks that have been moved and then assigned to a virtual machine. In comparison to EDA-GA, Fig. 6 demonstrates that EDA-IDEA-based scheduling schemes are used much more often. When it comes to sched-uling, EDA-GA has an overutilization rate of around 14%, but EDA-IDEA has a rate of less than 12%.

Table 1: Comparison of existing and proposed scheduling algorithm

Technique / Parameters	Existing	Proposed
Make span	98	79
Over Utilization	14	11
Response Time(RTT)	130	100

A comparison between the current schedule with the proposed scheduling is shown in Table 1. The makespan of the existing approach is 98 milliseconds, whereas that of the new technique is 79. In the case of the present system, there are 14 instances of overutilization, whereas the suggested system only has 11. The response time for the conventional approach is 130 seconds, whereas the suggested technique takes just 100 seconds.

V. CONCLUSION

This execution is pointing towards the basis of executing subjective examination on make range in VM task allocation and process according to their cutoff time, which was then actualized in CloudSim using the Java programming language. In this section, a substantial amount of em-phasis is placed on the research of dead-line-based errand planning calculation with heteroge-neous cloud-based assets, which is then followed by an in-depth evaluation of various calcula-tions in cloud computing with regard to adaptability, homogeneity or heterogeneity, and pro-cess booking.

A previous paper also reveals that changes in MIPS will effect the response time, and that in-creases in MIPS vs VM will result in a reduction in the reaction time. When the image size of the VM is compared to the VM data transmission, there is not a significant influence discovered on the response time, and it remains constant for which these limits are investigated. In any event, if there is an instance of Cloudlet long length against Host transfer speed, an illustration is observed in which response time increases in a corresponding manner.

By using the IDEA-based EDA technique, a reduction in the reaction time and make span of the various operations was achieved, as was shown in the findings. According to the findings, it is unmistakable that the recommended structure used the project cutoff time as an information barrier in order to enhance results.

Future Scope:

1. A server administrator is able to relocate a running virtual machine or application to multi-ple physical machines without disconnecting the client or the application when using deadline-based job scheduling of VMs.

2. The total allocation time and the downtime are two major performance measures that the customers of a VM service worry about the most, since they are worried about the degree to which the service degrades and the amount of time that the service is entirely unavailable.

3. Cloud computing is a crucial component, and in the context of this line of enquiry, compara-ble other criteria may be compared and contrasted in order to determine their impact on load balancing as part of a future course of action.

REFERENCES

[1]. Pang, S.; Li, W.; He, H.; Shan, Z.; Wang, A.X.; “An EDA-GA Hybrid Algorithm for Multi-Objective Task Scheduling in Cloud Computing”, IEEE, vol: 7, 2019, pp: 146379-146389

[2]. Saleh, H.; Nashaat, H.; Saber, W.; Harb, H.M.; “IPSO Task Scheduling Algorithm for Large Scale Data in Cloud Computing Environment”, IEEE, vol: 7, 2019, pp: 5412-5420

[3]. Sudheer, M.S.; Krishna, M.V.; “Dynamic PSO for Task Scheduling Optimization in Cloud Computing”, International Journal of Recent Technology and Engineering, vol: 8, 2019, pp: 332-338

[4]. Devarasetty, P.; Reddy, S.; “Research of Task Management and Resource Allocation in Cloud Computing”, International Journal of Innovative Technology and Exploring Engi-neering, vol: 8, 2019, pp: 938-941

[5]. Djamegni, C.T.; Kamla, V.C..; Mboula, J.E.N.; “Cost-time trade-off efficient workflow scheduling in cloud”, Simulation Modelling Practice and Theory, 2020, Vol: 103, pp:102-107

[6]. Kołodziej, E.; Wilczynski, A.; “Modelling and simulation of security-aware task scheduling in cloud computing based on Blockchain technology”, Simulation Modelling Practice and Theory, vol: 99, 2020

[7]. Sen, P.; Sarddar, D.; Sinha, S.K.; Pandit, R.; “Web Service Scheduling in Multi-Cloud Envi-ronment”, International Journal of Computer Sciences and Engineering, vol: 7, 2019, pp: 30-38

[8]. Kaur, R.; Kaur, A.; “Efficient job scheduling algorithm based on weighted average priori-ty in cloud computing”, International Journal of Advance Research, Ideas and Innova-tions in Technology, vol: 5, 2019, pp: 354-349

[9]. Zeng, J.; Kang, K.; Zhao, Y.; Fan, X.; Ding, D.; “Q-learning based dynamic task scheduling for energy-efficient cloud computing”, Future Generation Computer Systems, vol: 108, 2020, pp: 361-371

[10]. Chong, C.K.; Mohamad, M.S.; Deris, S.; Shamsir, M.D.; Choon, Y.W.; Chai, L.E.; “Im-proved Differential Evolution Algorithm for Parameter Estimation to Improve the Pro-duction of Biochemical Pathway”, International Journal of Artificial Intelligence and In-teractive Multimedia, Vol: 1, No 5, pp: 22-29

[11]. Zang, L.; Zhou, L.; Salah, A.; “Efficient scientific workflow scheduling for deadline-constrained parallel tasks in cloud computing environments”, Information Sciences, vol: 531, 2020, pp: 31-46

[12]. Sanaj, M.S.; Prathap, P.M.J.; “Nature inspired chaotic squirrel search algorithm (CSSA) for multi objective task scheduling in an IAAS cloud computing atmosphere”, Engineer-ing Science and Technology, an International Journal, 2019

[13]. Refieyan, E.; Khorsand, R.; Ramezanpour, M.; “An adaptive scheduling approach based on integrated best-worst and VIKOR for cloud computing”, Computers & Industrial Engi-neering, Vol: 140, 2020

[14]. Chaudhary, D.; Kumar, B.; “Cost optimized Hybrid Genetic-Gravitational Search Algo-rithm for load scheduling in Cloud Computing”, Applied Soft Computing, Vol. 83, 2019

[15]. Aloboud, E.; Kurdi, H.; “Cuckoo-inspired Job Scheduling Algorithm for Cloud Computing”, Procedia Computer Science, vol: 151, 2019

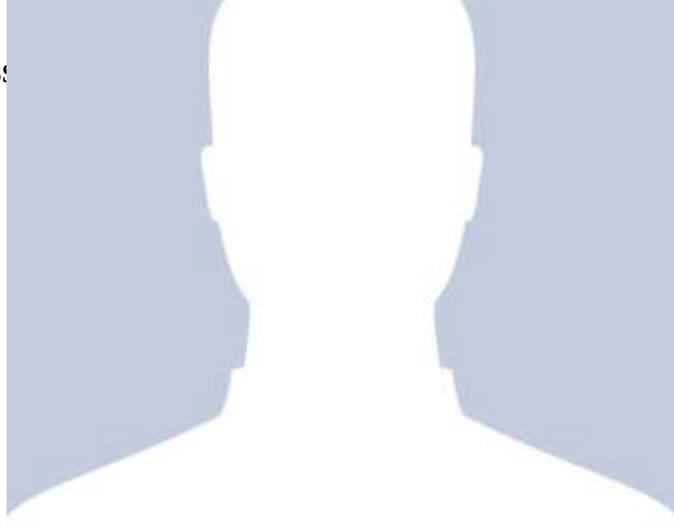
[16]. Sharma, N.; Tyagi, S.; “A Comparative Analysis of Min-Min and Max-Min Algorithms based on the Makespan Parameter”, International Journal of Advanced Research in Computer Science, Vol: 8, No: 3, 2017, pp: 1038-1041

[17]. Miglani, N.; Sharma, G.; “Modified Particle Swarm Optimization based upon Task cate-gorization in Cloud Environment”, International Journal of Engineering and Advanced Technology, Vol: 8, Issue: 4C, 2019, pp: 67-72

[18]. Patel, S.; Bhoi, U.; “Priority Based Job Scheduling Techniques In Cloud Computing: A Sys-tematic Review”, International Journal Of Scientific & Technology Research, Vol: 2, Is-sue: 11, 2013, pp: 147-152

[19]. Moon, Y.J.; Yu, H.; Gil, J.M.; Lim, J.B.; “A slave ants based ant colony optimization algo-rithm for task scheduling in cloud computing environments”, Human-centric Computing and Information Sciences, Vol: 7, number: 1, 2017

[20]. Srichandan, S.; Kumar, T.A.; Bibhudatta, S.; “Task scheduling for cloud computing using multi-objective hybrid bacteriaforaging algorithm”, Future Computing and Informatics Journal, vol: 3, 2018, pp: 210-230



[21]. Nayak, S.C.; Parida, S.; Tripathy, C.; Pattnaik, P.K.; "An enhanced deadline constraint based task scheduling mechanism for cloud environment", Journal of King Saud Univer-sity – Computer and Information Sciences, 2018, pp: 1-13

[22]. Madhukara, E.; Raghunathanb, T.; "Efficient Scheduling Algorithm for Cloud", 2nd Inter-national Symposium on Big Data and Cloud Computing, vol. 50, 2015, pp: 353-356

[23]. Huang, Z.; Chen, Y.; "An Improved Differential Evolution Algorithm Based on Adaptive Parameter", Journal of Control Science and Engineering, vol. 2013, 2013, pp: 1-5

[24]. https://www.intechopen.com/books/scheduling-problems-new-applications-and-trends/types-of-task-scheduling-algorithms-in-cloud-computing-environment