International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 /doi:10.14445/22315381/IJETT-V69I2P225

Volume 69 Issue 2, 178-187, February 2021

A Compact Analytical Survey on Task Scheduling in Cloud Computing Environment

Prashant B. Jawade[1], D Sai Kumar[2], S. Ramachandram[3]
Osmania University

[1]prashantjawade1234@gmail.com, [2]sai.daripalli@gmail.com. [3]schandarm@gmail.com

Abstract — *A computing environment is conveyed by Cloud computing, in which diverse resources are being conveyed via the internet as services to the users or the numerous occupants. In a cloud computing environment, task scheduling is said to be the basic as well as the most significant one. The task scheduling is mainly utilized to designate certain assignments to specific resources at a specific time occasion. Numerous strategies have been proposed to take care of the issues of task scheduling in the cloud environment. Typically, Task scheduling improves the productive use of assets and yields less response time with the goal that the execution of submitted tasks happens inside a potential least time. This paper talks about the investigation of different task scheduling algorithms in a distributed computing condition. This review provides a clear view of different techniques utilized for task scheduling. Further, the security-based task scheduling works are also analyzed. The performance evaluation of different task scheduling techniques is analyzed, and finally, the research gaps and challenges of different task scheduling models.*

Keywords — *Cloud Computing; Task Scheduling Algorithms; Mode of Scheduling; Performance Parameters; Research Gaps and Challenges.*

Nomenclature

Abbreviation	Description
ACO	Ant Colony Optimization
ATS	Aware Task Scheduling
BF	Bacterial Foraging
BT	Bitbrains Task
CDCs	Cloud Data-Centers
CH and CM	cloud heterogeneity and cost model
CPE	Critical Path Extraction Algorithm
CTD	Centralized Task Dispatcher
CTPS	Cloud Task Partitioning Scheduling
C-UP	continuously-updating policy
DAGP	DAG Partitioning Algorithm
DCLCA	Dynamic Clustering League Championship Algorithm
DE	Differential Evolution
DEFT	Dynamic Fault-tolerant Elastic Scheduling Algorithm
DRP & TS	dynamic resource provisioning and task scheduling
ECOS	Efficient Task clustering Based Cost-Effective Aware Scheduling Algorithm
EDA	Estimation Of Distribution Algorithm
EEITS	Energy Efficient Independent Task Scheduler
EPRD	Efficient Priority And Relative Distance
ETC	Expected Time To Complete
ETMCTSA	Energy-Performance Trade-Off Multi-Resource Cloud Task Scheduling Algorithm
FGWO	Fractional Grey-Wolf Optimization
FQVCS	Fuzzy Qualitative Value Calculation System
GA	Genetic Algorithm
GA-CCRA	Genetic Algorithm-Based Customer-Conscious Resource Allocation
GCPP	Google Cloud Platform Pricing
GWOA	Grouping Whale Optimization Algorithm
HAS	Harmony Search Algorithm
HIGA	Hybrid Metaheuristic Scheme Harmony-Inspired Genetic Algorithm
IaaS	Infrastructure As A Service
IRRO	Improved Raven Roosting Optimization Algorithm
IRRO-CSO	ICDSF Scheduling Framework, With The Aid Of The Hybrid
JRGA	Johnson'sRule-Based Genetic Algorithm
MAS	Mobility-Aware Scheduling
MCDM	Multicriteria Decision Making
MFGMTS	Modified Fractional Grey Wolf Optimizer For Multi-Objective Task Scheduling
MobMBAR	Mobility-Aware Heuristic Based Scheduling And Allocation Approach
MPSO	Modified Particle Swarm Optimization
MSA	Moth Search Algorithm
NIST	National Institute Of Standards And Technology
PaaS	Platform As A Service
PSO	Particle Swarm Optimization
QEEC	Q-Learning Based Task Scheduling Framework For Energy-Efficient Cloud Computing
QoS	Quality Of Service
RSS	Received Signal Strength
RTPSO	Ranging Function And Tuning Function-Based PSO
RTPSO-B	RTPSO With Bat Calculation
RTTSMCE	Response Time-Aware Task Scheduling In The Multi-Cloudlet Environment
S & F	store-and-forward
SaaS	Software As A Service
SLA-LB	Service Level Agreement-Based Load Balancing) Algorithm
TBTS	Threshold Based Task Scheduling Algorithm
TPSA-HMCE	task partitioning scheduling algorithms for a heterogeneous multi-cloud environment
UQMM	uncertainty based QoS Min–Min
VM	Virtual Machines
ABC	Artificial Bee Colony Algorithm
CPM	Cost Prediction Matrix
SCAS	Security And Cost Aware Scheduling
SM-PSO	Slow-Movement Particle Swarm Optimization
SABA	Security-Aware and Budget Aware
CSPs	Cloud Service Providers
MAS-CM	Multi-Agent System Based Cloud Monitoring

I. INTRODUCTION

The U.S. NIST describes Cloud-Computing as-" Cloud computing is a system that allows on-demand and extremely dynamic resources that can be conveniently provisioned network access to a shared pool of configurable computing and released involving least management effort or service provider interaction [9] [10]". In cloud computing, Scheduling includes various errands or jobs to be executed with the assets that are available to accomplish superior, least an ideal opportunity for reaction furthermore, the better assignment just as the use of the assets [25] [26]. The VM has utilized the cloud for task distribution, and there are numerous issues identified with a legitimate allotment and use of virtual assets utilizing scheduling [1] [15]. A better scheduling algorithm is the one with reduced execution cost and Execution time.

The scheduling algorithms are essential for performing a task effectively, and it also aids inadequately dealing with the resource utilization with the accentuation on load balance. A role is being played by the scheduling algorithm in the case of assigning tasks with various criteria onto the VM [9] [12]. The Cloud task scheduling is said to be an NP-hard problem. In the case of organizations, the clients present their business to the cloud scheduler in order to allocate tasks. A decent scheduling calculation consistently improves the CPU use, turnaround time, and total throughput.

The major contribution of this research works is:

- This paper conducts a compact study of task scheduling techniques and the related measurements reasonable for distributed computing situations. It examines the different issues identified with task scheduling strategies and the impediments to survival.
- The literature review will be sorted out dependent on alternate points of view like techniques, mode of scheduling, parameters of scheduling, and Performance of scheduling. Along with this, the security-based task scheduling models are also analyzed in this survey.
- Furthermore, research gaps and challenges alongside future course identified with task planning for distributed computing is characterized toward the finish of the exploration work.

The rest of the paper is organized as: Section II depicts the fascinating literature works undergone in task scheduling in the cloud environment. Section III tells about the analytical review on task scheduling in the cloud computing environment. Section IV discusses the research gaps and challenges in the existing researchers. Finally, the survey paper is concluded with a strong conclusion in Section V.

II. LITERATURE REVIEW

A. Related works

In 2020, Dinga *et al.* [1] had proposed a QEEC. The proposed model had encapsulated two major phases. In the primary stage, the authors have deployed a CTD with the intention of executing the "M/M/S queueing model," which was good in allocating the tasks to the cloud server on the basis of the client demands. In the subsequent phase, the "Q-learning based scheduler" with respect to every server had organized all the solicitations by "task laxity and task life" time. Further, they have utilized a C-UP in order to appoint tasks to VM and have applied the incentives to compensate the assignments that can limit task reaction time and augment CPU utilization of every server.

In 2019, Lavanya *et al.* [2] had developed two new scheduling algorithms referred to as TBTS and SLA-LB. In the case of TBTS, the Task scheduling takes place in a cluster and has generated the threshold data on the basis of the ETC matrix. On the other hand, the SLA-LB algorithm had performed dynamic task scheduling with regards to the user's requirements like the deadline and financial plan.

In 2020, Abdelmoneema *et al.* [3] had developed a productive IoT framework referred to as MAS and have allocation protocols for medicinal services. This approach had bolsters the portability of the patients through a versatile RSS-based handoff component. This approach traduced the MobMBAR to permit the dynamic appropriation of medicinal services tasks among computational hubs, whether cloud gadgets or haze gadgets. The major intention behind this approach was to diminish the total schedule time during the positioning and reallocation stages through the usage of the task features like "critical level and the maximum response time."

In 2020, Garg and Sharma [4] have developed an EEITS with the aid of the supervised NN to decrease "makespan, utilization of energy, execution overhead and the count of dynamic racks. The genetic algorithm was utilized to produce an enormous dataset "(~18 million training instances)" and with this generated dataset, they have trained the neural network.

In 2019, Rjoub *et al.* [5] had developed "BigTrustScheduling, a trust-aware scheduling solution to override the issue in cloud computing." The proposed model encapsulates three phases: trust level calculation of virtual machine, level assurance on the basis of the priority of tasks, and trust-aware scheduling. The authors have conducted the investigation of the proposed model with the collected real-world datasets gathered from the GCPP and BT and resource requirements.

In 2020, Zhang *et al.* [6] have introduced an EPRD algorithm with the target of limiting the length of task scheduling for priority obliged work process applications without abusing the end-to-end cutoff time limitation. The proposed model comprises two procedures. Initially, they have built a task priority queue and have further mapped a VM for a task as per its relative separation. The proposed strategy was said to have adequately improved the utilization of VM as well as the performance of scheduling.

In 2019, Wilczynski *et al.* [7] had constructed a novel proof–of–schedule" consensus algorithm (rather than proof–of–work") and have enhanced the endorsement of the produced schedules with the aid of Stackelberg games. The authors have experimentally simulated the created model and have approved it by utilizing the new unique cloud test simulator. The analyses had revealed that the applied methodology had essentially improved the proficiency of arranged schedules.

In 2020, Tuli *et al*. [8] have centered around the construction of the existing DRP & TS algorithms in hybrid cloud environments for providing better QoS in information concentrated applications in a common record task condition. They have exhibited the productivity of the proposed algorithm by sending them on "Microsoft Azure utilizing Aneka," a stage for creating versatile applications on the Cloud.

In 2019, Elaziz *et al*. [9] had developed an elective technique for task scheduling on various VMs in the cloud environment with limited makespan. This approach was based on the "improvement of the MSA" by utilizing the principle of the DE model. The proposed MSDE algorithm was validated, and the assessment resultant have exhibited that the proposed algorithm outflanked different calculations as indicated by the exhibition measures (makespan).

In 2019, Sharma and Garg [10] had introduced a novel HIGA to handle the issue of energy-effective task planning on a current cloud server farm. This HIGA was constructed by the amalgamation of the GA and exploitation capability of HAS. The essential destinations in this work were to diminish the "makespan and energy." Further, the auxiliary goals of this work were to lessen the energy devoured by the "resources other than processing assets" (resources) and lessen execution overhead connected with the scheduler. The outcome has shown that the proposed approach had bought about higher energy savings as well as lower execution overhead and makespan.

In 2019, Mansouri *et al*. [11] have developed FMPSO, a "hybrid task scheduling algorithm" that was based on the Fuzzy framework and MPSO method. The target of this research was to upgrade the throughput of the cloud and its load balancing potential. The input to the proposed FMPSO model was "task length, CPU speed, RAM size" and total execution time. Further, the execution time, as well as the utilization of the resources, was lessened by deploying fuzzy systems.

In 2018, Mao *et al*. [12] had introduced two good quality algorithms for heterogeneous cloud-based task scheduling, and these methods were referred to as "a time-aware algorithm and an energy-aware algorithm." In addition, they have developed ETMCTSA to control the performance of the energy and to manage the flexibility of the cloud system. The exploratory outcomes showed that with a pre-determined probability parameter α, the ETMCTSA was found to be more time-efficient and energy-efficient.

In 2018, Sobhanayak *et al*. [13] had developed hybrid biologically-inspired heuristic algorithms for investigating the task Scheduling mechanism in the cloud environment. The proposed model was the integration of "GAs and the BF algorithms." The fundamental commitments of this article were twofold. The makespan was reduced with the scheduling algorithm and, secondly, lessens the utilization of energy (both monetary and natural viewpoints). Exploratory outcomes had revealed that the presentation of the proposed algorithm overrides those of different algorithms with respect to dependability and arrangement decent variety.

In 2018, Wu *et al*. [14] have proposed a delicate error-aware energy-efficient task scheduling approach in the cloud data centers of DVFS for workflow applications. Under unwavering quality and finish time limitations mentioned by inhabitants, the proposed methodology was able to produce energy-proficient task schedules for work processes by assigning undertakings to proper VM with explicit working frequencies.

In 2018, Yan *et al*. [15] have acquainted the vulnerability with the "task runtime estimation model" and have provided a flaw lenient task assignment tool that deliberately utilizes two fault-tolerant task scheduling models. In addition, an innovative DEFT was proposed for real-time task scheduling with performance volatility consideration in the cloud. The outcomes from the broad tests on the remaining task at hand of the Google trace logs showed that the proposed DEFT had ensured adaptation to internal failure while accomplishing high asset usage.

In 2018, Nayak and Tripathy [16] proposed a new approach for the utilization of the MCDM with the VIKOR strategy. The component had explored the best perfect errand among the comparable assignments and positions them for scheduling. The presentation was assessed by investigating the number of scheduled tasks, resource usage, and dismissal tasks, which was better in contrast with the current algorithm.

In 2019, PANG *et al*. [17] had developed a hybrid scheduling algorithm referred to as EDA-GA by merging the GA and the EDA. The major intention behind this approach was to lessen the task completion time and to enhance its load balancing potential. They have produced a specific size of plausible arrangements by utilizing the EDA sampling method and the probability model. Then, the inquiry scope of arrangements was extended by utilizing the crossover and mutation operations of GA. At last, the ideal scheduling procedure for delegating tasks to virtual machines was figured out. The exploratory outcomes show that the EDA-GA mixture calculation had adequately decreased the errand fruition time and had improved the capacity of load balancing.

In 2014, Xiong *et al*. [18] have dissected the issues of "task scheduling for CDCs" and have built up a scientific model of the scheduling of two-phase tasks. In addition, they have introduced the JRGA by joining the GA with Johnson's rule. The major intention behind this amalgamation was to optimize the makespan of the tasks in the cloud by considers the qualities of multiprocessor tasks.

In 2015, ZUO *et al*. [19] had developed a "multi-objective optimization method" in distributed computing environment for efficient task scheduling. Initially, they have developed a resource cost model in order to characterize the interest of tasks on assets with more subtleties. Further, on the basis of the resource cost model, they have developed a multi-objective optimization scheduling method with the target of accomplishing multi-objective advancement of both execution and cost.

In 2018, LU *et al.* [20] have defined an optimization problem by considering the S &F anycast plans to boost the average time benefit from serving information situated tasks in a data centers of cloud framework. Further, the Lyapunov optimization techniques were leveraged with the aim of formulating a proficient task scheduling algorithm (GlobalAny). They have lessened the data-transfer latency by proposing a data-transfer acceleration scheme.

In 2016, Panda *et al.* [21] have developed a UQMM algorithm in heterogeneous multi-cloud conditions. This approach takes into consideration the QoS dependent on vulnerability parameters. They have performed broad simulations on the proposed approach by utilizing the synthetic as well as benchmark datasets in terms of different measurements.

In 2013, Adabi *et al.* [22] had introduced a new bi-level advanced reservation strategy on the basis of the first performing scheduling worldwide and afterward directing schedule locally. In addition, they have proposed CPE on the basis of the resource requirement as well as the specification of DAGs. A novel dynamic score-based approach was developed with regard to the DAGP in order to deal with separate sub-work processes. Further, they have deployed a new FQVCS on the basis of the critical paths to evaluate the cloud environment.

In 2017, Moon *et al*. [23] had introduced a novel "cloud task scheduling algorithm" on the basis of the ACO to allocate tasks to VM in the distributed conditions. Further, with the slave ants of ACO, they have deployed the diversification and reinforcement procedures in order to enhance the task scheduling performance.

In 2014, Netjinda *et al.* [24] have concentrated on enhancing the expense of buying "infrastructure-as-a-service cloud" abilities to accomplish logical work process execution inside the particular cutoff times. The proposed framework considers the quantity of bought samples, an option of purchasing, buying choices, and task scheduling as parameters in a streamlining procedure. The optimal solution was explored by augmenting the variable neighborhood search technique with the PSO.

In 2016, Abdulhamid *et al.* [25] have proffered a DCLCA scheduling technique in order to address the execution of the tasks in the cloud. This was based on the fault tolerance awareness, and hence the resultant had exhibited a reduction in the untimely failure of the tasks. The resultant of the proposed model had exhibited remarkable fault reduction while contrasted to the existing approaches in terms of failure rate.

In 2017, Panda *et al*. [26] have projected three TPSA-HMCE, and these models were referred to as "CTPS, cloud min–min task partitioning scheduling and cloud max-min task partitioning scheduling." Among this algorithm, the "min–min task partitioning scheduling" and "cloud max-min task partitioning scheduling" are said to be the offline scheduling algorithm, whereas the cloud task partitioning scheduling was an online scheduling algorithm.

In 2015, Pandra *et al.* [27] had developed an allocation-ATS algorithm with three diverse phases: "matching, allocating and scheduling" for task scheduling in "heterogeneous multi-cloud systems." The target behind this research was to minimize the makespan by means of mapping the necessities of the customers to the virtual machine. The authors had undergone rigorous experiments on the multi-cloud with the aid of the synthetic as well as benchmark datasets. The experimental outcomes have uncovered the superiority of the presented work in terms of usage of average cloud and makespan.

In 2019, Dong *et al.* [28] have developed an ECOS for cloud-based task scheduling. The major objective behind this research was to alleviate the consumed cost with no comprises in its task completion deadline. Initially, the authors have developed a task-clustering problem with respect to the constraints like multi-type work processes, CH and CM. Further, the ECOS was devised with the aid of two key advances: (a) vertical clustering: to lessen the transferring time by joining the sequential tasks within the workflow, (b) greedy allocation as well as horizontal clustering: to alleviate the cost of the task scheduling with no compromise.

In 2017, Jena *et al*. [29] have developed GA-CCRA and "Task Scheduling in the environment of multi-cloud." The major intention behind this research work was to delineate errands to VMs within minimum makespan time as well as most extreme consumer loyalty. In the initial phase of this research work, the resource allocation was done via the genetic algorithm, and in the subsequent phase, the scheduling was done on the basis of the shortest task-based priority. Thorough analyses were done with the aid of the synthetic data and were contrasted over the current scheduling models. Consequences of experimental outcomes had delineated that the proposed model was better than the current ones according to concerning measurements.

In 2018, Sreenu & Malempati [30] had proffered MFGMTS, a "multi-objective optimization algorithm for task scheduling in the cloud computing environment." The penalty cost function, as well as the epsilon-constraint, were utilized to compute the targets, "execution time, execution cost, correspondence time, correspondence cost, utilization of energy and resources." They have persuaded the algorithm by FGWO with an alteration in the position update, where an extra term is consolidated utilizing the mix of alpha and beta arrangements.

In 2019, Rama Subba Reddy and Sasikala [31] developed RTTSMCE for task scheduling in the cloud environment. This model was constructed to tolerate two major issues: (a) response time-based cloudlet server selection and (b) load balancing algorithms in cloudlets for task scheduling in the cloud server. The proposed algorithm had exhibited a superior execution contrasted with customary load balancing models.

In 2020, Sanaj *et al*. [32] had constructed an efficient task-scheduling algorithm in the cloud for task allocation to the VM on the basis of accomplishing the least use of the resource, least preparing time, high proficiency, and greatest profit. They have deployed the whale optimization algorithm, a new met heuristic technique, to solve the issue of task-scheduling. The resultant had shown that the proposed WOA calculation had extraordinarily expanded the proficiency and have accomplished the most extreme benefit for the private cloud.

In 2017, Valarmathi *et al*. [33] had developed an enhanced PSO algorithm in the cloud environment to improve the task scheduling ability of VMs. The authors have introduced an RTPSO in view of the information region for fathoming the dormancy weight task issue in the existing PSO calculation for scheduling the tasks. In addition, they have consolidated the RTPSO-B with the aim of enhancing the optimization. Further, the task scheduling was simulated in the cloud environment with Cloudsim.

In 2018, Torabi *et al*. [34] had developed an IRRO-CSO meta-heuristic methodology, which was dependent on the CSO algorithm and IRRO. The CSO was utilized for its productivity in fulfilling the harmony between the nearby and IRRO calculation solved the premature convergence issue in greater pursuit spaces.

In 2019, Hemasian-Etefagh [35] had introduced an improved variant of the Whale optimization algorithm referred to as GWOA. The early convergence problem was overridden initially, and the optimal solution was found, making a balance between the global as well as local search. Further, at a high workload, the authors have deployed the grouping Whale optimization algorithm within the cloud computing scheduler in order to lessen the normal execution time, increase the throughput and limit the response time.

In 2019, Thanka *et al.* [36] had proposed an improved efficiency— ABC in the cloud environment in order to achieve the target of security and QoS aware scheduling. The optimal virtual machine was assigned with the tasks based on the required security level of the user as well as service policy qualities. Further, in the virtual machine, the authors have maintained the hive table in each data center for reducing the "makespan, cost, security risk, task migration."

In 2020, Sujana *et al.* [37] had developed a novel model based on the CPM, which lessens the cost calculation based on the child's child task. Further, in the cloud, on the basis of the security level, the required VM was selected by the authors with a fuzzy-based decision model. In the workflow, they have amalgamated the CPM with the fuzzy model and hence named it as SCPS algorithm. The resultants had exhibited a reduction in the makespan.

In 2016, Li *et al.* [38] have projected a SCAS algorithm in the cloud environment for scientific workflow in the case of heterogeneous tasks. The authors have reduced the workflow execution cost by means of deploying the PSO. The effectiveness of the proposed model was exhibited in terms of "security, constraints of risk rate and a deadline."

In 2020, Zhang *et al.* [39] have developed two SM-PSO algorithms in order to solve the issues related to the resultant NP-hard. They have proposed a "position-based mapping scheme" in order to produce solutions with higher quality. Further, to generate more quality-aware solutions with higher security, a novel particle updating strategy was introduced in addition. The effectiveness of the proposed model had exhibited the improvement of the proposed work over the others.

In 2017, Shishido *et al.* [40] have made an attempt to comprehend the betterment in workflow scheduling, The performance among the two metaheuristic scheduling techniques, namely PSO and GA. The performance evaluation was made on these approaches using a security and cost-aware workflow scheduling algorithm. In the end, they have found that GA was much superior to PSO in terms of cost-effectiveness, security, and response time.

In 2014, Zeng *et al.* [41]have introduced an immoveable dataset concept on the basis of the security and cost in the cloud environment. In addition, they have proposed a SABA workflow scheduling strategy in order to serve the consumers with lower makespan as well as higher security tasks. The available CSPs were given the tasks on the basis of the economical distribution.

In 2017, Grzonka *et al.* [42] had proffered a novel MAS-CM model s in large-scale service-oriented environments in order to achieve the objective of higher security levels even under the scheduling and execution processes. They have monitored as well as bar the injection of the unauthorized task into the VM by optimizing the scheduling process and resource usage maximization.

III. ANALYTICAL REVIEW ON TASK SCHEDULING IN CLOUD COMPUTING ENVIRONMENT

The chronological literature review portrays each work as per when it was distributed, beginning with the most punctual accessible data. Considering this, the current survey paper on task scheduling in the cloud environment is assessed for each indispensable time span (i.e., year). Since task scheduling is a newer topic, most of the works are being developed in recent years, and henceforth this paper had also discussed the works developed in the recent past (2013 to 2020). A 3% contribution is provided by the collected papers in the year 2013. In the years 2014, 2015, and 2016, a contribution of 8%, 5%, and 8% is provided by the research works discussed in the literature. Further, the task scheduling papers collected in the year 2017, 2018, 2019 and 2020 is 14%, 19%, 26%, and 17%, respectively. All these research works have together contributed to the current survey to be more comprehensive.

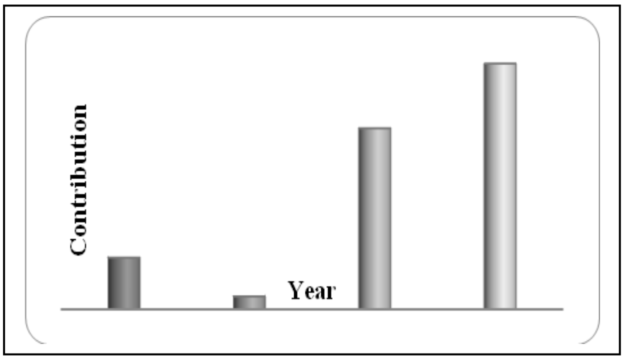


Fig. 1.Bar chart representing chronological review on task scheduling models in a cloud computing environment

A. Algorithmic Analysis

Fig. 2 represents the diverse approaches and techniques utilized in different scheduling models. The QEEC approach [1], TBTS and SLA-LB[2] , MobMBAR[3] are some of the techniques. The machine learning models like ANN [4], Fuzzy framework [11] [12], [22] are followed. The other techniques utilized are: BigTrustScheduling[5], efficient priority and relative distance [6], proof–of–schedule' consensus algorithm[7], Aneka[8], soft error-aware energy- efficient task scheduling [14] , *ECOS* [28] , DEFT[15], VIKOR method[16], GlobalAny[20] , UQMM[21], DCLCA[25], CTPS[26], ATS[27], RTTSMCE[31]. In addition, the Optimization based approaches implied are Hybrid MSA and DE[9], HIGA[10],MPSO[11],BF[13], EDA-GA [17], JRGA[18],ACO[19] [23],PSO[24] [38] [39] [40] ,GA[29],[40], MFGMTS[30], WOA[32], RTPSO[33], IRRO-CSO [34] and GWOA[35] , ABC [36], SCPS algorithm in [37], SABA [41] and MAS-CM [42].

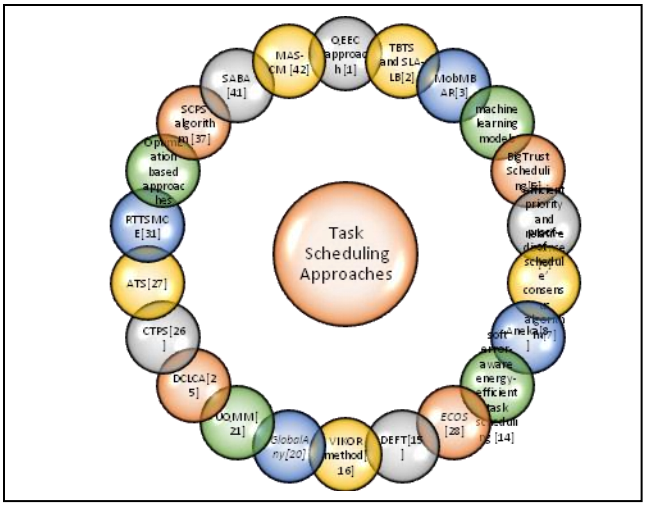


Fig. 2.Task scheduling approaches and techniques

B. Mode of Scheduling

The task scheduling takes place in VM, either statistically or dynamically. In the case of static scheduling, the tasks are allocated to the scheduler at the same instant of time, and they are independent of the availability as well as resources. On the other hand, the tasks arrive at the scheduler at a varying period of time, and it is dependent on the machine state of the VM. From the collected research papers under this survey, the dynamic scheduling has been paid little higher interest by the researchers while compared to the statistic scheduling. The dynamic scheduling is implied in [1], [2] ,[4], [5] ,[6] ,[8] ,[9] ,[10] ,[11], [14] ,[15], [20] , [21] , [22] ,[25] ,[28] ,[33] ,[34], [36], [37] ,[38], [40] ,[41] ,[42] and [35], respectively. The rest of the papers are dependent on the static scheduling approach. Fig. 3 represents the model of scheduling.

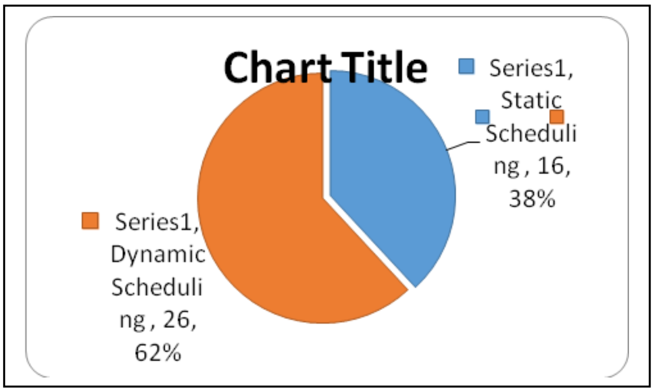
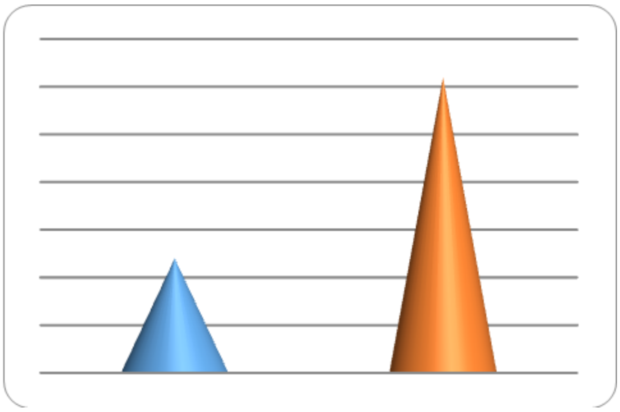


Fig. 3.Mode of scheduling: Static and Dynamic

C. Objective-based Scheduling and multi-cloudet

The scheduling is accomplished on the basis of Single objective of multi-objective on the basis of the user requirement. The multi-objective based scheduling is followed in [2], [12], [13], [16], [17], [18], [19], [21], [22], [26], [40] and [30]. The overall contribution of multi-objective task scheduling is 28%, while the rest, 73% of the collected papers, have undergone single objective-based scheduling. Fig. 4 represents the objective-based scheduling.



Further, multi-cloud-based task scheduling is a fresh topic, and hence only a few counts of researchers have focussed on it. The [7], [8], [21],[26],[26],[27], [29], [31] and [32] have accomplished the task scheduling on the multi-cloud environment.

D. Performance Evaluation: Based on Parameters

The performance of the collected model is reviewed in terms of diverse parameters like QoS, reduced energy consumption, meets the deadline, improved load balancing, reduced response time, reduced makespan, reduced time complexity, reduced resource cost, fault-tolerance, and improved resource utilization. In addition, the security constraint is considered in [7], [32],[36] [37] [38] [39] ,[40] ,[41] and [42]. Table I exhibits the Parameter based performance evaluation.

TABLE I. ANALYSIS OF PERFORMANCE MEASURES

[Citation]	QoS	Energy	Deadline	Load balancing	Response time	Makesp an	Resource cost	Fault tolerance	Resource utilization	Time complexity
[1].		✓			✓				✓	
[2].			✓	✓		✓	✓		✓	✓
[3].		✓	✓						✓	
[4].		✓				✓				
[5].	✓					✓		✓		
[6].		✓		✓	✓					✓
[7].		✓				✓	✓			
[8].	✓		✓		✓		✓			
[9].						✓	✓		✓	✓
[10].		✓	✓			✓	✓		✓	
[11].		✓		✓	✓				✓	✓
[12].		✓			✓				✓	✓
[13].		✓			✓			✓		
[14].		✓	✓		✓			✓		✓
[15].							✓	✓	✓	
[16].			✓		✓		✓			✓
[17].				✓	✓		✓			
[18].									✓	
[19].			✓			✓	✓		✓	
[20].					✓		✓			✓
[21].	✓		✓				✓	✓		
[22].	✓				✓		✓		✓	✓
[23].						✓	✓			
[24].			✓			✓	✓			✓
[25].					✓	✓		✓		✓
[26].					✓	✓			✓	✓
[27].						✓			✓	✓
[28].			✓	✓	✓					✓
[29].				✓		✓	✓			
[30].		✓					✓		✓	✓
[31].				✓	✓		✓			
[32].							✓			
[33].						✓	✓		✓	
[34].					✓	✓				✓
[35].					✓					✓
[36].	✓		✓	✓	✓	✓	✓	✓	✓	✓
[37].			✓			✓	✓	✓	✓	✓
[38].	✓	✓	✓			✓				✓
[39].		✓	✓		✓	✓	✓		✓	✓
[40].	✓	✓	✓		✓	✓	✓		✓	✓
[41].						✓	✓		✓	✓
[42].	✓	✓		✓		✓	✓		✓	✓

E. Maximum Attained Value

The performance of each of the works is shown in Table II. In [2], the makespan achieves the lowest value, and its makespan of Proposed TBTS= 1.3597E+04. In [4], when the CPU utilization is 100%, the power consumed for performing the tasks has been reduced, and it is 117 W (Watts). In [6], the Received signal strength threshold varied between -70 dbm to -90 dbm. The average processing capability in [15] is 800 MIPS. In case of [24], the Average cloud resource utilization = 0.9048. In [36], the risk rate ranges from 0.1 to 1.

TABLE II. ANALYSIS ON PERFORMANCE MEASURES

[Citation]	Parameters and their best values	[Citation]	Parameters and their best values
[1]	Request Arrival Rate of M/M/S in Homogeneous cloud=46.02 (num per sec.).In Heterogeneous=26.64 (num per sec.)	[18]	CPU computational ability=1860 mips, Bandwidth =100M/s,
[2]	Makes pan of Proposed TBTS= 1.3597E+04 Makes pan of Proposed SLA-LB= 2.1033E+04 Cloudutilization of	[19]	Average throughput=1.13(*10^6) and computation time= 57.1sec

	Proposed TBTS= 0.402 Cloud Utilization of Proposed SLA-LB= 0.4199		
[3]	Makespan reduced upto 88% and Energy consumption reduced upto 92%	[20]	Overall makespan=74, Average cloud execution time= 71.33sec, Uncertainty time=12.1s
[4]	CPU utilization (%)= 100% Power consumption =117 W (Watts)	[21]	Total count of submitted tasks=15,000
[5]	VM=50, number of tasks to 7,884	[22]	The number of cloud tasks is 200, makespan=15s
[6]	Received signal strength threshold =-70 dbm to -90 dbm	[23]	Total cost= 100.73$
[7]	Minimum number of operations= 1000000	[24]	Lower makespan= 57.8, 53.6, 24.3 and 13.4 % in the first scenario
[8]	Speedup =1.67 Response time reduced by 40.12%	[29]	Minimum execution time= 0.186243, Minimum communication time= 0.174782, Minimum execution cost= 0.016045, Minimum communication cost= 0.087023, Minimum energy consumption= 0.012259, And minimum resource utilization= 0.564528,
[9]	Ɛ = 0.8	[30]	Response time=1.965
[10]	Execution overhead reduced up to up to 21–39%.	[31]	Maximum average time= 16013.52 ms Profit= 2212.8$
[11	Total execution time= 733 seconds, Makespan= 172, Degree of imbalance=23	[32]	Processing capacity of VM= 150 MIPS
[12	ETMCTSA with α=0.3 costs 13.16% reduced ETMCTSA with α=0.6 is 10.5% reduced in energy consumption	[33]	Response time is reduced upto 20.22%
[13]	The mean coverage ratios= 0.52256,Standard deviation= 0.2641	[34]	Execution time=500ms Response time= 1.55e+04
[14]	Energy savings reduced upto reliability requirement reliability requirement= 0.996	[35]	CPU computational ability=1860 mips, Bandwidth =100M/s,
[15]	The average processing capability= 800 MIPS	[36]	risk rate ranges from 0.1 to 1
[16]	Resource utilization reduced upto 28.7%	[37]	Computational capability= 1860MHZ probability of security> 0.85
[17]	Fittness value increased upto 5.4%	[38]	Overhead= 13.50 (kB/ms), security levels=1
[18]	Makespan (_s) for 100 tasks= 2577	[39]	computation times= 431.44 ms
[25]	Average cloud resource utilization = 0.9048	[40]	Overhead (kB/ms)=13.50, Security level=1.0
[26]	Makespan= 0.8961 Average cloud utilization= 0.93	[41]	Computation Ratio (CCR)=0.1
[27]	Average memory usage=7.54 Average running time (seconds)=635.7	[42]	Makespan of 5 task batches per hour (in [sec.]) Active= 2097, Passive= 2426 and Reference=3365
[28]	Waiting time for resource= varies from 0.01 to 0.05		

IV. RESEARCH GAPS AND CHALLENGES

The ideal joint asset allotment conspires proper in any event for numerous asset types as the designation task is straightforward, and it is done at the same instance to individual request. Essentially, a reasonable joint ass*et al*lotment strategy can adjust the total scope of key resources doled out for each service at each block time. However, the primary issue of these plans is that it doesn't bolster on task prioritization and allotment of resources as per the request. The planning system that expects to accomplish better load adjusting of the resources of VM is significant in sorting out the assets (resources).

In the scheduling system, the dynamic migration, as well as ideal load balancing, is accomplished by choosing the least effective solution, and it is much time multifaceted in nature. However, in the virtual machine, this framework conveys productive load adjustments and improves its asset utility, the time multifaceted nature is high.

The tasks can be relocated without degrading the nearby activity execution level by performing the priority-based scheduling running in queues and multilevel feedback queue scheduling. An efficient task allocation is accomplished with clustering instead of between every single accessible hub by the Priority-based scheduling scheme in order to accomplish the processing power. But, in the cloud condition, the Priority-based scheduling system framework isn't persuading regarding client prerequisites.

In the literature section, the cost-based scheduling algorithm-based works are explained, as it bolsters better access assets in the cloud. The computation, as well as communication proportion, is upgraded in the Cost-based scheduling by a blend of the client tasks. However, in the cloud condition, the algorithmic improvement doesn't focus on the autonomous scheduling of tasks. Likewise, there are requirements for an extra spotlight on the structure of appropriated redirection and administration scaling procedures.

The issues regarding security are said to be prominently increasing, and this is of much concern in today's academic and industries. On the basis of the defects in the cloud infrastructure due to attacks, the security of the data becomes lower, and so diverse intrusion detection approaches with optimization algorithms can be deployed for enhancing the security of the sensitive data as well as operational authority. The cross-virtual machine-side channel attacks can be defended by paying more attention to cloud migration in the VM. The illegal utilization of the rights in public clouds of CSP can be effectively limited by building up a security defense policy. Along these lines, the future proposed works might consider the abovementioned limits, contingent upon which recommendations rise as proper procedures to satisfy the downsides with commitment.

V. CONCLUSION AND FUTURE SCOPE

This paper had provided a compact, clear explanation of various task scheduling algorithms in cloud computing. This survey had also given a vivid explanation of the different approaches utilized for task scheduling in diverse works. This paper had provided a clear view of the model of scheduling and the parameters utilized in each of the approaches.

In conclusion

- This survey paper had reviewed 42 research papers corresponding to various task scheduling approaches in the cloud computing environment and had exhibited the benefits of each of the research.
- Initially, this research had reviewed the different task scheduling models. In addition, these models are grouped under: a model of scheduling (static or dynamic), objective-based scheduling (multi-objective or single objective).
- The parameter-based performance evaluation was undergone for each of the collected papers. In addition, this analysis had portrayed the best values of each performance metric concerning image compression.
- In the end, a clear explanation was provided about the research issues in various task scheduling models in cloud computing, and this can be significant for future researches on task scheduling techniques.

The future direction of this research can focus on “quickly achieve target security and personalized privacy protection.”

- The purpose of the users in terms of private information can be protected with secure search technologies by means of providing real-time warnings of behavior privacy.
- In big data, the knowledge extraction technology, as well as Information fusion, can be integrated into the cloud for enhancing the users' search needs. The key-based sensitive data hiding approaches can be introduced, and the based optimal key can be selected with self-adaptive optimization algorithms or hybrid optimization algorithms.
- The lightweight cryptographic algorithm can be introduced to ensure security as well as confidentiality for real-time big data processing.

References

[1] Ding Dinga, Xiaocong Fan, Yihuan Zhaoa, Kaixuan Kanga, Qian Yina, Jing Zenga, Q-Learning Based Dynamic Task Scheduling for Energy-Efficient Cloud Computing, Future Generation Computer Systems, (2020).

[2] M.Lavanya, B. Shanthi, S.Saravanan, multi-objective task scheduling algorithm based on sla and processing time suitable for the cloud environment, Computer Communications, (2019).

[3] Randa M. Abdelmoneema, Abderrahim Benslimane, Eman Shaabana, Mobility-aware task scheduling in cloud-Fog IoT-based healthcare architectures, Computer Networks, 179(2020).

[4] Sharma M, Garg R, An artificial neural network-based approach for energy-efficient task scheduling in cloud data centers, Sustainable Computing: Informatics and Systems,(2020).

[5] Gaith Rjoub, Jamal Bentahar, Omar Abdel Wahab, BigTrustScheduling: Trust-aware big data task scheduling approach in cloud computing environments, Future Generation Computer Systems, November (2019).

[6] Longxin Zhang, Liqian Zhou, Ahmad Salahb, Efficient scientific workflow scheduling for deadline-constrained parallel tasks in cloud computing environments, Information Sciences, 531(2020) 31-46.

[7] Andrzej Wilczynski, Joanna Ko lodziej, Modelling and Simulation of Security-aware Task Scheduling in Cloud Computing Based on Blockchain Technology, Simulation Modelling Practice and Theory, (2019).

[8] Shreshth Tuli, Rajinder Sandhu, Rajkumar Buyya, Shared data-aware dynamic resource provisioning and task scheduling for data-intensive applications on hybrid clouds using Aneka, Future Generation Computer Systems, 106(2020) 596-606.

[9] Mohamed Abd Elaziz, Shengwu Xiong, K.P.N. Jayasena, Lin Li, Task scheduling in cloud computing based on hybrid moth search algorithm and differential evolution, Knowledge-Based Systems, (2019).

[10] Mohan Sharma and Ritu Garg, HIGA: Harmony-inspired genetic algorithm for rack-aware energy-efficient task scheduling in cloud data centers, Engineering Science and Technology, an International Journal,(2019).

[11] Najme Mansouri, Behnam Mohammad Hasani Zade, and Mohammad Masoud Javidi, Hybrid task scheduling strategy for cloud computing by modified particle swarm optimization and fuzzy theory, Computers & Industrial Engineering, 130(2019) 597-633.

[12] Li Mao, Yin Li, Gaofeng Peng, Xiyao Xu, Weiwei Lin, A Multi-Resource Task Scheduling Algorithm for Energy-Performance Trade-offs in Green Clouds, Sustainable Computing: Informatics and Systems,(2018).

[13] Srichandan Sobhanayak, Ashok Kumar Turuk, Bibhudatta Sahoo, Task scheduling for cloud computing using multi-objective hybrid bacteria foraging algorithm, Future Computing and Informatics Journal, March (2018).

[14] Tingming Wu, Haifeng Gu, Junlong Zhou, Tongquan Wei, Xiao Liu, Mingsong Chen, Soft Error-Aware Energy-EfÞcient Task Scheduling for Workßow Applications in DVFS-Enabled Cloud, Journal of Systems Architecture, (2018).

[15] Hui Yan, Xiaomin Zhu, Huangke Chen, Hui Guo, Wen Zhou, Weidong Bao, DEFT: Dynamic Fault-Tolerant Elastic Scheduling for Tasks with Uncertain Runtime in Cloud, Information Sciences, (2018).

[16] Suvendu Chandan Nayak, Chitaranjan Tripathy, Deadline based task scheduling using multi-criteria decision-making in a cloud environment, Ain Shams Engineering Journal, (2017).

[17] Shanchen Pang, Wenhao Li1, Hua He, Zhiguang Shan, And Xun Wang, An EDA-GA Hybrid Algorithm for Multi-Objective Task Scheduling in Cloud Computing, IEEE Access, (2019).

[18] Yonghua Xiong, Suzhen Huang, Min Wu, Jinhua She and Keyuan Jiang, A Johnson'sRule-Based Genetic Algorithm for Two-Stage-Task Scheduling Problem in Data-Centers of Cloud Computing, JOURNAL OF LATEX CLASS FILES, 13(9)(2014).

[19] Liyun Zuo1, Lei Shu, Shoubin Dong, Chunsheng Zhu, And Takahiro Hara, A Multi-Objective Optimization Scheduling Method Based on the Ant Colony Algorithm in Cloud Computing, IEEE Access, (2015).

[20] Wei Lu, Ping Lu, Quanying Sun, Shui Yu, And Zuqing Zhu, Profit-Aware Distributed Online Scheduling for Data-Oriented Tasks in Cloud Datacenters, IEEE Access, (2018).

[21] Sanjaya K. Panda, Prasanta K. Jana, Uncertainty-Based QoS Min–Min Algorithm for Heterogeneous Multi-cloud Environment, Arab J Sci Eng, (2016).

[22] Sahar Adabi, Ali Movaghar, Amir Masoud Rahmani, Bi-level fuzzy-based advanced reservation of Cloud workflow applications on distributed Grid resources, J Supercomput, (2013).

[23] Yong Ju Moon, HeonChang Yu, Joon-Min Gil, and JongBeom Lim, A slave ants based ant colony optimization algorithm for task scheduling in cloud computing environments, Human- centric computing a dn information sciences, (2017).

[24] Nuttapong Netjinda, Booncharoen Sirinaovaku, Tiranee Achalaku, Cost optimal scheduling in IaaS for dependent workload with particle swarm optimization, J Supercomput, (2014).

[25] Shafi'i Muhammad Abdulhamid, Muhammad Shafie Abd Latiff, Syed Hamid Hussain Madni, Mohammed Abdullahi, Fault tolerance aware scheduling technique for cloud computing environment using a dynamic clustering algorithm, Neural Comput & Application, 29(2018) 279-293.

[26] Sanjaya Kumar Panda, Sohan Kumar Pande, Satyabrata Das, Task Partitioning Scheduling Algorithms for Heterogeneous Multi-Cloud Environment, Arab J Sci Eng, (2017).

[27] Sanjaya K. Pandaa, Indrajeet Gupta and Prasanta K. Jana,: Allocation-Aware Task Scheduling for Heterogeneous Multi-Cloud Systems, Procedia Computer Science, 50(2015) 176 – 184.

[28] Minggang Dong, Lili Fan, Chao Jing, ECOS: An efficient task-clustering based cost-effective aware scheduling algorithm for scientific workflows execution on heterogeneous cloud systems, The Journal of Systems and Software, 158(2019).

[29] Tamanna Jena and J. R. Mohanty, GA-Based Customer-Conscious Resource Allocation and Task Scheduling in Multi-cloud Computing, Arab J Sci Eng, (2017).

[30] Karnam Sreenu & Sreelatha Malempati, MFGMTS: Epsilon Constraint-Based Modified Fractional Grey Wolf Optimizer for Multi-Objective Task Scheduling in Cloud Computing, IETE JOURNAL OF RESEARCH, (2017).

[31] Somula Ramasubbareddy & R. Sasikala, RTTSMCE: a response time aware task scheduling in the multi-cloudlet environment, International Journal of Computers and Applications, (2019).

[32] Sanaj MS, Joe Prathap P M & Valanto Alappatt,Profit maximization based task scheduling in hybrid clouds using whale optimization technique, Information Security Journal: A Global Perspective,(2020).

[33] R. Valarmathi, T. Sheela, Ranging and tuning based particle swarm optimization with bat algorithm for task scheduling in cloud computing, Cluster Computing, (2017).

[34] Shadi Torabi, Faramarz Safi-Esfahani, A dynamic task scheduling framework based on chicken swarm and improved raven roosting optimization methods in cloud computing, J Supercomput, (2018).

[35] Farinaz Hemasian-Etefagh, Faramarz Safi-Esfahani, Dynamic scheduling applying new population grouping of whales meta-heuristic in cloud computing," The Journal of Supercomputing, (2019).

[36] M. Roshni Thanka, P. Uma Maheswari & E. Bijolin Edwin, An improved efficient: Artificial Bee Colony algorithm for security and QoS aware scheduling in the cloud computing environment, Cluster Computing, 22(2019) 10905–10913.

[37] Angela Jennifa Sujana, J, Revathi, T., Joshua Rajanayagam, S., Fuzzy-based Security-Driven Optimistic Scheduling of Scientific Workflows in Cloud Computing, IETE Journal of Research, 66(2)(2020).

[38] Zhongjin Li, Jidong Ge, Hongji Yang, Liguo Huang, Haiyang Hu, Hao Hu, Bin Luo, A security and cost-aware scheduling algorithm for heterogeneous tasks of scientific workflow in clouds, Future Generation Computer Systems, 65(2016) 140-152.

[39] Yi Zhang, Yu Liu, Junlong Zhou, Jin Sun, Keqin Li, Slow-movement particle swarm optimization algorithms for scheduling security-critical tasks in resource-limited mobile edge computing, Future Generation Computer Systems, 112(2020) 148-161.

[40] Henrique Yoshikazu Shishido, Júlio Cezar Estrella, Claudio Fabiano Motta Toledo, and Marcio Silva Arantes, Genetic-based algorithms applied to a workflow scheduling algorithm with security and deadline constraints in clouds, Computers and Electrical Engineering, (2017).

[41] Langfang Zeng, Bharadwaj Veeravalli, Xiaorong Li, SABA: A security-aware and budget-aware workflow scheduling strategy in clouds, J. Parallel Distrib. Comput., (2014).

[42] Daniel Grzonka, Agnieszka Jakóbik, Joanna Kołodziej, Sabri Pllana, Using a multi-agent system and artificial intelligence for monitoring and improving the cloud performance and security, Future Generation Computer Systems, (2017).

[43] Waleed Abd Elkhalik, Ahmad Salah, Ibrahim El-Henawy A Survey on Cloud Computing Scheduling Algorithms, International Journal of Engineering Trends and Technology (IJETT), V60(1) 65-70 (2018).