Int J Syst Assur Eng Manag (October 2022) 13(5):2132–2150
https://doi.org/10.1007/s13198-022-01685-3

REVIEW PAPERS

Valuable survey on scheduling algorithms in the cloud with various publications

Nidhi Bansal[1] · Ajay Kumar Singh[2]

Received: 11 January 2021 / Revised: 5 January 2022 / Accepted: 18 May 2022 / Published online: 11 June 2022

Abstract Computational and comprehensive applications with administrative computing are emerging to be the gateway to development. Cloud computing technology came as a gift in this context to leverage resources as needed. A large amount of intensive or embedded applications are in the market to drive any proposal designed by the authors. Scheduling is a serious issue in cloud computing, which needs to be available with the resources needed for easy computing. To achieve user satisfaction and system efficiency by providing essential services to users, the cloud data center maintains the management of various resources. The paper analyzes several research articles published by various publishing houses along with their respective factors as well as their impact value. Recent trends also explained various approaches to stakeholders related to the issues and challenges that come with task scheduling. Not only one or a few specified factors, scientists should also focus on individual dynamic factors, as unique and ambiguous requests are facing problems during scheduling to perform advanced factors such as security, trust, etc. Ultimately, this gives the field of future research to do more research with more factors.

Nidhi Bansal and Ajay Kumar Singh: Formerly Uttar Pradesh Technical University.

✉ Nidhi Bansal
nidhi18jul@gmail.com

Ajay Kumar Singh
ajay41274@gmail.com

[1] Computer Science and Engineering Department, PhD Student of AKTU (Formerly Uttar Pradesh Technical University), Lucknow, India

[2] Computer Science and Engineering Department, KIET Group of Institutions, Delhi-NCR, Ghaziabad, Affiliated With AKTU (Formerly Uttar Pradesh Technical University), Lucknow, India

Keywords Cloud computing · Task scheduling · Optimization techniques · Publication house

1 Introduction

Thanks to the everywhere cloud, everyone is eager to use cloud computing technology and wants to create a result to be compared with other traditional and natural technologies. From cloud technology, there is a synergy between different technologies that are running with many algorithms. People now came to know about cloud technology in how data is processed in the cloud with various scheduling algorithms. Multiprocessing is also done by interconnecting various visualized resources under task scheduling. A series of different configuration systems collaborate with different types of resources to achieve critical computing. It provides a range of useful tools and advanced services to make a scientific and practical world for embedded technology. Cloud computing manages the essential things in a very pleasant way. As in the world map, everything is visible to everyone; similarly, computational capabilities can also be tracked by providing open doors for developers to create cloud map visibility. Cloud computing is like a pool where every technology is deeply defined with its advantages and disadvantages. Pools include service migration, replication, security, storage space, practical tools for development, commodity management, and more. The specialty of one fruit here is that no one is dependent on the other in any way. All users are unique; so it don't create dependencies on anyone. The term development serves as an innovation, which must be followed in all aspects. Assurance of development and advancement using cloud computing is on the completion line. Cloud computing technology services have many names, all defining their key points with an in-depth analysis of practical, as well as theoretical aspects. Executed results can be deployed with different architectures of the cloud (Salamy 2019). Versions of the cloud enable the user to run their work with their own spaces and responsibilities.

Application based projects are running in the market with high name and fame. Cloud computing is also designed for accomplishment of this platform. It works through an Internet connection to access data from the server to user processes. Cloud computing works in two phases, one phase for the user end and the other for the system end. The user end allows access to the stored data and the system end provides data storage facilities with its complete privacy. The cloud server provides assurance about the services provided with the help of all connected components, such as the repository, the server, and the main one CSP (Cloud Service Provider). Only the CSP can communicate and provide cloud services to the customer with their needs and fulfillment of commitments. Services are being provided by terms and conditions designed with a particular authorized channel. The channel maintains governance about the functioning of the system and hierarchy. Several security gates have also been installed on the servers to reduce routine violations. Multinationals companies are recognizing the benefits of cloud-based architecture and build their own systems to allow developers to use utility support, to take screenshots of the golden digital world. The organization works as a CSP (cloud service provider) to store a layout for technology workers. The working model for cloud computing is depicted by Fig. 1a, b. Since a large number of requests are generated day to day to receive services from the cloud, many customers are soon ready in a queue to get their chance. The entire activity is focused by strong connectivity, as the CSP receives the request via send/receive and then it merges with the server system to complete the process.

The CSP is divided with a description of the scheduling process. In which, the task is mapped with the applicable scheduling and criteria are set for obtaining the optimized result by considering the specific factor. All CSPs differ in their services, so they can design a chart according to their internal imperatives. For availing the services, User first searches the CSPs according to their needs and then sends a request to an established CSP. The service provider informs the customer about the terms and conditions of obtaining the commitment. Communication then takes a legal place in the system to provide the benefit of essential services to the customer. Then system set the requests in

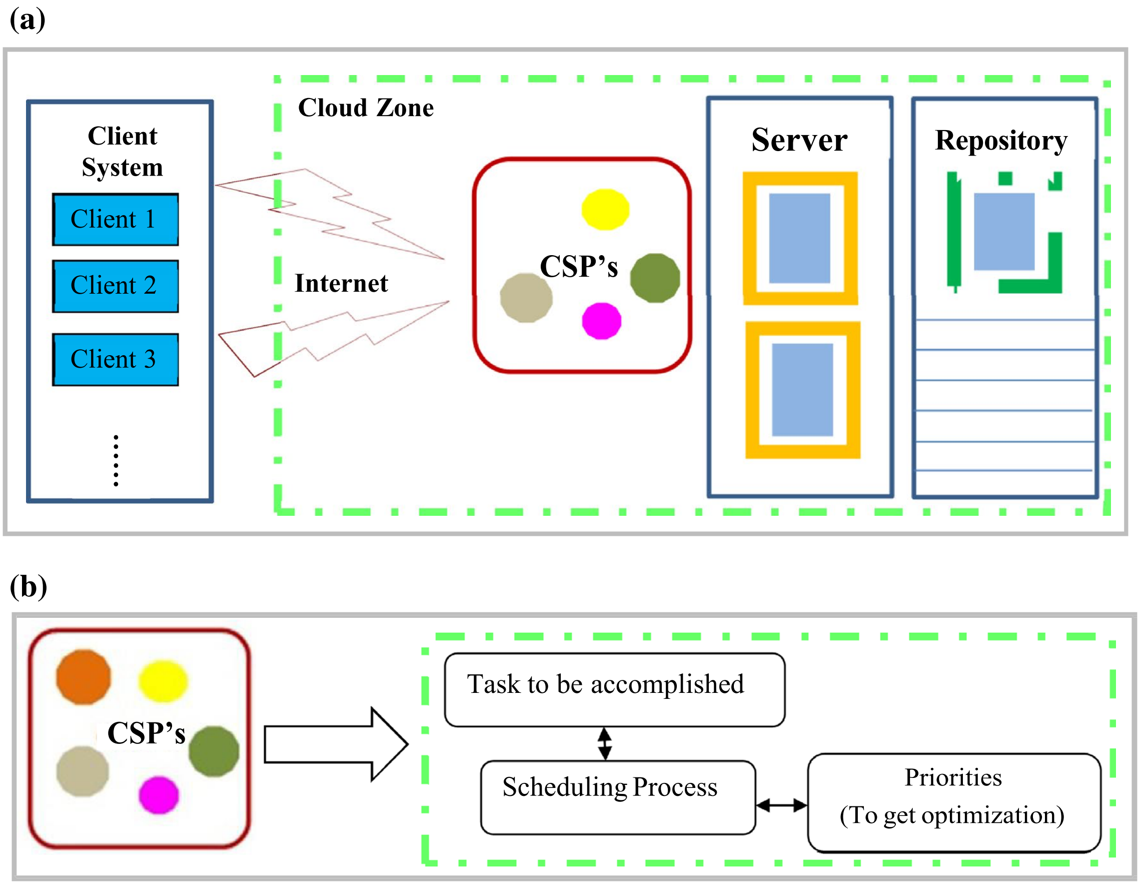

Fig. 1 **a**. Working model for cloud computing. **b** Working model for cloud computing

queue with the preferences of the requested policies and related factors, now the request is available and mapped to the required configuration machine, and then it produces the optimized results.

Types of Clouds Private Clouds, not for everyone, it provides services to the preferred user over private confidential networks rather than the general public. Public Cloud is open to all; it provides services by the third party over the universal internet. It is accessible for any type of user who wants to use the provided services (Anjuli Garg and Krishna 2014). With the passage of time, the provider can change the rules and services according to the user's response. To increase the market and introduce new services, the technology needs more refinement. Hybrid cloud requires an amalgamation of the two to complete a single project. Public cloud is very helpful for dealing with public morality and private is very safe and secure for people working in the same organization. Community Cloud, a combination of efforts, has given freedom to various organizations that work similarly to development by sharing tools and facilities to the respective organization. Details about the cloud type are shown by Fig. 2a, b.

The characteristics of various clouds are listed below. Public and private clouds are intended as a personal concern, while community and hybrid clouds are based on mutual concern meetings.

Interest in real life is an important point to achieve targeted work. In the same way, scientists are eager to introduce the concept of cloud; Cloud is neither a product nor a process. The cloud is just the concept, with whatever researchers have analyzed what is happening in the developing environment. After getting an in-depth analysis, the theories found that everyone wants easily accessibility of the services in one place instead of going here and there. Then cloud computing technology comes and starts counting growth only in the cloud as the cloud gives freedom to use different types of services on the same server. Services as follows IAAS (infrastructure as a service), PAAS (platform as a service), SAAS (software as a service). Further services were introduced from these three core services, they are: STAAS (storage as a service), APIAAS (API as a service), DAAS (data as a service), TAAS (trust as a service), TEAAS (testing as a service), AAAS (application as a service), SECAAS (security as a service).

Now everyone is accustomed to cloud computing technology with its useful activities. In order to process this paper will have to introduce a phase of cloud computing i.e. scheduling. Scheduling is also working as a service for users

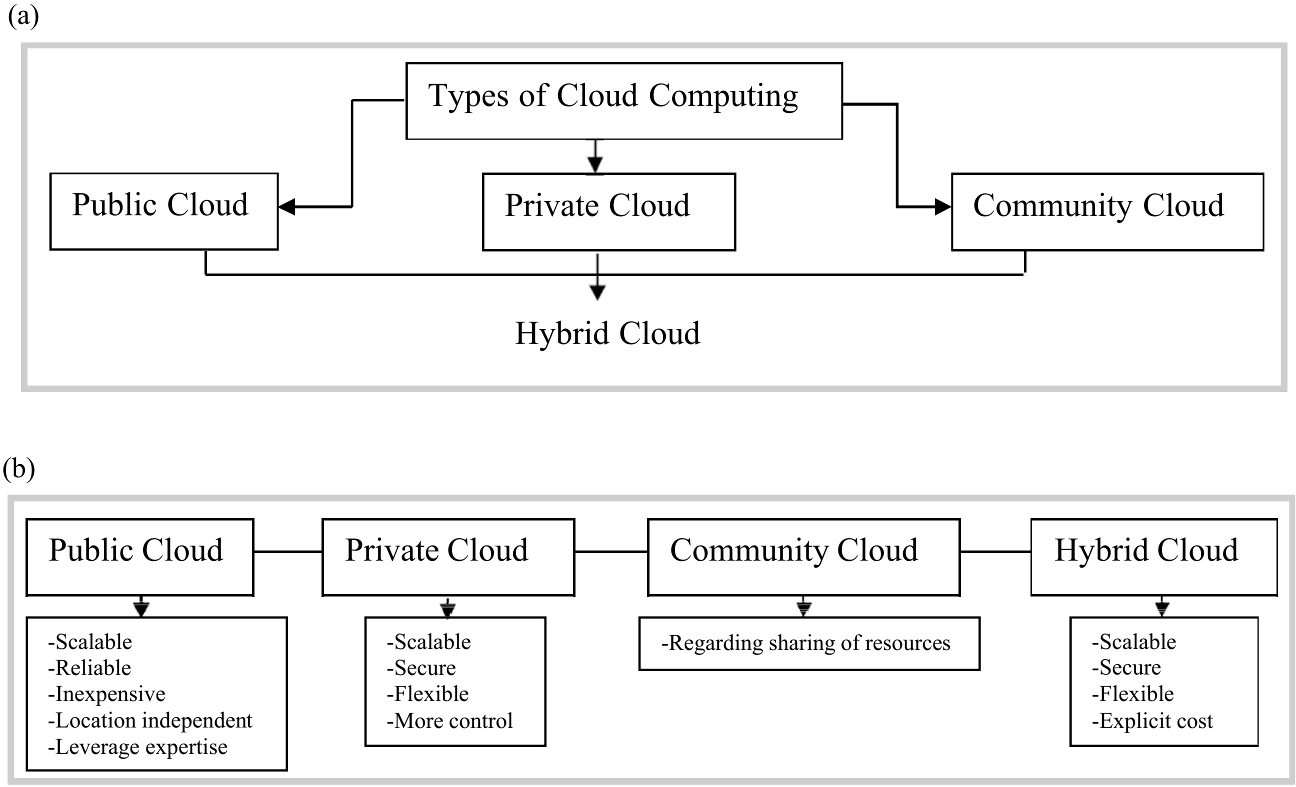

Fig. 2 **a**. Types of cloud computing, **b**. Types of cloud computing

by obtaining more coordinated working model to process the task very well. How to schedule a work from the cloud server, through which other assignments should not be interrupted and secure the good results of the scheduling work. Thousands of scheduling algorithms have been developed to get better results than in the past. And it produces big chaos in today's market, how to select the best scheduling algorithm to accomplish a task in a fruitful way (Bansal 2018; Bansal et al. 2016; Bansal and Dutta 2014). This paper is expanding further by writing literature surveys. The process of this paper is as follows in Fig. 3:

2 Scheduling objective and methodology

Scheduling starts with the mapping of the resources to the machine. If resources are defines as R = {R1, R2, R3…Rn} and machines are defines as M = {M1, M2, M3…Mn} then mapping function can be defined as

$$MpF_{1 \to n} : R \to T \in M$$

where T defines as several task T = {T1, T2, T3…Tn} and MpF ranges from 1 < = n. Task allocated on the prescribed machine with required resource gives a complete mapping of the function to generate fruitful result. The scheduling process is depicted in Fig. 4:

Factors explanation is to be considered in various designed scheduling, makes the scheduling successfully. Factors are in the main role for any designed scheduling, because of its experimented value, scheduling can be goes for judgement in terms of reliability and get decided that proposed scheduling is good to use up to calculated value. Factors may be profitable for user as well as for service provider. According to the user and service provider perspective, factors are here with clear explanation for getting more acceptances.

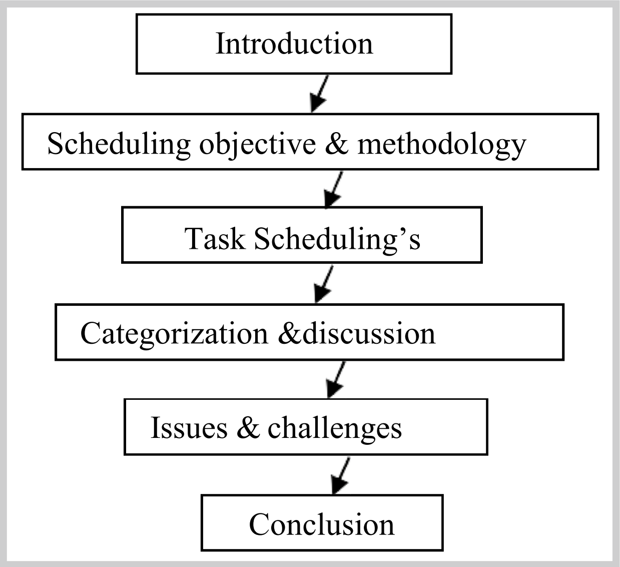
Introduction
Scheduling objective & methodology
Task Scheduling's
Categorization &discussion
Issues & challenges
Conclusion

Fig. 3 Flow of the paper

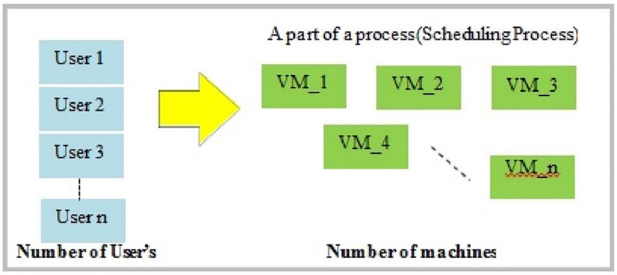

Fig. 4 Scheduling processing

Throughput Successful rate of the transmission is during the way of communication defined as a throughput. It includes slender latency and various rules for a specific schedule can perform in an efficient way. The analysed percentage value of the factor is 09% in today's scheduling development era.

Makespan It is defines as the difference in between the start and end time of a queue of tasks. Most of the survey papers have set the target to get the minimum makespan, i.e. 81%. Due to reduce this factor, many others factor are also getting improved value. Such as cost, resource utilization, schedule length. Many factors are dependent to each other. Due to this good feature, more factors get optimized in a single designed scheduling.

Power consumption energy consumption is indirectly proportional to the utilization of the resources. Machines usage will be good then power consumption will be low. Sometimes machines will not get proper use due to heavy requirement of resources, at that time systems produces high energy and the cost will also get increase automatically. The power factor consumed will help to analyze how energy consumption is considered in developing optimized results i.e. 14% value obtained.

Local optima In the computational world, the best solution is provided by local optima through a problem occurring over a small area. For the global optimum value, the local optima perform well with the cream layer under different possible generated layers or outcomes. This factor needs more calculation as it has yielded 05%.

Cost and Time it contains many factors such as transmission, allocation, execution, completion, management etc. Both are directly promoted to each other, if the time will be reduced, the cost will be reduced automatically the value also. Spend less time, expenses will be less. The user will gain access to the services for a limited and required time period, and will pay the provider based on its revaluation. Mathematics is a very strong keyword from where people can find formulas for their specific work. The modification can be applied to statistical instructions through the user to achieve the desired output. The esteemed values obtained through analysis for time and cost are 25% in this paper.

Availability of resources The available resources satisfy the user that he has all the materials to achieve success and complete all the tasks in the design project. The Reservation Factor may be considered by researchers to ensure this factor makes it easier to schedule task. 01% gained by this factor in current survey.

Schedule length Due to high demand and limited number of resources, the length of the schedule to execute the task and to meet the target increases. Resources are being generated to reduce the length value. Paper analysed the value for this factor is 04%.

Resource utilization Use can be done within the range, never crossing the limit value. Each machine must achieve a uniform or average load during processing for full utilization of resources. To get higher mileage, the fare amount should be clear to the user. 55% value is reserved for utilization.

Bandwidth Through single or multiple channels, the amount of data migrated is considered in the definition of bandwidth. High bandwidth is considered a good result in the computation of many data on a channel. Analysed value for bandwidth factor in this paper is 04%.

QoS Many values of factors can be achieved through quality service. A single entity defines many more factors or values. Whatever services the service provider provides to the user, the evaluation result falls under the quality factor. 07% papers are looking for QoS.

Fault Tolerance Within the failure, the system generates the output very clearly, defining as the property of fault tolerance. It is an intelligence factor when a machine calculates a task properly in a non-active way. It should be kept in mind by the authors in later papers that the factor analysis is found to be very low i.e. 01%

Security This is a very important and non-avoidable factor, as resources are varying in nature due to varying demand by the user. Data security becomes a very crucial goal after multi-tent features due to virtual century. Very limited or no research paper is trying to overcome this factor. The fear of data leakage and vulnerabilities is playing a main role in virtualization and distribution of facilities. To achieve an efficient and secure virtual processing, the security factor must be highlighted with the highest priorities.

Percentage values are calculated based on the consideration of a particular factor in current research. The use of the factors is noted from the articles and the values for each of them are summarized.

3 Task scheduling

The research paper published from various prestigious publications presented here. Scheduling is a term that defines the perfect match between two entities. These two entities can be anything; it can be a resource or a task. Various types of scheduling methods are in nature to get optimized results. Optimization can be considered in terms of a good result or decision. Task scheduling processes the task efficiently and effectively based on their characteristics. Many more concepts or terminologies are in use under task scheduling, these characteristics are explored here with published articles.

A novel hybrid cost-based prioritization concept has been used to design new task scheduling to obtain better values of factors such as throughput, makespan, and average waiting time (Satish 2019). A fuzzy based modified particle swarm algorithm has been optimized which is working perfectly in the paper and showing improvements such as makespan, correction ratio, imbalance degree, efficiency and total execution time (Mansouri 2019). A valuable technique has been devised to move the load on different machines for better performance. In this paper, the author works on the earliest time limit (EDF) and increases the value of power consumption (Salamy 2019). Inequality of work makes things indisputable. To overcome this, the combination may move towards a catastrophic situation. By applying the Hungarian algorithm, the author achieves the required results (KumarPanda et al. 2018). As scheduling demand increases, new challenges take their place to compete with the other. To provide assurance, within the time limit, the author proposed an assurance by proposing a backfilling mechanism (ChandanNayak 2018). Author refines the factors throughput, local optima by using hybrid load balancing algorithm with the combination of Teaching-Learning-Based Optimization (TLBO) and Grey Wolves Optimization algorithms (GWO) (Mousavi 2018). The energy consumption factor is modified with the use of the equal speed method for each task in the queue and is ready for execution (Li 2018). To save the energy consumed by the system, it is best to use an approximation algorithm for the first fit decreasing concept (Tian et al. 2018). Amalgamation of two different factors may provide an efficient result. By combining genetic algorithms (GAs) and the bacterial foraging (BF), algorithm gets reduction in makespan and energy consumption (Srichandan 2018). The most crucial factor i.e. cost has been considered in this paper. Author reduces the cost efficiency by using the new resource provisioning and scheduling strategy that designed specifically for Workflow as a service (WaaS) environments (Rodriguez and Buyya 2018). To maximize the CPU real time, a novel scheduling has been proposed, it analyses the resources usage over the time passing by their executions in order to achieve optimized performance

(Sotiriadis and Bessis 2018). A unique technique is applied in this paper i.e. an alternative direction method of multipliers, which makes room for users to obtain greater availability of resources (Sebastio 2017). Ant colony optimization has been used to provide better in terms of scheduling enhancements in robustness and reduce machine inequality. To improve resource utilization, proper use of resources should be made during the selection rule proposed by the author (Wei et al. 2012). Get amendment in various factors like, energy efficiency and execution time by implementing Clonal Selection Algorithm (CSA) (Jena 2017). Most common factors makespan and resource utilization are getting more refinement by proposing a novel load balancing technique (Mohit Kumar and Sharma 2017). To reduce the length of the schedule for each task, a conversion is implemented as a budget constraint for an application (Weihong et al. 2017). Makespan reduces its value by using the method of heterogeneous earliest finish time in the proposed article (KalkaDubey and Sharma 2017). The author eliminates the time value by completing logs that take previous scheduling information and evaluate them as needed (Ellendula Madhukara and Thirumalaisamy Ragunathan 2015). Lower value found in proposed paper for mentioned factors makespan, execution time, round trip time, and transmission cost by adopting load balancing modified particle swarm optimization (LBMPSO) (Awad et al. 2015). The author improves makespan by accepting unbeatable approach ie zone based honeybees. This approach was effectively done by modification made in the real honey bees nature life cycle (Doreen Hephzibah Miriam et al. 2015). Proper use of resources is a very important factor that should be included in the scenario for any research. The credit-based scheduling concept was proposed in a related paper under the supervision of the corresponding author (Antony Thomas et al. 2015). Makespan and the cloud usage factor are envisaged by the author by proving allocation-aware task scheduling method (Sanjaya 2015a). By using multipurpose nested particle swarm optimization, power consumption and processing time are reduced (Jena 2015). Marginal time completed by applying fuzzy scheduling on referenced paper (Zadeh and Hashemi 2013). The loading factor is considered by the author to reduce execution time and waiting time using the nature of the honeybee (Dhinesh Babu and Venkata Krishna 2013). Quality (QoS), the most popular factor of service, is enhanced by performing tasks according to the highest priority. Another advantage of this algorithm is that it also provides minimum completion time (Xiaonian et al. 2013). Over-use over-processing To achieve this tag, two online dynamic resource allocation algorithms have been implemented for IaaS cloud systems with pre-emption applied to functions (JiayinLi et al. 2012). The curious factor comes here i.e. throughput and user satisfaction. These are achieved by presenting fuzzy logic in various arguments (Fahmy 2010). To achieve increased energy efficiency, adaptive power-aware virtual machine connectors (APA-VMPs) are recommended (Jeyarani et al. 2011). By introducing a new method, namely an online potential timeout, this approach has been designed and implemented using CloudSim Toolkit, which improves various factors of task scheduling based on the increased width of the road to drive heavy traffic smoothly (Aida 2018). Computation cost and makespan and many more QoS factors are minimized by announcing parent preference-based mechanisms (Muhammad Shahzad Arif et al. 2019). The author works on various factors such as: makespan, CPU time, load balancing, stability, and planning speed in this paper using the firefly algorithm (Kashikolaei et al. 2019). The proposed paper, using a load balancing mechanism based on probability distribution, yielded useful results in resource utilization (Panda and Jana 2019). To facilitate energy efficiency and makespan, energy-efficient work scheduling algorithms are recommended (Sanjaya 2019a). In order to elevate the makespan, heuristics approaches have been prescribed for implementation (Nayak and Padhi 2018). Several factors have been appropriated in this paper using the Modified Analytic Hierarchy Process (MAHP) with bandwidth aware divisible scheduling (BATS) and BAR optimization. These apply with the longest expected processing time pre-emption (LEPT) and divide-and-conquer methods (Gawali and Shinde 2018). Developed cuckoo search algorithm to get more refinement in resource usage (Agarwal and Srivastava 2018). The author minimizes makespan and maximizes throughput by polishing the common method with social group optimization and short-job-first scheduling (Phani Praveen et al. 2018). To achieve convalesce in makespan, execution time based sufferage algorithm facilitates more optimization (Krishnaveni and Sinthu 2018a). Several enhancements have been made to the possible algorithm. Makespan, resource utilization and allocation costs are achieved by considering and increasing function-based particle swam optimization with the BAT algorithm (Valarmathi and Sheela 2017). The author summarizes the cost of bandwidth, memory, and storage under a budget constraint schedule by evaluating aggregation into groups of tasks (Alworafi et al. 2017). Genetic algorithm-based customer-consciousness resource allocation and task scheduling in multi-cloud computing are able to reduce makespan and maximize customer satisfaction (Tamanna Jena and Mohanty 2018). Authors accelerate the makespan, cloud utilization, gain and penalty cost of services using service level agreement based scheduling techniques (Sanjaya 2017). The proposed paper shows improvements in makespan and cloud usage by working on allocation-aware task scheduling (Sanjaya 2019b). The author survives long trails whose pheromones are mainly misguided by ants. Thepaper uses reinforcement learning techniques to perform well all the way. (Moon et al. 2017). Flower pollination algorithms are a very useful method to achieve growth in makespan and cloud resource usage (Gupta et al. 2017). The author improves makespan and cloud usage by using distribution and near-radix scaling (Sanjaya 2018). It was very effective to find optimizations with the grouping algorithm by obtaining short performance times for all tasks (Parthasarathy 2017). In the proposed paper, management cost and makespan decreases by using genetic algorithm (Ajeena Beegom and Rajasree 2015). To reduce energy consumption, the proposed scheduling was very useful based on the processing time of the operation and the use of VMs (Panda and Jana 2016). Makespan and cloud resource utilization have been improved by using several methods: MCC, MEMAX and CMMN (Sanjaya 2015b). The author is able to reduce the total execution time and cost by using genetic algorithm and amplification of fuzzy theory (Javanmardi et al. 2014). Using the AVL tree system in the proposed paper improves real-time average system load (Chiu et al. 2014). The author improves resource utilization by using load balancing sensitive genetic algorithm with min-min and max–min method (Zhan and Zhang 2014). Ant colony adaptation proves to be very efficient to reduce the makespan legally (Wang 2013). The author improves the time to complete a task and resource utilization by applying flexible scheduling in real time constraints (Ma et al. 2012). User provision is considered an important factor in the care of service changes. To accomplish this, two-level load balancing task scheduling proves beneficial for resource utilization (Fang et al. 2010). Incompleteness is detected by new challenging scheduling. Therefore fuzzy logic based methods are able to reduce the average waiting time and turnaround time by implementing tasks based on priorities (Shatha 2010). To obtain information about the right service provider, the author relied on the use of the Honey Life scheduling concept to find the trust factor in terms of assurance of quality service (Firdhous et al. 2011). The author improves makespan and resource usage by implementing comprehensive simulations on benchmark data with innovative ideas (Jyoti Gupta et al. 2016). To provide higher resource utilization, genetic gives better results in today's scenario, so the author takes into account two factors. These are: resource utilization and closing costs. These factors have to be optimized using the double-fitness algorithm (Yin et al. 2018). The author improves execution time and fault tolerance by showing separation between different frameworks, such as Amazon or more (Soualhia et al. 2018). The ant colony method is of great use to optimize the results. The author also used it to reduce makespan by implementing the proposed task scheduling algorithm (Gupta and Garg 2017). The author uses energy-aware dynamic task scheduling to save energy for computing many more tasks (Li et al. 2017). Everyone wants the cheapest and best tag to be received from any service. Using a genetic algorithm, the paper minimizes several factors such as performance cost, performance time, and schedule time (Gupta and Tewari 2017). The published paper evaluates the cost of execution using the Bayes classification and shows their optimal results (Zhang and Zhou 2018). In this paper improvements have been made using priority measures for meta tasks, and based on the Min-Min algorithm obtains better makespan, and resource utilization (George Amalarethinam and Kavitha 2017). Dominant Resource Priority and Maximum Utilization Allocation were launched to improve a novel scheduling concept and achieve minimum makespan (Zhang et al. 2016). This paper presents that applying ranking-based scheduling algorithms improves performance in time and resource utilization (Ettikyala 2016). The authors reduce power consumption by using scheduling based on "redundant shutdown, turn on demand" based on potential matching with low energy resource allocation algorithm (Xiaolong et al. 2016). By introducing an optimized algorithm, the author improves resource usage and makespan (Mittal and Katal 2016). Using space-sharing mechanisms in allocated memory, the paper gets refined results by reducing the length and time limit of work (Chauhan and Jaglan 2016). The author improves the use of various factors such as throughput, execution time, and allocation cost and resource usage by using Honeybe's life cycle (Kaur and Kaur 2016). Proposed paper improves makespan and resource usage by using a generalized multi- purpose minimum-minimum maximum-minimum rating (Gajera et al. 2016). Published paper improves work benefit, work fine, throughput, provider benefits and user loss using space sharing scheduling (Himani and Sidhu 2015). The use of genetic algorithm in published articles improves makespan and resource usage (Alaka Ananth 2015). The author developed a smoothing concept and improved the average cloud usage and makespan (Panda et al. 2014). To get a novel solution for better makespan and resource usage, the min min and max min algorithm has been applied (Panda and Jana 2015). The author writes an algorithm to increase resource usage by using a swarm based algorithm (Rana et al. 2014). In the proposed article, the author uses a natural concept i.e. Honey bees which reduce the allocation cost and increases makespan (Anjuli Garg and Krishna 2014). Selected article reduces energy consumption by using energy- saving task consolidation (Panda and Jana 2014). By using cost-conscious cloud workload performance, resource utilization increases. The result of the paper shows that the complete use of machines is very important today because there are many compilations in today's real problems (Thanasias et al. 2014). Most authors are using natural behaviour as an adaptation to achieve success in their implementation. The whole world is dependent on nature. Natural things can never harm anyone. So this letter also uses the Honeybees behaviour to reduce the considerable factor i.e. makespan (Bitam 2012). The referenced article obtains a minimum makespan using the max–min (Devipriya and Ramesh 2013). Author improves processing usage and makespan using Ant Colony Optimization (Li et al. 2011). Fuzzy logic enables more than one option in front of the author to get an optimum result. Therefore, using this argument, many researchers are getting more variation in optimal results through which they can choose the best result (Omara and Zohier 2010). The author improves processing usage and execution time by implementing the work based on its priority (Zhenxia 2008). Makespan, total cost and use of the cloud, has been refined using intimate efforts on some synthetic and benchmark data sets in the proposed article (Panda and Jana 2015). A heterogeneous approach has been used to design this paper. It makes schedules for the task over time. Using this, the author gives good results like schedule length and cost (Topcuoglu and Hariri 1999). A heuristic approach has been used to implement the proposed idea. Author reduces execution time by using it (El-Rewini et al. 1995). The algorithm inspired by the nature used to solve global optimization problems. The author took advantage of this and achieved a reduction in the execution time of the scheduling algorithm (Aujla 2015). By implementing shortest job first, author reduced waiting time and turnaround time (Krishnaveni and Sinthu 2018b). At task benefit and time limit, the author reduces various factors using priority based scheduling (Tapale et al. 2019). The author uses the most powerful meta-heuristic approach i.e. Imperialist competitor algorithms and obtains low performance time (Habibi and Navimipour 2016). To move forward in complex engineering and management processes, the author uses Bee colony optimization technology to reduce execution time(Phani Sheetal and Ravindranath 2019). Author improves performance time, power consumption and cost by using social learning optimization (Zhizhong Liu et al. 2017). According to the weighted queue, the work is arranged and then fixed in the SJF manner. And then author gets optimal results through IOT mobile devices (Preethi and Jayavel 2018). Author gets lower makespan by using deadline & suffrage aware task scheduling in referenced article (Chaudhary and Varsha 2017). The referenced article improves makespan, cost and resource usage. These are the key goals of this optimization model is written here. Model uses lion optimization and adversary optimization to show good performance (Krishnadoss and Prem Jacob 2019). The author uses the basic priority concept to reduce execution time and get updated results. Round robin scheduling works with quantum time duration in which the time is equal to execution of each task (Amit Agarwal and Saloni Jain 2014). Each task gets its value and according to their surffage value, each job goes to evaluate. The author improves makespan and resource usage by using time- based suffrage algorithm (Krishnaveni et al. 2018). Writer reduces makespan by applying a development-based assessment strategy (Ahluwalia and Sharma 2016). The combination of genetic algorithm and fuzzy logic is very effective to get the best results. The author reduces the allocation cost by using this combination (Dahiya 2015). The author improves execution time, resource usage and cost from implementing a novel algorithm based on cost criteria (Mohammed et al. 2016). The author lowers the cost parameter by considering the driver's strategy in the dynamic path (Xie et al. 2016). Least level value density first Method applied in paper to overcome situations related to proper use of cloud resources with appropriate time constraints (Kuang and Zhang 1864). The author uses a population- based incremental learning and genetic algorithm to show a better estimate of distribution algorithms and accelerates convergence speed faster (Sun et al. 1955). The author improves execution time and resource usage by applying an ideal algorithm (Kanna Babu and Sree Latha 2018). Considering heavy weight on cloud, the author designs a new algorithm i.e. random resource selection algorithm. Applying it to real problems, the author shows good response times by performed simulation (Vijayalakshmi and Venkatesa Kumar 2014). The author saves energy by applying a combination of three types of algorithms. These are: Energy Aware VM available time scheduling, heuristic algorithms and credit scheduling (Loganathan et al. 2017). The author uses the best search optimization algorithm which saves energy. This customizable algorithm defines with firefly and bats algorithm features (M. Senthil kumar 2018). By achieving the minimum value of the makespan, it means that the resources are properly used for execution. The author uses Differential Evolution-Gravitational Search Algorithms to reduce makespan in the proposed paper (Sharma and Tyagi 2016). By grouping the tasks with similar specification, author decreases the makespan by taking the superiority of k mean algorithm (Dandhwani andVipul Vekariya 2016). Author improves decent factor i.e. local optima by using Particle Swarm Optimization and Shuffled Frog Leaping Algorithm (Xie et al. 2016). Author improves response time, waiting time, processing cost and power consumption by using skewness algorithm (Kaur and Navtej 2017). Using Black Hole based algorithm, the author improves makespan and resource usage (Ebadifard and Borhanifard 2016). Author reduces waiting time for the tasks waiting for execution in a queue by applying the fruitfulness of fuzzy technology (Kundu 2015). Author improves quality of service, resource utilization and power saving by using fuzzy. It controls Vm's configuration to serve the proposed theory (Goswami et al. 2018). To get less performance time and cost, the article proposed by the author has one or more objectives under consideration (Leena et al. 2016). The paper shows the reduction in cost parameters using the CloudSim toolkit with different number of virtual machines using priority based on user needs (Bansal et al. 2015).

4 Categorization and discussion

Various task scheduling algorithms has been implemented by providing various benefits for user as well as service provider perspective. After analysing many novel research articles and review articles, found that all are talking about the algorithm and focusing on a single or many more factor whereas other important factors are in challenge in today's market. Security is taking a big place in challenge that how can one process the task securely on the virtual machines. No paper has been written about secure task scheduling algorithms in cloud computing. The main objective of this-paper is to show the actual scenario of the algorithms and concern objectives (Table 1).

Various reputed publications are publishing research articles day by day with full honesty and sincerity. Figure 5 showing a chart representation for deliberative purposes in publishing houses with data isolation. Publishing companies have placed their distinctive value in development, given the most appropriate concern in the research era. Elsevier is one of the most famous or valuable publishing company that specialized for scientific proposals and articles. Annually it publishes around 480,000 research material considering diverse fields. Springer is holding a great value with benefits in the same field by publishing ebooks, articles, and new thinking towards the wounds of humanity. The total number of published books and papers is more than 170,000, established in Springer's database with various fields of study. IEEE has a name, Institute of Electrical and Electronics Engineers, which has the largest body of professionals. It is associated with over 423,000 technical persons. Many more gateways exist in publications, including attractive artefacts that focus primarily on research and development. The summary chart is displayed from Fig. 6, The conclusion is that more research work has been done on the makespan factor only. The researcher should also focus on other factors to ensure the benefits offered by the services provided through cloud computing (Vijayalakshmi and Venkatesa Kumar 2014).

A novel algorithm can be based on any technology but it must reliable in any circumstances. An algorithm must produce optimized results in anyway. The flow of calculated parameters has been conducted mostly on distributed and focused utility. Trust and demands are directly proportional to reducing the unwanted response of the fiasco. The results should always be in favor of positive decisions so that the possibility of satisfaction of the users can be considered.

Multi issues and challenges related to task scheduling in cloud computing justified in the paper. Optimization of the digital world is involved with this issue with full appropriation. Multi scale architecture and more advanced versions of the technology may resolve this issue according to the coming requests. This plan would be very effective to build support to create a highly advanced era.

Table 1 Various publications with various factors

Name of factor publications	Throughput (1)	Makespan (2)	Power consumption (3)	Local optima (4)	Cost (5)	Availability of resources (6)	Schedule length (7)	Resource utilization (8)	Bandwidth (9)	QoS (10)	Fault tolerance (11)
Elsevier	5	27	7	2	5	1	2	14	0	2	0
Springer	2	24	2	1	6	0	0	18	2	2	0
IEEE	2	23	3	1	9	0	3	20	1	2	1
Others	2	24	5	2	11	0	0	15	2	2	0

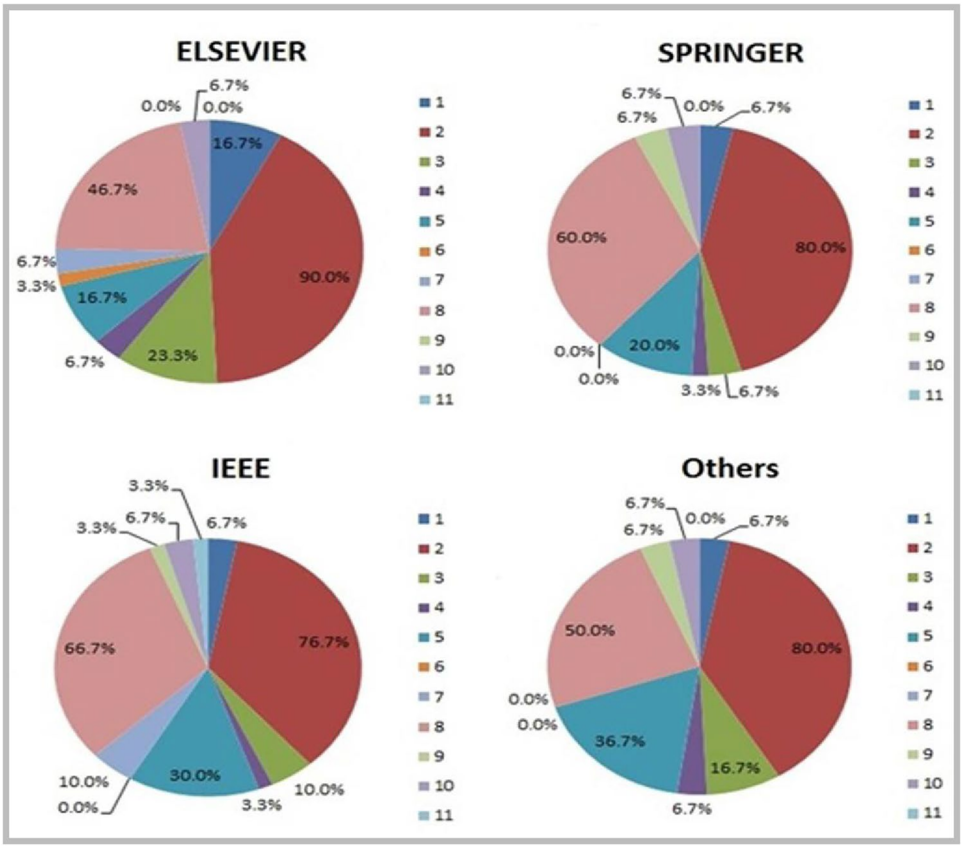

Fig. 5 Analysis of publications with various factors

5 Challenges and issues

Task scheduling needs more accuracy towards utilization and activities performances, promising with the user by service provider. To get optimal results, cloud computing is comply with many challenges such as: Charismatic and vibrating workloads: Due to heavy and uncertain workload, adaptability becomes a major challenge in cloud computing. If predictable variation came during execution, that may be cure in advance while designing the program with proper management of resources within the time frame. Assuring profitable resource utilization: According to the fluctuations in user's demand, assignment of the resources has to be done immediately. Service provider has to keep the knowledge related with the intake workloads, because of the agreement done with the user has to be accomplished. Optimal scheduling is in need due to this scenario to execute the tasks completely. Composite nodes in cloud datacentres: allocated tasks are accessible across the distributed network in distinct stations with fluctuating power, infrastructure, database repository, etc. Each node processes different tasks in each execution. Elevated granularity in optimized scheduling: If you want more customization, more problems will come automatically. The heavy workload becomes the support of technology. Most of the time machines are moving to incompatible states during processing due to the large number of applications. Migration of virtual machines is heavily processed by applying partitions to the heavyweights.

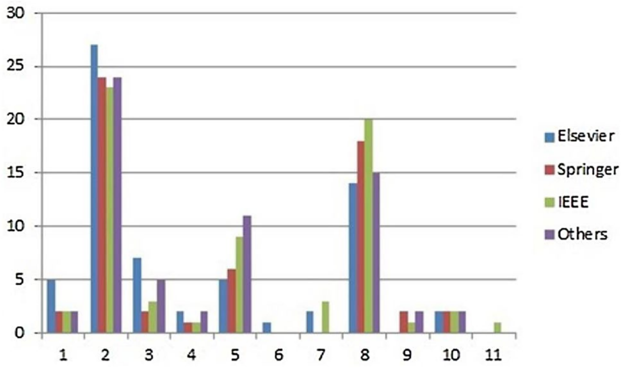

Fig. 6 Parameters frequency and flow

Many inimitable issues came in the administration of optimization achievement in cloud environment. Time and cost are the common factors for a scheduling in cloud computing. To improve the value for these two is a very difficult task. To accomplish this many scheduling are in publication process. As the requirement of the resources is growing day by day, allocation to every node of datacentre becomes a very tuff scenario. Need for monitoring: government organizations need modernize system that would be capable for

Table 2 Comparison of various objectives based task scheduling Cloud Computing

Refs	Simulator/environment	Scheduling objective										
		Throuhgt put	Makes-pan (Time)	Power consump-tion	Local optima	Cost	Availability of resources	Sched-ule length	Resource utilization	Bandwidth	QoS	Fault tolerance
Satish (2019)	CloudSim	✓	✓			✓			✓			
Mansouri et al. (2019); Dhinesh Babu and Venkata Krishna (2013); Phani Sheetal and Ravindranath (2019)	CloudSim	✓	✓						✓			
Salamy (2019); Jena (2017)	Real time		✓	✓					✓			
Kumar Panda et al. (2018); Doreen et al. (2015); Nayak and Padhi (2018); Soualhia et al. (2018); Bitam (2012); Devipriya and Ramesh (2013); Krishnaveni and Sinthu (2018b); Kundu (2015)	Cloud		✓									
ChandanNayak (2018)	Nebula cloud		✓									
Mousavi (2018)	CloudSim	✓	✓		✓				✓			
Li (2018)	Cloud		✓	✓								
Tian et al. (2018)	Java/cloud		✓	✓								
Srichandan (2018)	MaTLab/cloud		✓	✓								
Rodriguez and Buyya (2018)	Cloud		✓			✓						
Sotiriadis and Bessis (2018); Mohit Kumar and Sharma (2017); Gupta et al. (2017); Sanjaya (2018); Ma et al. (2012); Jyoti Gupta et al. (2016); Gupta and Garg (2017); George Amalarethinam and Kavitha (2017); Zhang et al. (2016); Ettikyala (2016); Mittal and Katal (2016); Gajera et al. (2016); Alaka Ananth (2015); Panda et al. (2014); Panda and Jana (2015); Li et al. (2011); Zhenxia (2008); Amit Agarwal Saloni Jain (2014); Krishnaveni et al. (2018); Sharma and Tyagi (2016); Ebadifard and Borhanifard (2016)	CloudSim		✓						✓			
Sebastio (2017)	MaTLab/cloud		✓		✓							
Wei et al. 2012)	Grid		✓						✓			
Weihong chen et al. (2017)	Java					✓		✓				

Table 2 (continued)

Refs	Simulator/environment	Scheduling objective										
		Throuhgt put	Makes-pan (Time)	Power consump-tion	Local optima	Cost	Availability of resources	Sched-ule length	Resource utilization	Bandwidth	QoS	Fault tolerance
KalkaDubey and Sharma (2017); Ellendula Madhukara and Thiru-malaisamy Ragunathan (2015); Antony Thomas et al. (2015); Zadeh and Hashemi (2013); Tamanna Jena (2018); Parthasar-athy (2017); Wang (2013); Ahlu-walia and Sharma (2016); Sun et al. (1955); Vijayalakshmi and Ven-katesa Kumar (2014); Dandhwani and Vipul Vekariya (2016)	CloudSim		✓									
Awad et al. (2015)	CloudSim		✓			✓			✓			
Sanjaya (2015a)	Matlab/cloud		✓						✓			
Jena (2015)	CloudSim		✓	✓								
Xiaonian et al. (2013)	CloudSim		✓				✓		✓		✓	
Jiayin Li (2012)	Cloud								✓			
Fahmy (2010)	Real time	✓	✓									
Jeyarani et al. (2011)	Virtual			✓								
Aida (2018)	CloudSim		✓			✓		✓	✓		✓	
Muhammad Shahzad Arif (2019)	CloudSim		✓			✓						
Kashikolaei et al. (2019; Sanjaya (2015b); Zhan and Zhang (2014); Kuang and Zhang (1864); Kanna Babu and Sree Latha (2018)	Cloud		✓						✓			
Panda and Jana (2019)	Cloud								✓			
Sanjaya (2019a)	Cloud		✓	✓								
Gawali and Shinde (2018)	CloudSim		✓						✓	✓		
Agarwal and Srivastava (2018)	CloudSim		✓						✓		✓	
Phani Praveen et al. (2018)	CloudSim	✓	✓									
Krishnaveni and Sinthu (2018a)	Java/cloud		✓						✓			
Valarmathi and Sheela (2017)	CloudSim		✓			✓			✓			
Alworafi et al. (2017)	CloudSim					✓				✓		
Sanjaya (2017)	CloudSim		✓			✓			✓			
Sanjaya (2019b)	Cloud	✓	✓						✓			
Moon et al. (2017)	Cloud		✓		✓				✓			
Ajeena Beegom and Rajasree (2015)	Cloud		✓			✓						

Table 2 (continued)

Refs	Simulator/environment	Scheduling objective										
		Throuhgt put	Makes-pan (Time)	Power consump-tion	Local optima	Cost	Availability of resources	Sched-ule length	Resource utilization	Bandwidth	QoS	Fault tolerance
Panda and Jana (2016)	Cloud			✓					✓			
Javanmardi et al. (2014)	CloudSim		✓			✓						
Chiu et al. (2014)	Cloud								✓			
Fang et al. (2010)	CloudSim								✓			
Shatha (2010)	Real time		✓									
Firdhous et al. (2011)	Cloud										✓	
Yin et al. (2018)	CloudSim					✓			✓			
Li et al. (2017)	Cloud			✓								
Gupta and Tewari (2017)	Cloud		✓			✓			✓		✓	✓
Zhang and Zhou (2018)	Cloud		✓			✓						
Xiaolong et al. (2016)	Cloud			✓					✓			
Chauhan and Jaglan (2016)	Cloud		✓					✓				
Kaur and Kaur (2016)	Cloud	✓	✓			✓			✓			
Himani and Sidhu (2015)	CloudSim	✓				✓						
Rana et al. (2014)	CloudSim		✓			✓			✓			
Anjuli Garg and Krishna (2014)	CloudSim		✓			✓						
Panda and Jana (2014)	Cloud			✓					✓			
Thanasias et al. (2014)	Cloud								✓			
Omara and Zohier (2010)	Not mentioned				✓				✓			
Panda and Jana (2015)	CloudSim		✓			✓			✓			
Topcuoglu and Hariri (1999)	Not mentioned		✓			✓		✓		✓		
El-Rewini et al. (1995)	Machine Learning		✓					✓			✓	
Aujla (2015)	Visual studio/cloud		✓									
Tapale et al. (2019)	CloudSim		✓			✓			✓	✓		
Habibi and Navimipour (2016)	MATLAB/cloud		✓									
Zhizhong Li et al. (2017)	CloudSim		✓	✓	✓	✓			✓		✓	
Preethi and Jayavel (2018)	CloudSim								✓			
Chaudhary and Varsha (2017; Krishnadoss and Prem Jacob (2019)	CloudSim		✓			✓			✓			
Dahiya (2015)	MATLAB/cloud					✓						
Mohammed et al. (2016)	CloudSim		✓			✓			✓	✓		
Xie et al. (2016)	MATLAB/cloud		✓			✓						
Loganathan et al. (2017)	CloudSim			✓								

Table 2 (continued)

Refs	Simulator/environment	Scheduling objective										
		Throuhgt put	Makes-pan (Time)	Power consump-tion	Local optima	Cost	Availability of resources	Sched-ule length	Resource utilization	Bandwidth	QoS	Fault tolerance
M. Senthil kumaR 2018)	MapReduce/	✓	✓	✓		✓						
Xie et al. (2016)	CloudSim				✓							
Kaur and Ghummsan (2017)	Cloud		✓	✓		✓						
Goswami et al. (2018)	Cloud			✓					✓		✓	
Leena et al. (2016)	CloudSim		✓			✓						
Bansal et al. (2015)	CloudSim					✓						

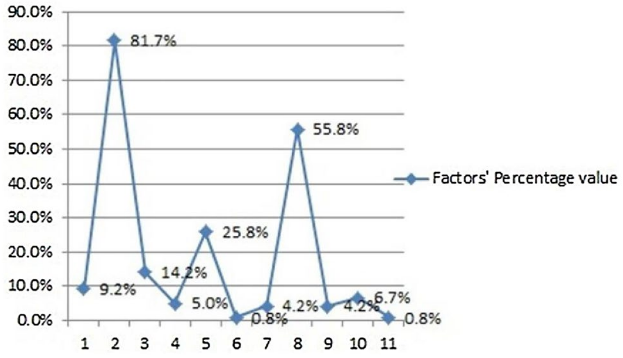

Fig. 7 Considered objective performances in the literature

grasping the new technology easily. But this is not a feasible situation for cloud service providers. Due to security reasons, consumers are going through indecision by acquiring more number of service providers. Doubt increases because users are very possessive for their data. As the technology is burdened due to unnatural demands of the users, the organization is not getting satisfactory results from its employers. It faces an uncontrollable situation from the market point of view, because employers have not improved their skills and are lacking to remove the unexpected error from systems. So organizations are moving to get the benefits from the cloud by using its services on payment basis. Identity recognition of cloud service provider: No one can judge the identity of service provider and how it will provide the services in best way. Based on verification card only, one can know that this is the authorized dealer for the cloud services. ICT consultant will provide the authenticity for providers without any factiousness.

6 Conclusion and future scope

Many complex applications can now achieve deployment of new paradigms through cloud computing and generate good value due to its distributed and administrative environments. No need to take care of physical devices to develop things, this is a best feature provided by cloud computing. Large-end applications can execute independently regardless of hardware requirement specifications by offering services such as cloud usage through virtualization. The flow of information can move from one region to another by crossing several kilometres within just a second, the only reason for cloud computing. By introducing different, unique and nature based algorithms, more and more optimization is going on. In addition, security is also an essential direction for better-optimized algorithms, which can be named as secure-trust-algorithms (STAs). This future proposed algorithm may consider factors such as access control models along with cryptographic techniques for further improvement towards cloud security. By generalizing the usage control model, some facts are included and some are not. The considered facts are designated as authority, obligations, conditions, active control and variability. Non-considered factors are designated as mandatory, discretionary and role-based access control. These factors are processed by the UCONABC model (Park and Sandhu 2004; Rajkumar and Sandhu 2020). This model covers all the facts whether it is traditional or adapted. Finally, the paper explores the literature survey with all measurement parameters analyzed for task scheduling performance through different platforms and architectures to process multiple applications at the same time. Apart from this access control is also in the picture to cover the key points of the security mechanism with smart IoT devices. A fine-grain access model with specified policies (Bhatt et al. 2021; Ge et al. 2013) has also become very useful. It also uses an encryption model that removes the traditional use of public-key encryption by introducing a certificate less encryption policy (Ge et al. 2009). By integrating different paradigms, it is possible to offer a powerful perspective to task scheduling in cloud computing with a more hybridized approach with multi-purpose applications. The entire theory shows the value of the factor based on market demands in the current situation. Through which new papers can include those factors, which have less experimentation. In a beneficial context, lower usage factors can give much efficiency to incoming applications for greater loyalty.

A table helps to demonstrate a clear approach to literature in any context. This paper also creates a Table 2 for the entire literature survey to consider various factors for task scheduling algorithms in a cloud environment.

All scheduling objectives are the main points for literature design. There are also many literary series to say that other factors should also be taken into consideration, although the ratio of failure to secure the system will always underline the problems and issues. Performance and analysis of the requirement may perhaps be influenced by this ratio. In order to make the system reliable and scalable for upcoming problems, problems should be considered in future objectives according to statements in already published articles.

In future more factors must be optimized for better execution so that all scheduling features can be provided in a single scheduling algorithm. Evaluation has been done for the mentioned factors with different publication values, depicted by Fig. 7.

Acknowledgements Not Applicable

Author contribution NB (PhD student) wrote the manuscript and designed the graphs. AKS (phd Supervisor) read and approved the manuscript.

Funding Self.

Data availability References included.

Declarations

Conflict of interest Author have no conflict of interest.

References

Agarwal M, Srivastava GMS (2018) A cuckoo search algorithm-based task scheduling in cloud computing, Adv Comput Comput Sci

Agarwal A, Jain S (2014) Efficient optimal algorithm of task scheduling in cloud computing environment. Int J Comput Trends Technol (IJCTT) 9(7):344–349

Ahluwalia A, Sharma V (2016) Differential evolution based optimal task scheduling in cloud computing. Int J Adv Res Comput Sci Softw Eng 6(6):340–347

AS Ajeena Beegom, MS Rajasree (2015) Genetic algorithm framework for bi-objective task scheduling in cloud computing systems, Lect Notes Comput Sci 356–359 https://doi.org/10.1007/978-3-319-14977-6_38

Ananth A, Chandrasekaran K (2015) Cooperative game theoretic approach for job scheduling in cloud computing. In: International Conference on Computing and Network Communications (CoCoNet) https://doi.org/10.1109/CoCoNet.2015.7411180

Alworafi MA, Al-Hashmi A, Dhari A, Suresha A, Darem B (2017) Task-scheduling in cloud computing environment cost priority approach lecture notes in networks and systems. Springer, Singapore, pp 99–108

Arif MS, Iqbal Z, Tariq R, Aadil F, Awais M (2019) Parental prioritization-based task scheduling in heterogeneous systems. Arab J Sci Eng 44(4):3943–3952. https://doi.org/10.1007/s13369-018-03698-2

Aujla S, Ummat A (2015) Task scheduling in cloud using hybrid cuckoo algorithm. Int J Comput Netw Appl (IJCNA) 2:3

Awad AI, El-Hefnawy NA, Abdel Kader HM (2015) Enhanced particle swarm optimization for task scheduling in cloud computing environments. Proc Comput Sci 65:920–929. https://doi.org/10.1016/j.procs.2015.09.064

Bansal N, Singh AK (2018) Trust for task scheduling in cloud computing unfold it through fruit congenial. Netw Commun Data Knowl Eng 4:41–48. https://doi.org/10.1007/978-981-10-4600-1_4

Bansal N, Maurya A, Kumar T, Singh M, Bansal S (2015) Cost performance of Qos driven task scheduling in cloud computing. Proc Comput Sci 57:126–130. https://doi.org/10.1016/j.procs.2015.07.384

Bansal N, Awasthi A, Bansal S (2016) Task scheduling algorithms with multiple factor in cloud computing environment. Inf Syst Des Intell Appl 433:619–627. https://doi.org/10.1007/978-81-322-2755-7_64,PrintISBN978-81-322-2753-3

Bansal N, Dutta M (2014) Performance evaluation of task scheduling with priority and non-priority in cloud computing. In: IEEE International Conference on Computational Intelligence and Computing Research pp. 1–4 https://doi.org/10.1109/ICCIC.2014.7238289.

Bhatt S, Pham TK, Gupta M, Benson J, Park J, Sandhu R (2021) Attribute-based access control for AWS internet of things and secure industries of the future. IEEE Access 9:107200

Bitam S (2012) Bees life algorithm for job scheduling in cloud computing. In: International Conference on Communications and Information Technology (ICCIT)

Chandan Nayak S, Parida S, Tripathy C, Kumar Pattnaik P (2018) An enhanced deadline constraint based task scheduling mechanism for cloud environment. J King Saud Univ Comput Inf Sci. https://doi.org/10.1016/j.jksuci.2018.10.009

Chaudhary P, Varsha D (2017) Deadline and suffrage aware task scheduling approach for cloud environment. Int Res J Eng Technol (IRJET) 4:8

Chauhan PK, Jaglan P, Dabas P (2016) An intensify deadline aware credit based cloud task scheduling. In: International Conference on Computing, Communication and Automation (ICCCA) https://doi.org/10.1109/CCAA.2016.7813911.

Chiu C-F, Hsu SJ, Jan S-R, Chen J-A (2014) Task scheduling based on load approximationin cloud computing environment. Future Inf Technol 309:803–808. https://doi.org/10.1007/978-3-642-55038-6_122

Dahiya S, Preety A (2015) Scheduling of independent tasks in cloud computing using modified genetic algorithm (FUZZY LOGIC). Int J Comput Sci Mob Comput 4(9):199–207

Dandhwani V, Vekariya V (2016) Multi-objective task scheduling using K-mean algorithm in cloud computing. Int J Innov Res Comput Commun Eng 4(11):19521–19524. https://doi.org/10.15680/IJIRCCE.2016

Devipriya S, Ramesh C (2013) Improved max-min heuristic model for task scheduling in cloud. In: 2013 International Conference on Green Computing, Communication and Conservation of Energy https://doi.org/10.1109/ICGCE.2013.6823559

Dhinesh Babu LD, Venkata Krishna P (2013) Honey bee behaviour inspired load balancing of tasks in cloud computing environments. Appl Soft Comput 13(5):2292–2303. https://doi.org/10.1016/j.asoc.2013.01.025

Doreen D, Miriam H, Prabha B, Felicia Lilian J (2015) An efficient job scheduling in isometric HPCLOUD using ZBLA optimization. Proc Comput Sci 50:307–315. https://doi.org/10.1016/j.procs.2015.04.043

Dubey K, Kumar M, Sharma SC (2017) Modified HEFT algorithm for task scheduling in cloud environment. Proc Comput Sci 125:725–732. https://doi.org/10.1016/j.procs.2017.12.093

Ebadifard F, Borhanifard Z (2016) A modified black hole-based task scheduling technique for cloud computing environment. Comput Eng Appl 5(2):77–90. https://doi.org/10.18495/comengapp.v5i2.172

El-Rewini H, Ali HH, Lewis T (1995) Task scheduling in multiprocessing systems. Computer 28(12):27–37. https://doi.org/10.1109/2.476197

Ettikyala K, Latha YV (2016) Rank based efficient task scheduler for cloud computing. In: International Conference on Data Mining and Advanced Computing (SAPIENCE) https://doi.org/10.1109/SAPIENCE.2016.7684151

Fahmy MMM (2010) A fuzzy algorithm for scheduling non-periodic jobs on soft real-time single processor system. Ain Shams Eng J 1(1):31–38. https://doi.org/10.1016/j.asej.2010.09.004

Fang Y, Wang F, Ge J (2010) A task scheduling algorithm based on load balancing in cloud computing. Lect Notes Comput Sci 6318:271–277. https://doi.org/10.1007/978-3-642-16515-3_34

Firdhous M, Ghazali O, Hassan S (2011) Applying bees algorithm for trust management in cloud computing. Lect Notes Inst Comput Sci 103:224–229. https://doi.org/10.1007/978-3-642-32711-7_21

Gajera V, Gupta R, Jana PK (2016) An effective multi-objective task scheduling algorithm using min-max normalization in cloud computing. In: International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT) https://doi.org/10.1109/ICATCCT.2016.7912111.

Garg A, Rama Krishna C (2014) An improved honey bees life scheduling algorithm for a public cloud. In: International Conference on Contemporary Computing and Informatics (IC3I) https://doi.org/10.1109/IC3I.2014.7019783.

Gawali MB, Shinde SK (2018) Task scheduling and resource allocation in cloud computing using a heuristic approach. J Cloud Comput. https://doi.org/10.1186/s13677-018-0105-8

Ge A, Zhang J, Zhang R, Ma C, Zhang Z (2013) Security analysis of a privacy-preserving decentralized key-policy attribute-based encryption scheme. IEEE Trans Parallel Distrib Syst 24(11):2319

Ge A, Chen S, Huang X (2009) A concrete certificateless signature scheme without pairings. In: IEEE International Conference on Multimedia Information Networking and Security

George Amalarethinam DI, Kavitha S (2017) Priority based performance improved algorithm for meta-task scheduling in cloud environment. In: International Conference on Computing and Communications Technologies (ICCCT) https://doi.org/10.1109/ICCCT2.2017.7972250.

Goswami V, Shrivastava RK (2018) HMM and fuzzy logic based algorithm for efficient task scheduling and resource management in cloud systems. Int J Math Trends Technol (IJMTT) 54(4):341–354. https://doi.org/10.14445/22315373/IJMTT-V54P540

Gupta A, Garg R (2017) Load balancing based task scheduling with ACO in cloud computing. Int Conf Comput Appl (ICCA). https://doi.org/10.1109/COMAPP.2017.8079781

Gupta P, Tewari P (2017) Monkey search algorithm for task scheduling in cloud IaaS. Int Conf Image Inf Process (ICIIP). https://doi.org/10.1109/ICIIP.2017.8313789

Gupta J, Azharuddin MD, Jana PK (2016) An effective task scheduling approach for cloud computing environment. Lect Notes Electr Eng 396:163–169. https://doi.org/10.1007/978-81-322-3589-7_17

Gupta I, Kaswan A, Jana PK (2017) A flower pollination algorithm based task scheduling in cloud computing. Commun Comput Inf Sci. https://doi.org/10.1007/978-981-10-6430-2_9

Habibi M, Navimipour NJ (2016) Multi-objective task scheduling in cloud computing using an imperialist competitive algorithm. Int J Adv Comput Sci Appl (IJACSA). https://doi.org/10.14569/IJACSA.2016.070540

Jadhao SR, Amdani SY (2019) Performance related trade-offs between fairness and throughput for job scheduling in cloud environment. Proc Comput Sci 152:122–129. https://doi.org/10.1016/j.procs.2019.05.034

Javanmardi S, Shojafar M, Amendola D, Cordeschi N, Liu H, Abraham A (2014) Hybrid job scheduling algorithm for cloud computing environment. Adv Intell Syst Comput. https://doi.org/10.1007/978-3-319-08156-4_5

Jena RK (2015) Multi objective task scheduling in cloud environment using nested PSO framework. Proc Comput Sci 57:1219–1227. https://doi.org/10.1016/j.procs.2015.07.419

Jena RK (2017) Energy efficient task scheduling in cloud environment. Energy Proc 141:222–227. https://doi.org/10.1016/j.egypro.2017.11.096

Jena T, Mohanty JR (2018) GA-based customer-conscious resource allocation and task scheduling in multi-cloud computing. Arab J Sci Eng 43(8):4115–4130. https://doi.org/10.1007/s13369-017-2766-x

Jeyarani R, Nagaveni N, Vasanth Ram R (2011) Design and implementation of adaptive power-aware virtual machine provisioned (APA-VMP) using swarm intelligence. Future Gener Comput Syst 28(5):811–821. https://doi.org/10.1016/j.future.2011.06.002

Kadhim SJ, Al-Aubidy KM (2010) Design and evaluation of a fuzzy-based CPU scheduling algorithm. Commun Comput Inf Sci 70:45–52. https://doi.org/10.1007/978-3-642-12214-9_9

Kanna Babu AMK, Sree Latha M (2018) Efficient ideal algorithm for task scheduling in cloud computing. Int J Eng Technol 7(3):5

Kashikolaei SMG, Hosseinabadi AAR, Saemi B, Shareh MB, Sangaiah AK, Bian G-B (2019) An enhancement of task scheduling in cloud computing based on imperialist competitive algorithm and firefly algorithm. J Supercomput 76:1–28. https://doi.org/10.1007/s11227-019-02816-7

Kaur S, Ghumman NS (2017) A novel approach of task classification and Vm skewness in cloud environment. Int J Comput Technol 16(7):6994–7001. https://doi.org/10.24297/ijct.v16i7.6416

Kaur A, Kaur B (2016) Load balancing in tasks using honey bee behaviour algorithm in cloud computing. Int Conf Wireless Netw Embed Syst (WECON). https://doi.org/10.1109/WECON.2016.7993460

Krishnadoss P, Jacob P (2019) OLOA: based task scheduling in heterogeneous clouds. Int J Intell Eng Syst 12(1):114. https://doi.org/10.22266/ijies2019.0228.12

Krishnaveni H, Sinthu Janita V (2018) Completion time based sufferage algorithm for static task scheduling in cloud environment. Int J Pure Appl Math 119(12):61–70

Krishnaveni H, Sinthu Janita Prakash V (2018a) Execution time based sufferage algorithm for static task scheduling in cloud. Adv Big Data Cloud Comput. https://doi.org/10.1007/978-981-13-1882-5_5

Krishnaveni H, Sinthu Janita Prakash V (2018b) Multi attribute completeness measure based shortest job first algorithm in cloud environment. Int J Sci Res Comput Sci Eng Inf Technol (IJSRCSEIT) 3:3

Kuang L, Zhang L (1864) A new task scheduling algorithm based on value and time for cloud platform. AIP Conf Proc 1864(1):2017. https://doi.org/10.1063/1.4992834

Kumar M, Sharma SC (2017) Dynamic load balancing algorithm for balancing the workload among virtual machine in cloud computing. Proc Comput Sci 115:322–329. https://doi.org/10.1016/j.procs.2017.09.141

KumarPanda S, SurachitaNanda S, KumarBhoi S (2018) A pair-based task scheduling algorithm for cloud computing environment. J King Saud Univ Comput Inf Sci. https://doi.org/10.1016/j.jksuci.2018.10.001

Kundu A (2015) A new approach for task scheduling of cloud computing using fuzzy. Int J Innov Res Comput Sci Technol (IJIRCST) 3(2):112–116

Leena VA, Ajeena Beegom AS, Rajasree MS (2016) Genetic algorithm based bi-objective task scheduling in hybrid cloud platform. Int J Comput Theory Eng 8(1):7–13. https://doi.org/10.7763/IJCTE.2016.V8.1012

Li K (2018) Scheduling parallel tasks with energy and time constraints on multiple many core processors in a cloud computing environment. Futur Gener Comput Syst 82:591–605. https://doi.org/10.1016/j.future.2017.01.010

Li K, Gaochao Xu, Zhao G, Dong Y, Wang D (2011) Cloud task scheduling based on load balancing ant colony. Optimization. https://doi.org/10.1109/ChinaGrid.2011.17

Li J, Qiu M, Ming Z, Quan G, Qin X, Gu Z (2012) Online optimization for scheduling preemptable tasks on IaaS cloud systems. J Parallel Distrib Comput 72(5):666–677. https://doi.org/10.1016/j.jpdc.2012.02.002

Li Y, Chen M, Dai W, Qiu M (2017) Energy optimization with dynamic task scheduling mobile cloud computing. IEEE Syst J 11(1):96–105. https://doi.org/10.1109/JSYST.2015.2442994

Lin W, Lihua A (2013) Task scheduling policy based on ant colony optimization in cloud computing environment. LISS. https://doi.org/10.1007/978-3-642-32054-5_133

Liu Z, Qin J, Peng W, Chao H (2017) Effective task scheduling in cloud computing based on improved social learning optimization algorithm. Int J Online Biomed Eng (iJOE) 13(6):4

Loganathan S, Saravanan RD, Mukherjee S (2017) Energy aware resource management and job scheduling in cloud datacenter. Int J Intell Eng Syst 10(4):175

Ma L, Yueming Lu, Zhang F, Sun S (2012) Dynamic task scheduling in cloud computing based on greedy strategy. Commun Comput Inf Sci 320:156–162. https://doi.org/10.1007/978-3-642-35795-4_20

Madhukar E, Ragunathan T (2015) Efficient scheduling algorithm for cloud. Proc Comput Sci 50:353–356. https://doi.org/10.1016/j.procs.2015.04.036

Mansouri N, Zade BMH, Javidi MM (2019) Hybrid task scheduling strategy for cloud computing by modified particle swarm optimization and fuzzy theory. Comput Ind Eng 130:597–633. https://doi.org/10.1016/j.cie.2019.03.006

Mittal S, Katal A (2016) An optimized task scheduling algorithm in cloud computing. Int Conf Adv Comput (IACC). https://doi.org/10.1109/IACC.2016.45

Moon Y, Yu H, Gil J-M, Lim J (2017) A slave ants based ant colony optimization algorithm for task scheduling in cloud computing environments. Hum Cent Comput Inf Sci. https://doi.org/10.1186/s13673-017-0109-2

Mosleh MAS, Radhamani G, Hazber MAG, Hasan SH (2016) Adaptive cost-based task scheduling in cloud environment. Sci Program 2016:1–9. https://doi.org/10.1155/2016/8239239

Mousavi S, Mosavi A, Várkonyi-Kóczy A (2018) A load balancing algorithm for resource allocation in cloud computing. 660:289-296 https://doi.org/10.1007/978-3-319-67459-9_36

Nasr AA, El-Bahnasawy NA, Attiya G, El-Sayed A (2018) A new online scheduling approach for enhancing QOS in cloud. Future Comput Inf J 3(2):424–435. https://doi.org/10.1016/j.fcij.2018.11.005

Nayak B, Padhi SK, Pattnaik PK (2019) Static task scheduling heuristic approach in cloud computing environment. Inf Syst Des Intell Appl. https://doi.org/10.1007/978-981-13-3329-3_44

Omara FA, Zohier RM(2010) Dynamic task scheduling using fuzzy logic in distributed memory systems. In: International Conference on Informatics and Systems (INFOS)

Panda SK, Jana PK (2015b) Efficient task scheduling algorithms for heterogeneous multi-cloud environment. J Supercomput 71(4):1505–1533. https://doi.org/10.1007/s11227-014-1376-6

Panda SK, Jana PK (2016) An efficient task consolidation algorithm for cloud computing systems. Lect Notes Comput Sci. https://doi.org/10.1007/978-3-319-28034-9_8

Panda SK, Jana PK (2017) SLA-based task scheduling algorithms for heterogeneous multi-cloud environment. J Supercomput 73(6):2730–2762. https://doi.org/10.1007/s11227-016-1952-z

Panda SK, Jana PK (2018) Normalization-based task scheduling algorithms for heterogeneous multi-cloud environment. Inf Syst Front 20(2):373–399. https://doi.org/10.1007/s10796-016-9683-5

Panda SK, Jana PK (2019) Load balanced task scheduling for cloud computing: a probabilistic approach. Knowl Inf Syst 61:1–25. https://doi.org/10.1007/s10115-019-01327-4

Panda SK, Jana PK (2019a) An energy-efficient task scheduling algorithm for heterogeneous cloud computing systems. Clust Comput 22(2):509–527. https://doi.org/10.1007/s10586-018-2858-8

Panda SK, Gupta I, Jana PK (2015a) Allocation-aware task scheduling for heterogeneous multi-cloud systems. Proc Comput Sci 50:176–184. https://doi.org/10.1016/j.procs.2015.04.081

Panda SK, Gupta I, Jana PK (2019b) Task scheduling algorithms for multi-cloud systems: allocation-aware approach. Inf Sys Front 21(2):241–259. https://doi.org/10.1007/s10796-017-9742-6

Panda SK, Jana PK (2014) An efficient energy saving task consolidation algorithm for cloud computing systems. In: International Conference on Parallel, Distributed and Grid Computing https://doi.org/10.1109/PDGC.2014.7030753.

Panda SK, Jana PK (2015) An efficient task scheduling algorithm for heterogeneous multi-cloud environment. In: international conference on advances in computing, communications and informatics (ICACCI), https://doi.org/10.1109/ICACCI.2014.6968253.

Panda SK, Jana PK (2015) A multi-objective task scheduling algorithm for heterogeneous multi- cloud environment. In: international conference on electronic design, computer networks & automated verification (EDCAV) https://doi.org/10.1109/EDCAV.2015.7060544.

Panda SK, Nag S, Jana PK (2014) A smoothing based task scheduling algorithm for heterogeneous multi-cloud environment. In: international conference on parallel, distributed and grid computing https://doi.org/10.1109/PDGC.2014.7030716.

Park J, Sandhu R (2004) The UCONABC usage control model. ACM Trans Inf Syst Secur 7(1):128

Parthasarathy S, Venkateswaran CJ (2017) Scheduling jobs using oppositional-GSO algorithm in cloud computing environment. Wireless Netw 23(8):2335–2345. https://doi.org/10.1007/s11276-016-1264-5

Phani Praveen S, Thirupathi Rao K, Janakiramaiah B (2018) Effective allocation of resources and task scheduling in cloud environment using social group optimization. Arab J Sci Eng 43(8):4265–4272. https://doi.org/10.1007/s13369-017-2926-z

Phani Sheetal A, Ravindranath K (2019) Priority based resource allocation and scheduling using artificial bee colony (ABC) optimization for cloud computing systems. Int J Innov Technol Explor Eng (IJITEE) 8:6

Preethi M, Jayavel K (2018) IOT based visualization of weightage based static task scheduling algorithm in datacenter. Int J Eng Technol. https://doi.org/10.14419/ijet.v7i2.8.10478

Rajkumar PV, Sandhu R (2020) Safety decidability for pre-authorization usage control with finite attribute domains. IEEE Trans Dependable Secure Comput 17(3):465–478

Rana M, Bilgaiyan S, Kar U (2014) A study on load balancing in cloud computing environment using evolutionary and swarm based algorithms. In: international conference on control, instrumentation, communication and computational technologies (ICCICCT) https://doi.org/10.1109/ICCICCT.2014.6992964.

Rodriguez MA, Buyya R (2018) Scheduling dynamic workloads in multi-tenant scientific workflow as a service platforms. Future Gener Comput Syst 79(2):739–750. https://doi.org/10.1016/j.future.2017.05.009

Salamy DRH (2019) Task allocation, migration and scheduling for energy-efficient real-time multiprocessor architectures. J Syst Architect 98:17–26. https://doi.org/10.1016/j.sysarc.2019.06.003

Sidhu HS (2015) Cost-deadline based task scheduling in cloud computing. In: International Conference on Advances in Computing and Communication Engineering https://doi.org/10.1109/ICACCE.2015.86

Senthil M, kumar, (2018) Energy-aware task scheduling using hybrid firefly-BAT (FFABAT) in big data. Cybern Inf Technol 18(2):98–111. https://doi.org/10.2478/cait-2018-0031

Sharma A, Tyagi S (2016) Differential evolution- GSA based optimal task scheduling in cloud computing. Int J Eng Sci Res Technol 5(7):1447–1451. https://doi.org/10.5281/zenodo.58606

Sotiriadis S, Bessis N, Buyya Rk (2018) Self-managed virtual machine scheduling in cloud systems. Inf Sci 433–434:381–400. https://doi.org/10.1016/j.ins.2017.07.006

Soualhia M, Khomh F, Tahar S (2018) A dynamic and failure-aware task scheduling framework for hadoop. IEEE Trans Cloud Comput. https://doi.org/10.1109/TCC.2018.2805812

Srichandan S, Kumar TA, Datta SB (2018) Task scheduling for cloud computing using multi-objective hybrid bacteria foraging algorithm. Future Comput Inf J 3(2):210–230. https://doi.org/10.1016/j.fcij.2018.03.004

Stefano S, Giorgio G, Alberto B (2017) Optimal distributed task scheduling in volunteer clouds. Comput Oper Res 81:231. https://doi.org/10.1016/j.cor.2016.11.004

Sun H, Rui Xu, Chen H (1955) Independent tasks scheduling in cloud computing via improved estimation of distribution algorithm. AIP Conf Proc 1955(1):2018. https://doi.org/10.1063/1.5033826

Tapale MT, Goudar RH, Birje MN (2019) Adaptive scheduling mechanism in cloud. Int J Eng Adv Technol (IJEAT) 8:4

Thanasias V, Lee C, Helal S (2014) Budget-aware task scheduling in the cloud. Sixth Int Conf Ubiquitous Future Netw (ICUFN). https://doi.org/10.1109/ICUFN.2014.6876802

Thomas A, Krishnalal G, Jagathy Raj VP (2015) Credit based scheduling algorithm in cloud computing environment. Proc Comput Sci 46:913–920. https://doi.org/10.1016/j.procs.2015.02.162

Tian W, He M, Guo W, Huang W, Shi X, Shang M, Toosi AN, Buyya R (2018) On minimizing total energy consumption in the scheduling of virtual machine reservations. J Netw Comput Appl 113:64–74. https://doi.org/10.1016/j.jnca.2018.03.033

Topcuoglu H, Hariri S, Wu M-Y (1999) Task scheduling algorithms for heterogeneous processors. In: heterogeneous computing workshop (HCW'99), 1999, https://doi.org/10.1109/HCW.1999.765092.

Valarmathi R, Sheela T (2017) Ranging and tuning based particle swarm optimization with bat algorithm for task scheduling in cloud computing. Clust Comput 22:1–14. https://doi.org/10.1007/s10586-017-1534-8

Vijayalakshmi M, Venkatesa Kumar V (2014) Investigations on job scheduling algorithms in cloud computing. Int J Adv Res Comput Sci Technol (IJARCST) 2(1):157–161

Wei L, Zhang X, Li Y, Li Y (2012) An improved ant algorithm for grid task scheduling strategy. Phys Procedia 24:1974–1981. https://doi.org/10.1016/j.phpro.2012.02.290

Weihong C, Gugoi X, Renfa L, Yang B, Chunnian F, Keqin L (2017) Efficient task scheduling for budget constrained parallel applications on heterogeneous cloud computing systems. Future Gener Comput Syst 74:1–11. https://doi.org/10.1016/j.future.2017.03.008

Xiaolong Xu, Cao L, Wang X (2016) Resource pre-allocation algorithms for low-energy task scheduling of cloud computing. J Syst Eng Electron 27(2):457–469. https://doi.org/10.1109/JSEE.2016.00047

Xiaonian Wu, Deng M, Zhang R, Zeng B, Zhou S (2013) A task Scheduling algorithm based on QoS-driven in cloud computing. Proc Comput Sci 17:1162–1169. https://doi.org/10.1016/j.procs.2013.05.148

Xie X, Liu R, Cheng X, Hu X, Ni J (2016) Trust-driven and PSO-SFLA based job scheduling algorithm on cloud. Intell Autom Soft Comput 22(4):561–566

Yin S, Ke P, Tao L (2018) An improved genetic algorithm for task scheduling in cloud computing. IEEE Conf Ind Electr Appl (ICIEA). https://doi.org/10.1109/ICIEA.2018.8397773

Zadeh AM, Hashemi SM (2013) A novel-scheduling algorithm for cloud computing based on fuzzy logic. Int J Appl Inf Syst Found Comput Sci 5(7):28

Zhan Z-H, Zhang G-Y, Lin Y, Gong Y-J, Zhang J (2014) Load balance aware genetic algorithm for task scheduling in cloud computing. Lect Notes Comput Sci 8886:644–655. https://doi.org/10.1007/978-3-319-13563-2_54

Zhang C, Cui Y, Zheng R, Jinlong E, Jianping W (2016) Multi-resource partial-ordered task scheduling in cloud computing. In: International Symposium on Quality of Service (IWQoS) https://doi.org/10.1109/IWQoS.2016.7590423

Zhang PeiYun, Zhou MengChu (2018) Dynamic cloud task scheduling based on a two-stage strategy. IEEE Trans Autom Sci Eng 15(2):772–783. https://doi.org/10.1109/TASE.2017.2693688

Zhenxia Y, Fang M, Sheng S (2008) Scheduling algorithm based on task priority in heterogeneous computing environment. In:

international conference on computer science and information technology https://doi.org/10.1109/ICCSIT.2008.194

Zhiqiang X, Xia S, Yu X (2016) A scheduling algorithm for cloud computing system based on the driver of dynamic essential path. PLoS ONE 11:052. https://doi.org/10.1371/journal.pone.0159932

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.