Journal of Autonomous Intelligence (2024) Volume 7 Issue 5
doi: 10.32629/jai.v7i5.1483

ORIGINAL RESEARCH ARTICLE

Utilizing Advanced Group Search Optimization (AGSO) methodology for resource optimization in cloud computing

Ajay. P

Department of Research and Analysis, Publon Solutions Private Limited, 679522, India; ajaynair707@gmail.com

ABSTRACT

The escalating demand for resources within cloud data centers has accentuated the critical need for robust resource selection strategies implemented by customers. The inefficiencies prevalent in current resource utilization underscore the urgency of addressing this complex challenge. While metaheuristics, in comparison to traditional heuristics, showcase superior capabilities in efficiently scheduling large requests, their potential in selecting customer services can be further augmented by mitigating issues related to slow convergence speed and achieving a more equitable balance between local and global search. The overarching goal of the cloud computing platform is to furnish users with optimal services, prioritizing privacy and confidentiality. Leveraging cloud computing has proven to be instrumental in profit maximization for company executives. In the intricate landscape of cloud computing environments, workflow scheduling algorithms play a pivotal role in optimizing the intricate scheduling processes. This research endeavors to introduce the novel Adaptive Group Search Optimization (AGSO) method, aiming to establish its significance in comparison to widely-used algorithms such as Particle Swarm Optimization (PSO), Genetic Algorithms (GA), Bat Algorithm (BA), Firefly Algorithm (FA), Gravitational Search Algorithm (GSA), and Group Search Optimization (GSO) in the dynamic realm of optimizing resource selection for cloud services. AGSO, an innovative approach, seeks to address and alleviate the inherent limitations observed in existing metaheuristics, offering improvements in convergence speed and striking a refined balance in the exploration-exploitation tradeoff.

Keywords: cloud computing; resource provisioning; visualization; group search

ARTICLE INFO

Received: 12 December 2023
Accepted: 10 January 2023
Available online: 28 May 2024

1. Introduction

The virtualization technology employed in cloud computing enables users to access computing resources on-demand. Memory, processors, storage, and applications are all treated as services and shared using the virtualization concept in the cloud. Dynamically provisioning resources in response to user needs is a pivotal aspect of cloud computing, facilitated by the concept of virtual machine migration (VMM). This not only ensures uninterrupted maintenance services but also prevents resource overload within the cloud infrastructure. Virtualization, a key technique in this context, offers various advantages, including power savings and efficient sharing of memory, bandwidth, storage, network, and computing resources. Security, particularly concerning data and network integrity, is paramount in cloud computing, where remote access to data centers is commonplace. Consequently, securing these data centers is a critical element of overall cloud security[1].

Resource allocation, in the context of cloud computing, involves breaking down substantial tasks into smaller components and assigning each subtask to a separate virtual machine for execution.

Task scheduling, on the other hand, encompasses the allocation of distinct subtasks to available resources. In the realm of infrastructure as a service (IaaS), effective resource allocation is integral to efficient resource management. This management not only distributes resources but also optimizes them to maximize energy utilization efficiency. In the landscape of cloud computing, resource management poses challenges such as ensuring quality of service, scalability, enhancing throughput while minimizing overhead and latency, and addressing scalability concerns. Streamlining resource allocation processes is a crucial step to boost productivity and curtail expenses[2]. Superior customer service is required by the Resource supply model. The resource scheduling algorithm aims to lower user task costs relative to completion time. Dynamic resource provisioning allocates varied amounts of resources to jobs based on work volumes. Memory, network bandwidth, and CPU time are among the parameters of a service level agreement, or SL. The auto scaling function intelligently distributes available resources based on user requirements and service level agreements (SLA). The auto scaling method ensures that cloud resources are utilized effectively and efficiently in a hybrid cloud environment. The virtual machines are capable of being dynamically allocated or de-allocated in response to fluctuating traffic demands. Load balancers are of great assistance in the process of building dependable and robust clouds. A technique known as dynamic load balancing distributes scalable workloads evenly across all VMs, also known as virtual machines, in cloud computing. The load balancer is the component that initially determines which virtual machines (VMs) will fulfill a cloud user's request for a specific service[3,4].

2. Literature review

The primary focus of this research revolves around optimizing resource utilization, with a specific emphasis on delay, profit, and cost management. The subsequent sections present an overview of dynamic network deployment in cloud computing[5]. To attain its objectives, the plan employs a combination of heuristic and mathematical grouping strategies, along with a local optimization method. The clustering algorithm utilized is a conventional heuristic clustering algorithm, considering factors such as load balancing, reliability, and cost-effectiveness. Users with shared keywords are grouped together. In contrast, the mathematical grouping technique organizes files mathematically using local optimization.

The heuristic grouping strategy in cloud computing employs the k-means clustering algorithm to meet requirements for resource utilization and energy efficiency. Visualization technology plays a crucial role in achieving these goals through server consolidation. In scenarios where multiple objectives need to be addressed, genetic algorithms frequently employ a grouping strategy. Additionally, a reliability-aware server consolidation strategy is employed to find the optimal solution for visualized data centers on a global scale[6]. This comprehensive approach incorporates various strategies and algorithms to effectively address the multifaceted challenges associated with resource utilization in cloud computing. In cloud computing, the hybrid and gravitational search algorithm achieves superior results for service composition when it comes to reducing costs. The hybridization-based imperialist competitive search algorithm has been found to be useful when providing composite complex services. Gravitational attraction search uses local search to solve cloud service composition[7].

Using a search heuristic, cloud software elements and data centres can be located. Planning requires information routing and network link capacities. MIP optimises network performance, Carbon intensity, capital expenditures, and operational outlay. In this work, the various dimensions of design considerations for feedback control are illustrated. To discover the best candidate architectures, a method based on searching is employed. A flexible design and estimation in the cloud have been provided for architectural analysis[8].

Implementing a parallel and distributed method holds the potential to significantly reduce the time required for executing new instances and the associated costs. The proposed algorithm adopts the branch-and-bound algorithm for effective load balancing across multiple tasks utilizing available processors. In addressing problems of global optimization, various characteristics of binary trees are considered. The visual impact of the branch-and-bound algorithm is diminished when employing the iterative bisection method of the longest edge. This global optimization technique proves valuable in determining the minimum and maximum values of an objective function F, albeit with a computational burden imposed by the longest edge bisection of the regular n-simplex method in a binary tree[9].

Efficient management and distribution of cloud resources are imperative. The mechanism employed incorporates a double combination auction based on a feedback evaluation scheme. Leveraging the visualization concept facilitates the application of the group search optimization algorithm to achieve optimal resource allocation. In decision-making processes, the backpropagation neural network algorithm is utilized following a scientific method. Overcoming the challenge of determining a winner brings cloud computing a step closer to its optimization objectives[10]. This comprehensive approach integrates parallel and distributed methods with optimization algorithms and auction-based mechanisms for efficient cloud resource management. For data management and equitable workload distribution, the data-intensive task employs a map reduce-based methodology. A method known as Block Slicer has been proposed for the purpose of resolving problems associated with entity matching in big data. A new load-balancing strategy was employed for the most efficient implementation of the block slice strategy. A greedy optimization algorithm assigns whole-block and sliced-block match tasks. Using block slicer to distribute blocking-based entity matching schemes in the map reduce framework is recommended. By inserting user data into distributed switching centers, mobile telecom cloud service providers can meet high-quality service requirements at a lower cost. The branch and bound method, which is based on linear programming, is utilized to solve a variety of optimization issues. The similarity-based clustering method is used to organize user membership into classes when managing mobile users in MTC. Data mining frequently employs Meta-heuristic algorithms like GA and PSO. This method determines the interval of association numeric attributes based on provided intervals[11].

Several studies have explored modifications to the Advanced Group Search Optimization (AGSO) algorithm to enhance its performance. One such modification is the Self-Adaptive Elitist Group Search Optimization (SEGSO), introduced to improve the convergence of Group Search Optimization (GSO). SEGSO maintains group heterogeneity through self-adaptive role distribution based on the producer's ConK, ultimately outperforming Particle Swarm Optimization (PSO) and basic GSO in terms of convergence speed and addressing neighborhood problems[12].

Shahid et al.[13] introduced the Focused Grouped GSO (FGGSO) to enhance the exploration rate and strike a balance between intensification and diversification. By analyzing the computational time configuration of the producer's angle searching approach, they developed a campaign approach. The inclusion of competition and rivalry control selection in the searching method serves to prevent exploitation. FGGSO demonstrated superior performance compared to various GSO variants across 11 numerical problems, showcasing significant improvements.

Khan and Algarni[14] improved the GSO algorithm by introducing a Quantum-Behaved Administrator for Scroungers. This strategy involves dividing scroungers into two groups: the first group restructures their relationships with Quantum Particle Swarm Optimization (QPSO) administrators, while the second group continues to access the manufacturer's assets. QPSO admins promote population differences within the GSO, preventing scroungers from getting trapped in the same area if more reliable positions cannot be found. The resulting Improved GSO (IGSO) was compared with other algorithms using a standard benchmark, demonstrating its efficacy and advancements in performance[14]. Rekha and Dakshayini improved GSO to fix the security forces system's error (self-learning). Hydrothermal optimization has identities and constraints. The driven agents in the first GSO paper don't meet the last hydro unit store area restriction. First paper found error. Spatial architecture improves rapid GSO's effectiveness and convergence. Praveen et al.[15] proposed the QGSO algorithm for semirigid iron structures. QGSO restricts sentence coefficients and studies history. Two-bay, five-layer steel QGSO seismic design was evaluated. QGSO outperformed PSO in convergence rate and ability to escape local optimality. QGSO optimises steel seismic structures. S Velliangiri et al.[16] proposed a forecast prototype using fuzzy interval series and an improved GSO. Based on the fuzzy logical ratio, the fuzzy time series accurately predicts whether future data will increase or decrease. Enhanced GSO modifies time. It predicts UA enrollment. The prototype's forecast error was the lowest.

Several studies solved optimization problems with hybrid GSO. Sadeeq[17] combined GSO and the metropolis rule to avoid local optimality. The proposed algorithm outperforms GSO and other algorithms for high-dimensional multi-model optimization problems. Sadeeq et al.[18] proposed a dynamic economic dispatch that considers valve point impacts. The proposed show is a non-convex optimization problem illuminated by GSO with various techniques[19]. The proposed method expands GEO investigation and abuse capabilities using a flexible covariance network and an efficient transformation operation. Two test cases and a comparison to other heuristic algorithms evaluate the proposed strategy. Plan addresses financial and energy speed. OBL and DE create a hybrid GSO to solve optimization problems. The OBL method broadens local search while the DE administrator improves it. In merging speed and arrangement precision, the proposed crossover GSO algorithm beat previous GSO and Goyal et al.[20] developed a turbo encoder that uses a hybrid GA/GSO (HGSO). High-dimensional data became low-dimensional by interleaving a new high dimension. Meta-heuristic seek led to statistics transformation and pattern technology. The proposed system outperforms ABC, GSO, FA, ABC, GA and random interleaver designs in SNR, BER, and FER. The proposed HGSO excessive dimension interleaved study outperformed others.

1) Proposed optimization strategy for the advanced group search

Load balancers can multitask thanks to group search optimization. Following is an explanation of the algorithm for the group search optimization scheme[21].

2) GSA stands for the Group Search Optimization Algorithm

Efficient resource scheduling is crucial for high-performance cloud workflows enclosed in **Figure 1**. The Ant Colony System (ACS) scheduling method proves effective in optimizing resource utilization, particularly in the context of deadline-based resource scheduling. This workflow scheduling model is designed to align with business needs, offering a strategic approach to reduce costs. A dynamic objective strategy, implemented through a genetic algorithm, is employed for deadline-constrained resource scheduling, optimizing both execution time and cost. Resource scheduling, addressing time constraints, plays a pivotal role in achieving these optimization goals[22].

In the domain of cloud computing's scientific workflow processes, a scheduling algorithm is deployed to handle variations in resource performance and virtual machines (VMs). The knapsack-based algorithm efficiently addresses user-defined deadlines, leveraging the dynamic capabilities of the cloud infrastructure to concurrently reduce costs. The application of dynamic programming further enhances the quality of schedules generated for scientific workflows explained in the Algorithm 1. Notably, the method of resource provisioning is extensively used to curtail expenses associated with executing user requirements[23,24]. This comprehensive approach underscores the importance of strategic scheduling and resource management in achieving high-performance and cost-effective cloud workflows. AGSO variants, applications, and results are studied. This article describes works that use AGSO to optimize benchmark functions, machine learning, engineering, and networks expressed in **Table 1**. AGSO uses scheduling, clustering, pattern recognition, and unconstrained optimization. Data shows AGSO solves all problems. Hybridization, modification, and other variants must boost AGSO's performance. This survey paper describes advancements, applications, advantages, and disadvantages.

Figure 1. Process of advanced group search methodology.

Algorithm 1 PSO Algorithm

1: Step 1: Start the process
2: Step 2: Enter multiple requests from users into the cloud.
3: Step 3: The initial solution generator will produce a variety of tasks based on the requests that have been given.
4: Step 4: Assign the first iteration to the fitness function generator based on the subtask that was generated.
5: Step 5: Produce a result based on output from previously generated iterations
6: Step 6: Modify the solution and velocity tables in accordance with the generated fitness value
7: Step 7: Take the previous solutions and combine them to come up with the most effective answer to the problem.
8: Step 8: Delegate the responsibility to the iteration that produced the best results most recently.
9: Step 9: Determine the best solution that will take the least amount of time and money to implement.
10: Step 10: End of the process

Table 1. Types of scheduling algorithms.

Scheduling algorithms	**Methods**	**Factors**	**Parameters**
PSO Algorithm	Various responsibilities	Factor of dependence	Time and resource utilization
Market-driven hierarchical planning	Clustering the data virtually	Planning at the service level and task level	Build-up, Effort, and Expense
An algorithm that bases scheduling decisions on costs	Batch processing	Unscheduled meeting of the task group	Expense, in relation to output
Resource scheduling that is more efficient	Batch mode is the most efficient way to work	Unscheduled meeting of the task group	Quickness as well as efficient use of resources
Scheduling with multiple QoS constraints	Batch and Dependency modes are both available	Various work flows	The amount of money spent, the amount of time spent, and how long the project will take are all important considerations

Table 1. (*Continued*).

Scheduling algorithms	Methods	Factors	Parameters
Job scheduling based on priority	Mode of reliance	Queue of work	prioritization queue
Scheduling with consideration for resources	Processing done in batches	Request a time slot	The rate at which resources are used

3. Comparative analysis

Table format is used to present the various workflow scheduling algorithms that are currently available. These algorithms are evaluated in the cloud environment using a wide variety of parameters, factors, and scheduling methods. For the process of optimization to work properly with a variety of cloud management tools, the scheduling algorithms need to be able to effectively take care of reliability, availability, cost, throughput, and time.

4. Results

The proposed method has been examined in light of the mandatory safety precautions. The data security has been focused on the preservation of the users' privacy when accessing confidential data files stored in cloud environments. The proposed plan's security is more effective than the algorithms that are currently in use. The analysis of the results demonstrates that the proposed work performs more quickly than other approaches. A representation of the file partition system is shown in **Figure 2**. The figure presents an illustration of the essential optimization procedure. For reasons related to cryptography, the plaintext was changed into the cipher text. During the process of uploading a new data file into a cloud environment, multiple cryptographic keys are generated. In the AGSO scheme, the selection of a cryptographic key is determined by choosing the key that provides the best optimization among a set of available keys. The adoption of analytics as a service has significantly reduced the cost of cloud resources. Cloud resource scheduling, specifically tailored for big data analytics, plays a crucial role in ensuring Quality of Service (QoS). To meet Service Level Agreement (SLA) requirements in cloud computing applications, the AGSO algorithm incorporates concepts of admission control and resource scheduling. This section conducts a comparative analysis of AGSO against other optimization algorithms, namely PSO, GA, BA, FA, and GSA, utilizing benchmark test functions with different characteristics. To facilitate a fair comparison, benchmark results are normalized between 0–1, and the Friedman test is employed to evaluate the results. AGSO exhibits commendable performance, initially outperforming three unimodal benchmarks. The algorithm's application to unimodal test functions demonstrates increased exploitation search and convergence. AGSO surpasses all multimodal optimizers, showcasing its ability to avoid local maxima through exploration. In the validation on composite benchmark functions, AGSO consistently demonstrates superiority in optimizing large-search-space problems. Furthermore, the performance of the parallel AGSO algorithm is normalized, and the last row of the table provides averages. AGSO's average value outperforms its competitors, positioning it favorably.

Figure 2. Arrangement for the partitioning of files.

5. Conclusion

The imperative to safeguard sensitive data without compromise necessitates robust protection mechanisms from cloud service providers. Cryptographic techniques emerge as pivotal tools in ensuring the security of cloud data storage. The deployment of dynamic load balancers within the cloud infrastructure serves as a critical mechanism for the efficient distribution of scalable workloads across all nodes of the host machine, facilitating optimal resource utilization explained in **Table 2**.

Table 2. The typical solutions obtained when benchmark functions are solved.

	PSO	GA	BA	FA	GSA	GSO	AGSO
S1	0.9116	0.9915	0.2652	0.3170	0.4907	0.4735	0.0712
S2	0.7166	0.7741	0.9832	0.9048	0.4678	0.0660	0.0102
S3	0.1531	0.2007	0.9201	0.8527	0.0984	0.1637	0.0452
S4	0.3071	0.7806	0.3591	0.7565	0.2225	0.7103	0.0200
S5	0.8605	0.4333	0.9842	0.5585	0.5466	0.1780	0.0568
S6	0.3867	0.6192	0.4401	0.3247	0.8617	0.5267	0.0116
S7	0.9820	0.3614	0.4785	0.5760	0.9188	0.6463	0.0905
S8	0.9795	0.9889	0.9778	0.7972	0.5179	0.2049	0.0810
S9	0.6260	0.3133	0.0476	0.8628	0.1958	0.6050	0.0745
S10	0.4660	0.8431	0.0977	0.5719	0.1404	0.5199	0.0607
S11	0.9347	0.3261	0.5224	0.1109	0.0123	0.3396	0.0330
S12	0.1297	0.1438	0.7995	0.6383	0.1929	0.0496	0.0289
S13	0.1996	0.5022	0.7350	0.0710	0.8443	0.0852	0.0675
S14	0.6448	0.8839	0.8034	0.3146	0.0334	0.3632	0.0414
S15	0.6448	0.7869	0.5061	0.8754	0.3249	0.4579	0.0969

The growing demand for resources in cloud data centers amplifies the urgency of devising effective resource selection strategies implemented by customers elaborated in the **Figure 3**. While this demand intensifies, addressing certain limitations associated with metaheuristics, especially the proposed AGSO method, can catapult their effectiveness in service selection to unprecedented heights. The cloud computing platform, committed to delivering the utmost in user satisfaction, steadfastly preserves user privacy and confidentiality, resulting in tangible profit maximization for company executives.

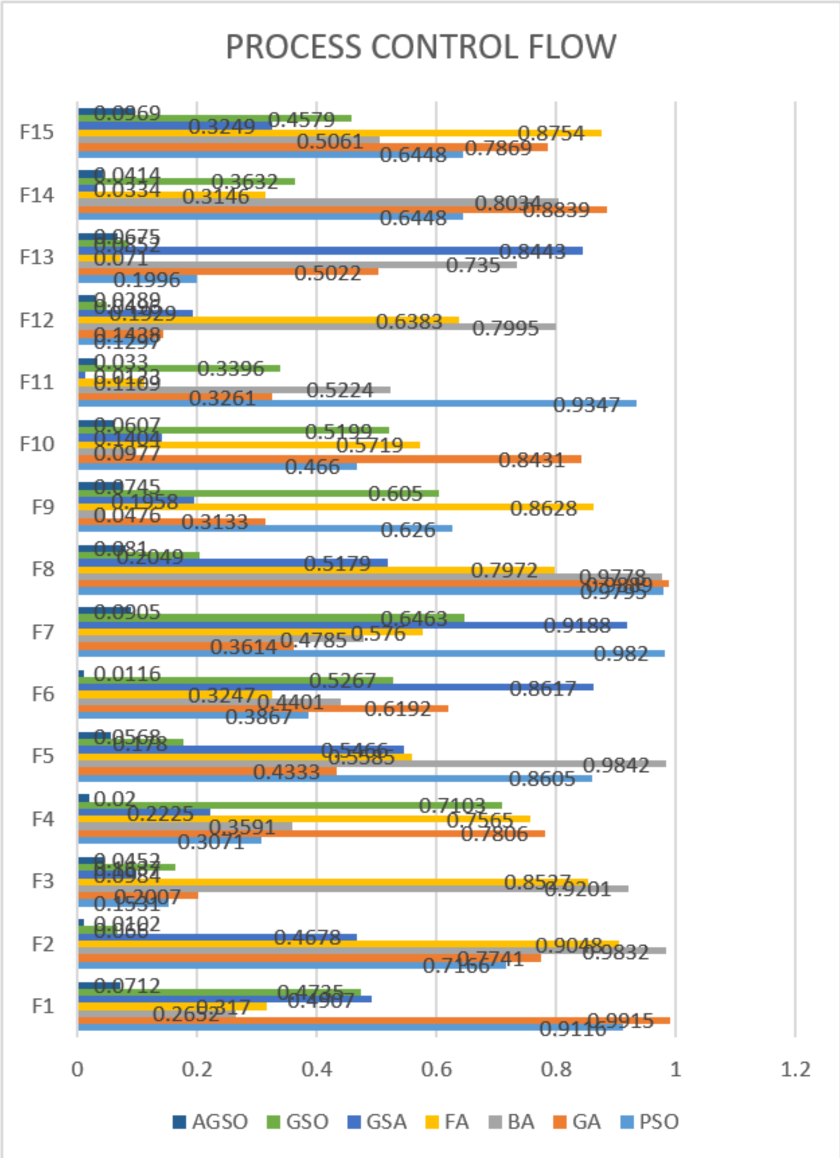

Figure 3. Comparitive Study of AGSO, GSO, GSA, FA, BA, GA, PSO.

In the vast and intricate landscape of cloud computing, AGSO stands out as a promising solution, showcasing superior capabilities when juxtaposed with traditional metaheuristics like PSO, GA, BA, FA, GSA, and GSO. The significance of AGSO in optimizing workflow scheduling within cloud environments marks a paradigm shift, heralding a new era of efficiency and effectiveness in the ever-evolving field of cloud computing.

Conflict of interest

The author declares no conflict of interest.

References

1. Yahia HS, Zeebaree SRM, Sadeeq MAM, et al. Comprehensive Survey for Cloud Computing Based Nature-Inspired Algorithms Optimization Scheduling. Asian Journal of Research in Computer Science. Published online May 3, 2021: 1-16. doi: 10.9734/ajrcos/2021/v8i230195
2. Mohammadi A, Rezvani MH. A novel optimized approach for resource reservation in cloud computing using

producer–consumer theory of microeconomics. The Journal of Supercomputing. 2019, 75(11): 7391-7425. doi: 10.1007/s11227-019-02951-1
3. Chen X, Cheng L, Liu C, et al. A WOA-Based Optimization Approach for Task Scheduling in Cloud Computing Systems. IEEE Systems Journal. 2020, 14(3): 3117-3128. doi: 10.1109/jsyst.2019.2960088
4. Attiya I, Abd Elaziz M, Xiong S. Job Scheduling in Cloud Computing Using a Modified Harris Hawks Optimization and Simulated Annealing Algorithm. Computational Intelligence and Neuroscience. 2020, 2020: 1-17. doi: 10.1155/2020/3504642
5. Torabi S, Safi-Esfahani F. A dynamic task scheduling framework based on chicken swarm and improved raven roosting optimization methods in cloud computing. The Journal of Supercomputing. 2018, 74(6): 2581-2626. doi: 10.1007/s11227-018-2291-z
6. Masadeh RMT, Sharieh AAA, Mahafzah BA. Humpback whale optimization algorithm based on vocal behavior for task scheduling in cloud computing. International Journal of Advanced Science and Technology. 2019, 13(3): 121-140.
7. Naseri A, Jafari Navimipour N. A new agent-based method for QoS-aware cloud service composition using particle swarm optimization algorithm. Journal of Ambient Intelligence and Humanized Computing. 2018, 10(5): 1851-1864. doi: 10.1007/s12652-018-0773-8
8. Prasanna Kumar KR, Kousalya K. Amelioration of task scheduling in cloud computing using crow search algorithm. Neural Computing and Applications. 2019, 32(10): 5901-5907. doi: 10.1007/s00521-019-04067-2
9. Shukur H, Zeebaree S, Zebari R, et al. Cloud Computing Virtualization of Resources Allocation for Distributed Systems. Journal of Applied Science and Technology Trends. 2020, 1(3): 98-105. doi: 10.38094/jastt1331
10. Pandey BK, Pandey D, Sahani SK. Autopilot control unmanned aerial vehicle system for sewage defect detection using deep learning. Engineering Reports.2024. p.e12852.
11. Ebadifard F, Babamir SM. A PSO-based task scheduling algorithm improved using a load-balancing technique for the cloud computing environment. Concurrency and Computation: Practice and Experience. 2017, 30(12). doi: 10.1002/cpe.4368
12. Ibrahim IM. Task Scheduling Algorithms in Cloud Computing: A Review. Turkish Journal of Computer and Mathematics Education (TURCOMAT). 2021, 12(4): 1041-1053. doi: 10.17762/turcomat.v12i4.612
13. Shahid MH, Hameed AR, ul Islam S, et al. Energy and delay efficient fog computing using caching mechanism. Computer Communications. 2020, 154: 534-541. doi: 10.1016/j.comcom.2020.03.001
14. Khan MA, Algarni F. A Healthcare Monitoring System for the Diagnosis of Heart Disease in the IoMT Cloud Environment Using MSSO-ANFIS. IEEE Access. 2020, 8: 122259-122269. doi: 10.1109/access.2020.3006424
15. Rekha PM, Dakshayini M. Efficient task allocation approach using genetic algorithm for cloud environment. Cluster Computing. 2019, 22(4): 1241-1251. doi: 10.1007/s10586-019-02909-1
16. Praveen SP, Rao KT, Janakiramaiah B. Effective Allocation of Resources and Task Scheduling in Cloud Environment using Social Group Optimization. Arabian Journal for Science and Engineering. 2017, 43(8): 4265-4272. doi: 10.1007/s13369-017-2926-z
17. Velliangiri S, Karthikeyan P, Arul Xavier VM, et al. Hybrid electro search with genetic algorithm for task scheduling in cloud computing. Ain Shams Engineering Journal. 2021, 12(1): 631-639. doi: 10.1016/j.asej.2020.07.003
18. Mohammed Sadeeq M, Abdulkareem NM, Zeebaree SRM, et al. IoT and Cloud Computing Issues, Challenges and Opportunities: A Review. Qubahan Academic Journal. 2021, 1(2): 1-7. doi: 10.48161/qaj.v1n2a36
19. Arunarani AR, Manjula D, Sugumaran V. Task scheduling techniques in cloud computing: A literature survey. Future Generation Computer Systems. 2019, 91: 407-415. doi: 10.1016/j.future.2018.09.014
20. Goyal S, Bhushan S, Kumar Y, et al. An Optimized Framework for Energy-Resource Allocation in a Cloud Environment based on the Whale Optimization Algorithm. Sensors. 2021, 21(5): 1583. doi: 10.3390/s21051583
21. Jena UK, Das PK, Kabat MR. Hybridization of meta-heuristic algorithm for load balancing in cloud computing environment. Journal of King Saud University-Computer and Information Sciences. 2022, 34(6): 2332-2342. doi: 10.1016/j.jksuci.2020.01.012
22. Abualigah L. Group search optimizer: A nature-inspired meta-heuristic optimization algorithm with its results, variants, and applications. Neural Computing and Applications. 2020, 33(7): 2949-2972. doi: 10.1007/s00521-020-05107-y
23. Tanaka S, Wang Z, Dehghani K, et al. Large Scale Field Development Optimization Using High Performance Parallel Simulation and Cloud Computing Technology. Published online September 24, 2018. doi: 10.2118/191728-ms
24. Zhu Y, Zhang W, Chen Y, et al. A novel approach to workload prediction using attention-based LSTM encoder-decoder network in cloud environment. EURASIP Journal on Wireless Communications and Networking. 2019, 2019(1). doi: 10.1186/s13638-019-1605-z