Indonesian Journal of Electrical Engineering and Computer Science
Vol. 35, No. 2, August 2024, pp. 1131~1138
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v35.i2.pp1131-1138

DNA computing and meta-heuristic-based algorithm for big data task scheduling in cloud computing

Visalaxi Gandhimathinathan[1], Muthukumaravel Alagesan[2]
[1]Department of Computer Science and Engineering, Bharath Institute of Higher Education and Research, Chennai, India
[2]Department of Arts and Science, Bharath Institute of Higher Education and Research, Chennai, India

Article Info

Article history:

Received Jan 20, 2024
Revised Mar 6, 2024
Accepted Apr 6, 2024

Keywords:

Bat sonar algorithm
Big data task scheduling
Cloud computing
DNA computing
Virtual machine

ABSTRACT

With the advent of cloud computing, there is a need to enhance both the methods and algorithms of big data workloads for task scheduling. Due to the global spread of services with changing task load circumstances and different cloud client demands, big data task scheduling in cloud systems is a time-consuming process. The proposed approach emphasises the necessity for efficient big data task scheduling in the cloud computing, which exacerbate data processing. Virtual machines frequently utilise all three types of physical resources: CPU, memory, and storage. Big data task scheduling is one of the most important implications of cloud computing application resource management, and this research work meticulously offers a task scheduling technique for advancing cloud computing.

Corresponding Author:

Visalaxi Gandhimathinathan
Department of Computer Science and Engineering, Bharath Institute of Higher Education and Research 600073, Chennai, India
Email: visalaxikumar@gmail.com

1. INTRODUCTION

Big data has been is predominately coupled with services of cloud computing for data-intensive computing using distributed frameworks such as MapReduce, Dryad and Hadoop. There will be a potential issue owing to huge big data task scheduling in cloud data centres. With effective dynamic task management, the overall performance of the cloud computing paradigm could be enhanced. The cloud ecosystem is a platform that may be used to host cloud-based application-oriented services. In recent years, academics have been interested in big data task scheduling in a cloud computing context [1]. A virtual cluster provides a flexible environment that may scale up and down in response to changes in user computing needs. To reduce energy usage, cloud services merge virtual clusters of various users into a real data centre. Due to data location and task activities, the demand for computing resources for each node in a virtual cluster may fluctuate [2]-[5]. Existing cloud solutions, on the other hand, have a static cluster setup, requiring each deteriorating virtual machine to be manually changed. Virtualization takes place on cloud-distributed networks, and a virtual cluster is thus created due to a wide set of virtual machines. Unlike the conventional physical cluster, the data-intensive platform operates on a virtual cluster. MapReduce has transformed parallel and distributed big data analysis in the last few years. As illustrated in Figure 1, MapReduce is a parallel and distributed computing framework that has evolved as the de facto standard for processing a wide range of unstructured data on commodity hardware clusters [6].

Sifting through device logs, conducting extract-transform load operations, and generating web indexes have all proven to be efficient batch applications [7]. Access to big data hardware resources is a key barrier to medium scale companies since big data analytics includes distributed computing on a scale that generally requires hundreds to thousands of machines. In recent technology advancement improves reliability and ushers in a new era of increasing data with advent of internet of things (IoT) that integrates numerous sensors with embedded networks to allow data to communicate and real-time data are stored in the cloud for subsequent analysis [8], [9].

An examination of different factors, including resource utilization, overall cost for executing user tasks, time taken for completion, power consumption, and the ability to handle faults during the task scheduling phase, revealed their crucial relevance within the realm of cloud computing. Task scheduling stands out as a pivotal aspect of managing resources in cloud computing applications. This study diligently presents a technique aimed at enhancing cloud computing through effective task scheduling methodologies.

The primary goal of cloud computing is to allow customers to organise their operations in the most effective way possible, resulting in load balancing and the best possible service quality. To develop increasingly efficient solutions to the issue, evolutionary computation uses biological principles like as populations, mutation, and survival of the fittest [10]-[12]. They are divided into two categories: evolutionary algorithms and swarm intelligence algorithms like ant and bat algorithms. In general, the evolutionary method is comprised of the agent and its environment. The agents respond in response to their surroundings. Agents use sensors to act, whereas the environment uses effectors.

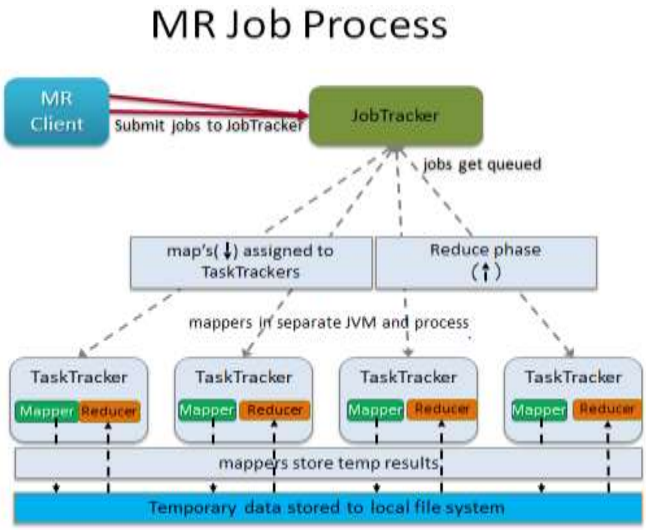

Figure 1. MapReduce

The architecture and agent programme make up an agent's structure. The architecture refers to the system on which the agent runs, while the agent programme refers to the implementation of the agent function. The term "mundane task" refers to tasks that do not require any special training, such as reasoning. It's also seen a lot in natural language processing. Formal logic and a few reasoning tasks, such as classification, in genetic algorithm instead of getting a single solution a group of potential solutions is obtained in parallel. In the population each individual is a unique solution. To produce offspring parents are subjected to reproduction. For new generation to create population 20% chromosomes which are best fit copied directly. Crossover and Mutation operators are used to produce remaining 80% chromosomes. This paper contributions as follows:

- Outline the problem of big data task scheduling in cloud computing environments.
- Identify unsolved problems and areas requiring improvement, particularly those addressed in this manuscript.
- Introduce new contributions, including the proposed technique for enhancing cloud computing through dynamic task scheduling.
- Demonstrate the relevance of these contributions through subsequent sections outlining the methodology, results, and discussion.

The upcoming sections are as follows: section 2 deliberates the methodology and explains the system operations. Section 3 emphasizes the proposed methodology and its contributions, section 4 reveals results and discussion.

2. MATERIALS AND METHODS

2.1. DNA based computing

Deoxyribonucleic acid (DNA) is a biomolecular computing branch, which employs DNA data instead of standard computer technology based on silicon. The concept of using individual living molecules for computing dates back to 1959, when American physicist Richard Feynman introduced his nanotechnology concepts. DNA was not physically carried out until 1994, however, when Leonard Adleman from American computer scientists revealed how molecules might be utilised to solve computer problems. The technique based on DNA computing is an emerging scientific topic that includes mathematics, computer science, and molecular biology as a viable approach to solving the computationally intractable issue. To isolate DNA from biological material, a number of DNA extraction techniques are available or have been employed. The generally tiny amount of appropriate beginning DNA is a highly essential and limiting element in processing molecular studies. DNA computation is characterised by a broad parallelization provided by DNA strands, which implies that the data structure with considerably more parallelization than neural networks and genetic algorithm is the initial power of DNA computation, for instance, DNA computing has found its footprints in advance logic gates presented in Figure 2. The complement between the two strands of DNA provides the second power of DNA computing.

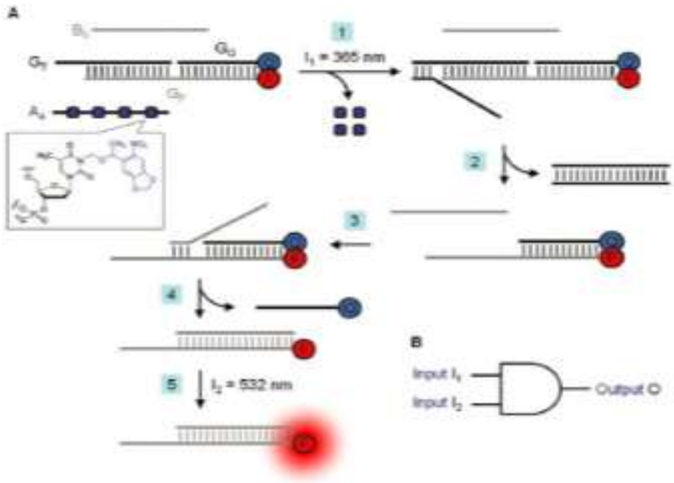

Figure 2. An example of DNA based solution for effective logic operations

Adenine (A), guanine (G), cytosine (C), and thymine (T) are the four nucleotides that make up DNA, according to DNA structure (T). They lied on the basis of a sugar-phosphate backbone. Watson=crick (W=C) complementary chemical bonds exist between two distinct DNA single strands. Using the DNA bases T, G, C, and A, DNA computing can be one of the greatest ways to optimise job scheduling (A). DNA computing is a kind of parallel calculation which may be carried out simultaneously utilising many distinct DNA molecules. DNA computing is based on the concept that molecular biological processes may utilise the information contained as DNA strands in an arithmetical and logic way.

2.2. Optimization using meta-heuristic algorithm

For many years, humans have been deeply intrigued by the way bats, scientifically known as Chiroptera, live their lives. Comprising over 900 species globally, bats represent about a quarter of all mammalian species, making them one of the most diverse and remarkable orders among mammals. Each bat species possesses unique characteristics and abilities, setting them apart within the animal kingdom. They are categorized into two suborders based on size: megachiroptera and microchiroptera.

Bats typically inhabit shared roosts, forming large colonies housing anywhere from 700 to 1,000 individuals. These roosts are often found in vertically positioned spaces like caves or the abandoned roofs of buildings, characterized by a horizontal ceiling measuring 0.75 to 1 inch wide and 16 to 24 inches deep. Around sunset, when the surroundings begin to darken, bats depart from their roosts, relying on their spatial memory to ensure synchronized departure among colony members.

Communication within bat colonies involves two primary types of auditory signals. Social sounds facilitate interaction and communication among bats, while echolocation noises aid in foraging and navigation. Within these colonies, four key mechanisms enable information sharing:

i) Intentional signaling: bats use various intentional signals such as mating calls, territorial calls, warning signals, and food calls. These serve the purpose of advertising food availability and attracting bats into foraging groups as they leave their cave roosts.
ii) Local enhancement: this unintentionally guides another bat to a specific location within the environment.
iii) Social facilitation: group foraging behavior within the colony results in increased individual foraging success.
iv) Bats may adopt foraging strategies from fellow colony members through imitation, learning from one another's behavior.

The bat method has been used to solve a variety of optimization issues in the field of engineering. Furthermore, some academics have attempted to enhance performance by combining the bat method with another algorithm, and others have attempted to change the original version of BA. For example, BA has been used to tackle difficulties arising from non-linearity. To find food, bats use the echolocation technique. The bats must use sonar to determine the target's elevation, size, velocity, and distance in order to locate food. The suggested technique offered an excellent optimal solution when compared to the other existing methods. Evolved bat algorithm (EBA) seems to be a modification of the original bat algorithm [10] and the author has studied and characterised the behaviour of the entire bat species. While dealing with the numerical optimization issue, this approach greatly reduced calculation time by improving accuracy. The authors developed a new method for using the bat algorithm in multi-objective optimization problems by welded beam design's multi objective used the method of the multi objective bat algorithm (MOBA), which works better with this algorithm [12], [13]. The bat method was used to solve the restricted optimization issue. When compared to other current algorithms, the optimum solution using bat algorithm gives a better result. The efficiency of the bat algorithm is improved by integrating chaotic sequence and chaotic Levy flight schemes into the current method. This technique strikes a nice balance between variety and intensity, which may help to enhance search behaviour. The authors proved that this technique was reliable with the combined estimate of parameter vector for the building of a dynamic biological system [14].

3. PROPOSED METHOD

The primary objective of task scheduling algorithms is to successfully attain these critical objectives, namely, task scheduling enables physical system make-up and resources to be minimized. For successful resource allocation [15]-[19], it is important to enhance both the procedures and algorithms of cloud workloads since the introduction of cloud computing. Since big data analytics includes distributed computing on a scale that generally requires hundreds to thousands of machines. Big data task scheduling is considered an important component of cloud computing since it has a direct influence on cloud computing performance. If a simpler and more dependable resource distribution system could be developed, consumer demands would be satisfied, and the provider's usage and profit would be maximised [20].

3.1. Architecture of the proposed work

An examination of different factors, including resource utilization, overall cost for executing user tasks, time taken for completion, power consumption, and the ability to handle faults during the task scheduling phase, revealed their crucial relevance within the realm of cloud computing. Task scheduling stands out as a pivotal aspect of managing resources in cloud computing applications. This study diligently presents a technique aimed at enhancing cloud computing through effective task scheduling methodologies. Cloud Workload architecture involves a crucial aspect: detecting anomalies while being aware of workloads. This process relies on outliers, deviations, and variations to unveil patterns that might affect the typical activity of a workload. Any abnormal behavior within workload patterns is termed a deviation. Detecting these irregularities in workload data sources is valuable for cloud service providers and clients as it aids in crafting customized resource allocation strategies.

In a multi-sonar unit, resembling a cluster of bats, the optimum solution is collectively determined. Each sonar unit, with its unique starting point in the associated search space, contributes to the simultaneous decision-making process. Contrastingly, in a single sonar unit, the integration of transmitted beam lengths with momentum is pivotal [21].

The proposed technique allocates m processors to j jobs, each comprising n tasks. Crucially, no task should be processed on more than one processor simultaneously, necessitating a meticulous scheduling approach [22]. The finish time, denoting the latest completion time scheduled by each processor, serves as a benchmark. The primary scheduling objective revolves around reducing completion times.

For core-based systems, establishing equivalence between test scheduling involves treating the cores as tasks. In the realm of DNA sequencing, determining the sequence for the DNA molecule-comprising A, G, C, and T is a critical process. The Sanger sequencing method involves denaturing the DNA into singlestranded form and dividing it into four processes. These processes require DNA polymerase and four distinct deoxynucleotides (dNTPs), specifically dATP, dGTP, dCTP, and dTTP. Each reaction incorporates one of the four distinct DDNTPs, like ddATP, ddGTP, and ddCTP [23]-[25].

Utilizing a binary type, the DNA sequence is encoded into a binary scheme. The DNA bases A, T, C, and G correspond to the binary codes 00, 11, 01, and 10 respectively. For instance, the "AATCGGAT" DNA sequence translates to the binary sequence 000011010100011. Notably, RNA replaces T with uracil (U). In the context of Hadoop, there's a rising need for dependable facilities ensuring fixed completion times for each task. However, meeting these deadlines within the existing Hadoop architecture poses challenges. Varying resource requirements for each task make determining necessary resources complex. Additionally, the scheduling order of jobs within Hadoop clusters significantly impacts task completion times.

Within the solution spectrum (SSsize) bounded by the upper (SSMax) and lower (SSMin) limits of the search space, a random value reflects the beam length (L).

$$SS_{size} = SS_{Max} - SS_{Min} \tag{1}$$

By keeping the constant value L:

$$L \leq Rand \times \left(\frac{SS_{size}}{10\% \times bats}\right) \tag{2}$$

integrated with a momentum (μ):

$$L_{new} = L_{old}(1 \pm \mu) \text{ where, } 0 < \mu < 1 \tag{3}$$

the problem can be overcome by limiting the horizontal axis among the beams:

$$\theta_i = \frac{(2\pi - \theta_m)}{NBeam} \tag{4}$$

the point position for transmitted beam obtained as:

$$pos_i = pos_{SP} + Lcos[\theta_m + (i-1)\theta] \tag{5}$$

where, i = 1, …, NBeams.

Due to the lack of minimal resources, modern technologies also alter optimization techniques. In addition, time is also another important parameter that could cause several applications to operate in real time. As a consequence, the capital allocation can be conveniently achieved with the reduced wait period. This leads to the adaptation of optimization methods in order to determine the best means of efficiently handling and delivering services with preconceived constraints in order to satisfy the demands of customers and to respond to the goals of the suppliers. In the other hand, with the expected optimization approach, which takes the same steps in each iteration process and yields the same result, certain multi-dimensional problems could not be resolved. This kind of multi-dimensional problem is part of a heuristic approach that could be dealt with effective fitness function. The iteration number is less than and equal to 100, as there are multiple steps in the adaptive bat sonar algorithm, which leads to enough bats to investigate in a search space dimension to decide the best global fitness value. Despite the fact that the adaptive bat sonar method has several stages, the number of iterations is fewer than or equal to 100, which is the same as the original bat sonar algorithm, resulting in enough bats to explore in a search space dimension to determine the global best fitness value. In general, the adaptive bat sonar method had a population of less than 1,000 bats. A huge number of bats will be available, and a pool of solutions may be readily identified in a speedier manner, in order to achieve the worldwide best fitness value in an effective manner. The random value represents beam length (L) in the solution range (SSsize) among the upper (SSMax) and lower search space (SSMin). The proposed approach improves the virtual machines' local disc by converting it into a globally shared data storage. As a result, applications can use the virtual machine's local disc instance, which can interchange input files and record intermediate data. This approach may be expanded to include effective data processing for general work-flow to improve data placement for file transfers across nodes.

4. IMPLEMENTATION AND RESULTS

The bat sonar method was developed by examining a connected network with numerous nodes. The bat is programmed to walk on the arcs starting from its start node and travelling to nearby nodes with each step until it finds a food supply node. The bat can be programmed to "read" and "write" to produce its self-pheromone on the arcs. In-order to evaluate the efficiency of the algorithm, the following fitness functions were examined by applying the algorithm:

i) A maximum value by variable-third order polynomial.
ii) A maximum value by variable-fifth order polynomial.
iii) A maximum value by polynomial with more than one variable.
iv) A maximum value by exponential with more than one variable.
v) A maximum value by trigonometric as well as periodic function with constant function in regular intervals.

All tests were included with initial parameters where N=5, Fixed $\theta=\pi/12$ as well as maximum iterations of 100. The best solution is determined by the degree of performance by the obtained the pool of solutions, which rates the efficiency of the algorithm. The algorithm was analysed and compared with genetic algorithm with the same fitness function. The results of the comparison were based on the fitness function as well as the execution time needed to accomplish every function. The obtained results reveal that algorithm performs well to accomplish best solution (values). Several problems in the bat algorithm were discovered during the adaption of the algorithm into the experimental research. The following are the challenges:

– When locating the target, the number of bats used in the algorithm is minimal and does not reflect the normal population size of a bat colony.
– During the search for the optimal fitness value in exploration and exploitation, the population in the search space should be substantial.
– The beams' transmission is static in terms of direction

These difficulties result in late convergence, resulting in lower effective accuracy, although in the optimal solution, better precision and quicker processing are achieved. However, because actual bats echolocation has numerous aspects that might be added, the algorithm does not accurately reflect the genuine behaviour of bats echolocation in nature.

As presented in Figure 3, the clusters are created using Apache Hadoop, the total number of processors on the cluster is presented by MaxProcs in the log. This does not represent the nodes in the log that the workers were able to reach and use. Furthermore, the record only includes jobs sent through the grid's resource brokers and excludes loads created locally. The data vectors are randomly spread over a 2D grid. Agents, or bats, are placed in a 2D grid at random. The bat visits each neighbourhood throughout each iteration step and decides whether or not to pick. The algorithm is terminated as soon as the ideal solution is found and the termination condition is met; otherwise, the procedure begins and continues. The data centres will provide a set of tasks that would be started and performed on the hosts, depending on the virtual machine type. This solution includes both static and dynamic type, depending on the current machine condition. The main parameters are virtual machine (VM) cost, network bandwidth, VM sort, random access memory (RAM) capacity, penalty, datacenter load, disc size, and CPU processing speed.

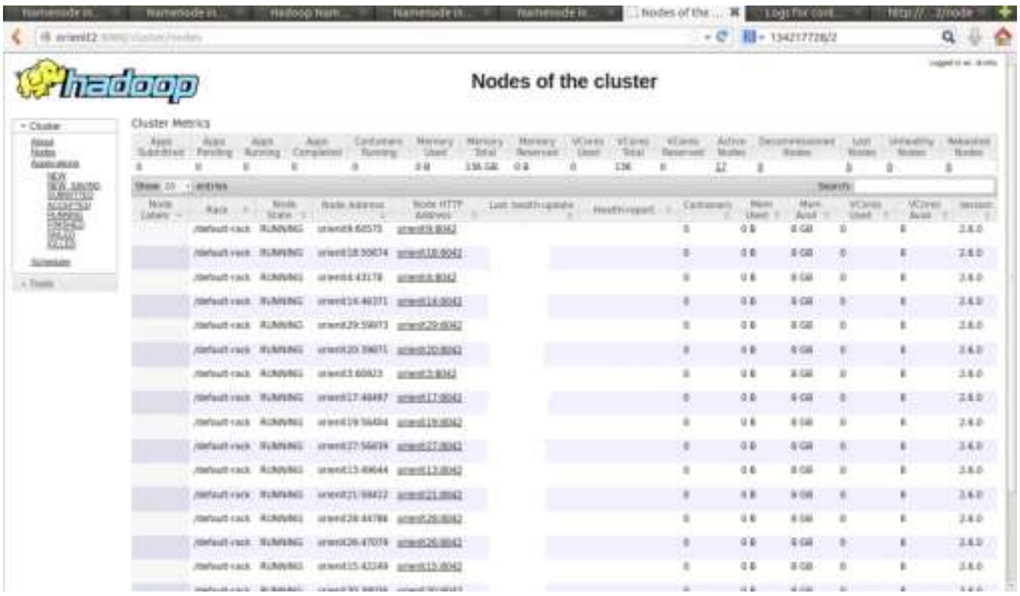

Figure 3. Clusters with node address

There is no general method for deciding their optimum combination when choosing parameters. As shown in Figures 4 and 5, the experiment relies on the aggregated throughput for big data task scheduling with higher concurrency. The graph in Figure 5 illustrates the relationship between the fitness value and the number of generations. This relationship demonstrates how the fitness of the population evolves over successive generations during the optimization process. Specifically, it highlights the trend of fitness improvement or degradation over time as the algorithm iterates through generations. This graphical representation serves to provide insight into the convergence behavior and effectiveness of the optimization algorithm employed in the study.

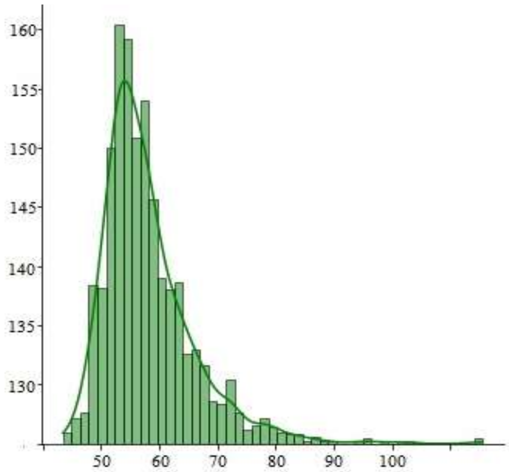

Figure 4. Performance of algorithm obtaining higher throughput

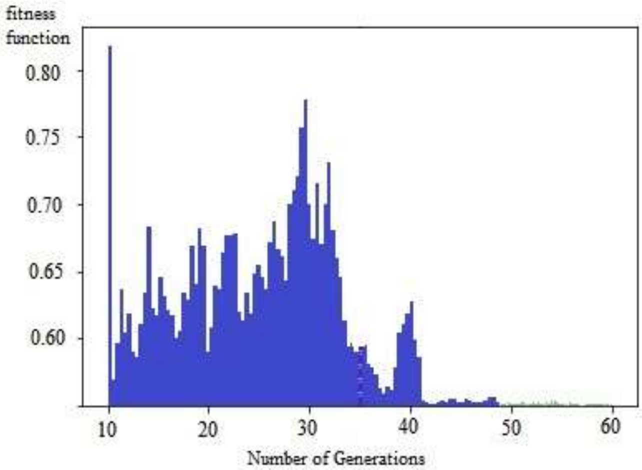

Figure 5. Fitness function result

5. CONCLUSION

The major goal of cloud computing is to provide cloud customers with the best possible big data task scheduling with the least amount of downtime, as well as load balancing with increased concurrency. The primary purpose of big data task scheduling algorithms is to efficiently fulfil these goals. Task scheduling is a crucial element in improving the overall efficiency of complicated resource allocation strategies, therefore the proposed approach tackles a wide range of task scheduling difficulties. For control and administration in the cloud computing environment, bat sonar algorithm is utilised. To tackle this resource allocation challenge, The bat algorithm and DNA based computing are projected to concurrently resolve all of the scheduling task's problems, including reaction time, waiting time, and turnaround time as well as provide efficient dynamic resource allocation, and process workload considerably faster.

REFERENCES

[1] D. A. Chekired and L. Khoukhi, "Smart grid solution for charging and discharging services based on cloud computing scheduling," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 6, pp. 3312–3321, Dec. 2017, doi: 10.1109/TII.2017.2718524.

[2] D. D. Shrimankar, "High performance computing approach for DNA motif discovery," *CSI Transactions on ICT*, vol. 7, no. 4, pp. 295–297, Dec. 2019, doi: 10.1007/s40012-019-00235-w.

[3] Y. Kim, M. Imani, N. Moshiri, and T. Rosing, "GenieHD: efficient DNA pattern matching accelerator using hyperdimensional computing," in *2020 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, Mar. 2020, pp. 115–120, doi: 10.23919/DATE48585.2020.9116397.

[4] R. O. Aburukba, M. AliKarrar, T. Landolsi, and K. El-Fakih, "Scheduling Internet of things requests to minimize latency in hybrid fog–cloud computing," *Future Generation Computer Systems*, vol. 111, pp. 539–551, Oct. 2020, doi: 10.1016/j.future.2019.09.039.

[5] A. I. Moustafa, "Subvention scheme for the cloud computing scheduling algorithm," *Journal of Computers*, vol. 14, no. 9, pp. 571–579, 2019, doi: 10.17706/jcp.14.9.571-579.

[6] D. Zhao, M. Mohamed, and H. Ludwig, "Locality-aware scheduling for containers in cloud computing," *IEEE Transactions on Cloud Computing*, vol. 8, no. 2, pp. 635–646, Apr. 2020, doi: 10.1109/TCC.2018.2794344.

[7] F. Marozzo and D. Talia, "Perspectives on big data, cloud-based data analysis and machine learning systems," *Big Data and Cognitive Computing*, vol. 7, no. 2, p. 104, May 2023, doi: 10.3390/bdcc7020104.

[8] Z. Li, Q. Chen, W. Mo, X. Wang, L. Hu, and Y. Cao, "Converging blockchain and deep learning in UAV network defense strategy: ensuring data security during flight," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 14509 LNCS, pp. 156–171, 2024, doi: 10.1007/978-981-99-9785-5_12.

[9] M. N. Bhuiyan, M. M. Rahman, M. M. Billah, and D. Saha, “Internet of things (IoT): a review of its enabling technologies in healthcare applications, standards protocols, security, and market opportunities,” *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10474–10498, Jul. 2021, doi: 10.1109/JIOT.2021.3062630.
[10] C. Roemer, A. Coulon, T. Disca, and Y. Bas, “Bat sonar and wing morphology predict species vertical niche,” *The Journal of the Acoustical Society of America*, vol. 145, no. 5, pp. 3242–3251, May 2019, doi: 10.1121/1.5102166.
[11] T. Bezdan, M. Zivkovic, E. Tuba, I. Strumberger, N. Bacanin, and M. Tuba, “Multi-objective task scheduling in cloud computing environment by hybridized bat algorithm,” *Advances in Intelligent Systems and Computing*, vol. 1197 AISC, pp. 718–725, 2021, doi: 10.1007/978-3-030-51156-2_83.
[12] A. K. Sandhu, “Big data with cloud computing: discussions and challenges,” *Big Data Mining and Analytics*, vol. 5, no. 1, pp. 32–40, Mar. 2022, doi: 10.26599/BDMA.2021.9020016.
[13] S. Biswas, L. K. Sharma, R. Ranjan, S. Saha, A. Chakraborty, and J. S. Banerjee, “Smart farming and water saving-based intelligent irrigation system implementation using the internet of things,” in *Recent Trends in Computational Intelligence Enabled Research*, Elsevier, 2021, pp. 339–354.
[14] J. J. Sportelli, B. L. Jones, and S. H. Ridgway, “Non-linear phenomena: a common acoustic feature of bottlenose dolphin (Tursiops truncatus) signature whistles,” *Bioacoustics*, vol. 32, no. 3, pp. 241–260, May 2023, doi: 10.1080/09524622.2022.2106306.
[15] S.-Y. Hsieh, C.-S. Liu, R. Buyya, and A. Y. Zomaya, “Utilization-prediction-aware virtual machine consolidation approach for energy-efficient cloud data centers,” *Journal of Parallel and Distributed Computing*, vol. 139, pp. 99–109, May 2020, doi: 10.1016/j.jpdc.2019.12.014.
[16] W. Shu, K. Cai, and N. N. Xiong, “Research on strong agile response task scheduling optimization enhancement with optimal resource usage in green cloud computing,” *Future Generation Computer Systems*, vol. 124, pp. 12–20, Nov. 2021, doi: 10.1016/j.future.2021.05.012.
[17] M. Imani, T. Nassar, A. Rahimi, and T. Rosing, “HDNA: energy-efficient DNA sequencing using hyperdimensional computing,” in *2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, Mar. 2018, pp. 271–274, doi: 10.1109/BHI.2018.8333421.
[18] H. Liu *et al.*, “Thermal-aware and DVFS-enabled big data task scheduling for data centers,” *IEEE Transactions on Big Data*, vol. 4, no. 2, pp. 177–190, Jun. 2018, doi: 10.1109/TBDATA.2017.2763612.
[19] G. Rjoub, J. Bentahar, and O. A. Wahab, “BigTrustScheduling: trust-aware big data task scheduling approach in cloud computing environments,” *Future Generation Computer Systems*, vol. 110, pp. 1079–1097, Sep. 2020, doi: 10.1016/j.future.2019.11.019.
[20] R. Valarmathi and T. Sheela, “Ranging and tuning based particle swarm optimization with bat algorithm for task scheduling in cloud computing,” *Cluster Computing*, vol. 22, pp. 11975–11988, 2019, doi: 10.1007/s10586-017-1534-8.
[21] C.-H. Chen, J.-W. Lin, and S.-Y. Kuo, “MapReduce scheduling for deadline-constrained jobs in heterogeneous cloud computing systems,” *IEEE Transactions on Cloud Computing*, vol. 6, no. 1, pp. 127–140, Jan. 2018, doi: 10.1109/TCC.2015.2474403.
[22] L. De Vito, G. Iadarola, F. Lamonaca, F. Picariello, S. Rapuano, and I. Tudosa, “Non-uniform wavelet bandpass sampling analog-to-information converter: a hardware implementation and its experimental assessment,” *Measurement*, vol. 134, pp. 739–749, Feb. 2019, doi: 10.1016/j.measurement.2018.11.015.
[23] K. Obaideen *et al.*, “An overview of smart irrigation systems using IoT,” *Energy Nexus*, vol. 7, p. 100124, Sep. 2022, doi: 10.1016/j.nexus.2022.100124.
[24] P. Ferrari, E. Sisinni, D. Brandao, and M. Rocha, “Evaluation of communication latency in industrial IoT applications,” in *2017 IEEE International Workshop on Measurement and Networking (M&N)*, Sep. 2017, pp. 1–6, doi: 10.1109/IWMN.2017.8078359.
[25] J. de C. Silva, P. H. M. Pereira, L. L. de Souza, C. N. M. Marins, G. A. B. Marcondes, and J. J. P. C. Rodrigues, “Performance evaluation of IoT network management platforms,” in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Sep. 2018, pp. 259–265, doi: 10.1109/ICACCI.2018.8554364.

BIOGRAPHIES OF AUTHORS

Visalaxi Gandhimathinathan is an assistant professor in the Computer Science and Engineering Department, in Bharath Institute of Higher Education and Research Engineering, Affiliated to Deemed University, Chennai. She received Master of Computer Application (MCA) degree in 2006 from Bharathidasan University, and M.E Computer Science at Jerusalem Engineering College, Chennai. Her research interests are cloud computing, medical image processing, data mining, cloud security, and network security, IoT, machine learning and Python. Pursuing Ph.D. in Bharath Institute of Higher Education and Research. She has Successfully Attained ELITE Certificate in (IoT)- NPTEL. She has published 10 papers in the reputed Journals and Conferences and also attended many Faculties Training Program in the Standard Universities and Colleges. She can be contacted at email: vesalaxi777gg@outlook.com.

Muthukumaravel Alagesan is currently working as a Dean, Arts and Science at Bharath Institute of Higher Education and Research Chennai, India. He obtained his Ph.D. in Computer Science from VELS University, India. His area of specializations includes database management systems and artificial neural network, MATLAB, data structures, object-oriented programming languages. He is having 23 years teaching experience, and 4 years of Ph.D. research work at VELS University, Chennai. He has 65 publications in national and international journals. He also has 50 Scopus indexed publication. He also attended 41 conferences, workshop, seminars and FDP. He is also serving as editor/reviewer for 8 international journals. He supervised 3 M. Phil student, and 7 Ph.D. students he produed. He can be contacted at email: muthukumaravel26a73@outlook.com.