Journal of Applied Engineering and Technological Science
Vol 5(1) 2023: 569-580

OPTIMIZING CLOUD-FOG-EDGE JOB SCHEDULING USING CATASTROPHIC GENETIC ALGORITHM AND BLOCK CHAIN-BASED TRUST: A COLLABORATIVE APPROACH

Nibras A. Mohammed Ali[1*], Firas A. Mohammed Ali[2]
Department Computer Science, College of Education for Women, University of Baghdad, Baghdad, Iraq[1]
Center for Strategic and International Studies, University of Baghdad, Baghdad, Iraq[2]
nebras.ali@coeduw.uobaghdad.edu.iq

Received: 24 August 2023, Revised: 31 October 2023, Accepted: 04 November 2023
Corresponding Authors

ABSTRACT
Collaborative edge-cloud features improve job scheduling. Cloud job scheduling is crucial. Pending delay completion. A cloud-edge mixed system replaced centralized cloud computing. Combining resource levels reduces terminal user service call latency. Decentralization, regionalization, and node dispersal autonomy increase ambiguity, unreliability, and instability. This paper will plan cloud-migrating tasks on edge devices or the cloud to achieve a global optimum. The objective of this research is to enhance the efficiency of job scheduling in cloud-fog edge environments through the integration of the Catastrophic Genetic Algorithm (CGA), a genetic algorithm inspired by natural evolution. Additionally, Berger's theory will be employed to develop a trust-enabled interaction framework based on blockchain technology. The CGA fitness function incorporates load balancing and reasonability in the coordination of services and scheduling of tasks, with the goal of maximizing performance. This article presents proposed improvements to the CGA, which involve the incorporation of mutation and crossover operators, roulette selection, and cataclysm. These changes aim to expand the search area and potentially discover schedules that are more optimal. The approach also effectively deals with the problem of premature convergence, guaranteeing ample time for the algorithm to comprehensively explore the solution space prior to reaching a final solution. The experimental findings indicate that the strategy put forward in this study yields a substantial reduction in task completion time, surpassing 97%. Furthermore, it effectively addresses the best local problem, hence showcasing competing options.

Keywords: *Edge Computing Task Scheduling, CGA, Ca Catastrophic Genetic Algorithm, Blockchain.*

1. Introduction

The conventional centralized cloud service model encounters considerable challenges in practical scenarios, including a single point of failure, network dependency, excessive latency, and failure scale effect. As a result, it is not suitable for accommodating instantaneous transactions. The convergence of cloud computing and edge computing is a prominent subject of investigation within the field of computer science and technology. Cloud computing is a paradigm that involves the storage and processing of data on centralized servers over the internet. In contrast, edge computing refers to the practice of processing data in close proximity to the network, thereby minimizing latency and improving the ability to do real-time processing tasks. The integration being discussed possesses the capacity to bring about significant transformations in several industries, including healthcare, manufacturing, and autonomous cars. This is achieved by facilitating quicker decision-making processes, reducing latency, and improving user experiences (Perumalla, 2021).

The objective of this work is to create a novel approach for job scheduling in cloud-fog edge environments, with the goal of optimizing task allocation and reducing job completion time. The proposed approach utilizes the Catastrophic Genetic Algorithm (CGA) for the purpose of identifying optimal job schedules. Additionally, Berger's theory is employed to construct a safe interaction framework. In addition to its focus on load balancing and reasonability, the technique further enhances the CGA algorithm and showcases substantial performance enhancements through extensive experimentation and simulations (Patil, 2022). This paper presents a novel approach for air scheduling in cloud-fog periphery scenarios, incorporating trust-enhancing mechanisms. A decentralized trust framework has been developed utilizing blockchain technology to tackle the problem of service trustworthiness. The utilization of blockchain technology is poised to become increasingly prevalent in industries that prioritize enhancing privacy and security measures (Li, 2020). The blockchain is a decentralized ledger that preserves unalterable, sequential data without a centralized storage location. In blockchain technology, nodes refer to the devices or participants involved in the network. In a decentralized blockchain network, all network nodes actively validate and verify data. The information that is intended to be stored within the blockchain will undergo encryption through the use of cryptography (Li et al., 2019). Every individual block comprises a timestamp, an encrypted hash, and a hash of the preceding block in the chain to which it will be appended. As a result, the data stored in the blockchain is resistant to tampering. Since participating users are verified within the network, utilizing blockchain technology eliminates any potential privacy risks (Murthy et al., 2020).

The scheduling of tasks can be determined based on energy consumption and latency, with the option of either scheduling at the edge or in a remote cloud regarding the issue that necessitates attention from the cloud center. The scheduling process incorporates a variety of performance-based indicators such as system utilization, execution time, load balancing, network connection cost, and latency (Abd Elaziz et al., 2019). Employing the heuristic approach to the work schedule makes it possible to identify the most favorable solution. Nevertheless, it cannot ensure optimal results and is susceptible to inadequate selection (Yu & Zhang, 2021). The Met heuristic algorithm is a refined iteration of the heuristic algorithm that merges the Cal search technique with random algorithms. The system effectively handles a significant volume of search space information and facilitates the investigation and extension of the search space. Furthermore, it can employ various learning methodologies to obtain and proficiently execute information, resulting in almost flawless solutions. According to recent research, genetic algorithms (GA) have emerged as the dominant evolutionary approach for scheduling tasks in Algeria (Acampora et al., 2023).

The auxiliary operator is expected to be the mutation operator, as it can conduct local searches. On the other hand, the primary operators are expected to be the crossover operators, given their ability to perform global searches. The utilization of genetic algorithms has the potential to achieve parity between the global and local search spaces. This study is centered around the optimization of the distribution of time-sensitive and low-resource-demand services within a network environment that encompasses fog and edge computing alongside conventional cloud computing. The issues that have been highlighted encompass time constraints, resource requirements, achieving an appropriate balance of resources, and minimizing service latency (Kumar, 2021). The proposed solution entails the integration of a unique computing architecture that amalgamates cloud, fog, and edge computing layers. This architecture incorporates a fog layer, which serves as an intermediary management system connecting edge devices and the core cloud infrastructure (Chavda, 2021).

This layer enables effective communication and the allocation of tasks between edge devices and cloud servers. The fog layer is designed to attain an ideal balance of resources, decrease the latency of services, and enable intelligent allocation of services with low resource demands. This strategy effectively manages resource allocation throughout the network, improves real-time functionalities, and mitigates the risk of excessive consumption of cloud resources. The study's findings suggest that the integration of cloud, fog, and edge computing inside a novel computing architecture is essential in order to get an optimal allocation of resources, reduce latency, and efficiently manage services that exhibit diverse resource requirements (Sohail, 2023).

2. Literature Review

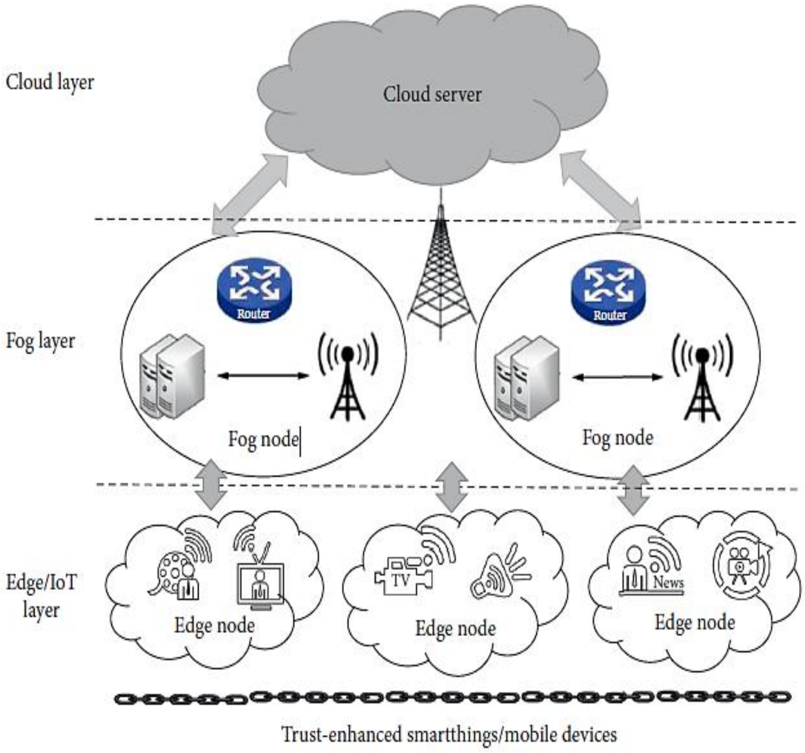
Although the field of edge-cloud collaboration research is still in its nascent stages, many domestic and international scholars have conducted studies and generated research outcomes on edge or cloud work schedules. This paper looks at all the research that has been done on the Moth Search Algorithm (MSA) and how it uses Differential Evolution (DE) to find and get around any problems that might make it less effective. The study pertains to nature-inspired algorithms, specifically the MSA, which emulate natural behaviors. The research focuses on the particular challenge of cloud job scheduling, which is a crucial concern in distributed computing settings. The inclusion of comparative studies is crucial in order to gain a comprehensive understanding of the efficacy and performance of the suggested methodology. Performance metrics are employed for the assessment of the MSDE algorithm, and the inclusion of experiments including both synthetic and actual trace data serves to validate its practicality. The study in the field of optimization and nature-inspired computing is strengthened by the implementation of a meticulous experimental design that incorporates the evaluation of other algorithms and the utilization of both synthetic and real trace data.

The objective of this study is to enhance the efficiency of job offloading in vehicular fog computing (VFC) through the utilization of a multi-objective evolutionary algorithm (MOEA). Four distinct execution and transmission models are put forth, which make use of vehicle resources for the purposes of executing tasks and transmitting data. Dijkstra's algorithm is a widely employed method for determining the shortest path between nodes. The simulation findings demonstrate that the incorporation of cars in the VFC system leads to a noteworthy reduction in latency and total energy when compared to alternative models and existing state-of-the-art approaches (Abdullah & Jabir, 2022). This research study introduces a novel approach that combines meta-heuristic techniques with the HEFT algorithm to address the work scheduling problem in cloud computing. The proposed method demonstrates superior performance in terms of makes pan when compared to three alternative heuristic and genetic algorithms. The evaluation is conducted on randomly generated direct acyclic graphs (DAGs). (Kamalinia & Ghaffari, 2017), a hybrid metaheuristic algorithm is proposed by integrating AG and PSO, resulting in improved performance through the combination of Completion Time for the algorithm. Johnson introduced a genetic algorithm that operates based on a set of predefined rules. (JRGA). Two-stage task scheduling is implemented in data centers, as described in reference Abdul Kareem & Hussein (2022).

The presented MPQGA algorithm uses a genetic algorithm (GA) along with a heuristic-based earliest finish time (EFT) method to efficiently sort subtasks and thoroughly look for solutions during the task-to-processor mapping process. In addition, it is responsible for the design of crossover, mutation, and fitness functions pertaining to the scheduling of directed acyclic graphs. The experimental findings demonstrate that it exhibits superior performance compared to non-evolutionary heuristics and random search techniques (Xu et al., 2014). The scheme above allocates tasks to processors following their respective priorities. Trust plays a significant role in mitigating issues related to reputation and reliability within a decentralized and inclusive network. The establishment of trust is often a consequence of many achievements. A study was conducted in (Jiang et al., 2015) regarding techniques for assessing trust in sensor cloud systems. The authors employed the cloud model and attribute-weighted clustering techniques to devise an approach for assessing the credibility of recommendations, as described in reference (Wang et al., 2018). The fog-based spider web algorithm (FSWA) is a heuristic methodology that aims to minimize delays and improve reaction time in processes within a fog network. The objective of this endeavor is to identify the closest nodes for computational purposes, thereby enhancing quality of service (QoS) metrics, optimizing resource allocation, and increasing service availability (DAR et al., 2020). About (Mohammed Ali & Al-Tamimi, 2022), This study presents a novel trust model that utilizes a trust certificate authority for the purpose of calculating domain and global trust in public cloud environments. The proposed model aims to decrease computational complexity and enhance overall performance. The experimental findings demonstrate the effective and precise calculation of trust levels. A context-aware trust prediction model was developed for edge computing in vehicles, as described in reference (Li et al., 2019).

3. System Overview

Model for Cloud Center Task Scheduling Finding a rational way to divide each task among several virtual machines to complete each task with the slightest delay is the challenge with cloud task scheduling (Yang et al., 2020; Mohamed & Al-Tamimi, 2020). To decide whether to execute in the cloud, each task with three categories must first be analyzed. The sensitivity of the task is established by the relationship between the task's delay and its duration. Finally, the cloud-based jobs will be planned to minimize overall execution time (Dubuque et al., 2005). To build a model in the IoT device, blockchain technology is introduced. The framework comprises three layers, as shown in figure (1): the cloud layer, the fog layer, and the trust-enhanced edge/IoT layer.



Fig, 1. A Hybrid Cloud-Fog-Edge Framework With Greater Trust.

4. Task Classification

To decide whether to execute on the cloud, one must first categorize the tasks that need to be processed. The sensitivity of the task is established by the relationship between the task's delay and its duration (Lucas, 2014). Finally, the cloud-based jobs will be planned to minimize overall execution time. Let f_i^c Indicate the computational power that the edge device has allocated to Task I. Consequently, can obtain the local Task I execution time (Rodionova et al., 2019).

$$as\mathbf{T}^{\mathbf{i}}_{\mathbf{local}} = \frac{\mathbf{data_i}}{\mathbf{f_i^c}}$$

the time transferred is defined as

$$\mathrm{T}^{\mathrm{i}}_{\mathrm{tran}} = \frac{\mathrm{d_i}}{\mathrm{Rate}}$$

Rate is the rate at which cloud-based duties are uploaded; in this case, the upload rate is a measurable quantity (Rylander et al, 2000). To facilitate the cloud task scheduling procedure, tasks must be categorized by sensitivity. You can define assignment sensitivity as follows:

$$\mathrm{senTi} = \frac{\mathrm{data_i}}{\mathrm{expT}_{\mathrm{i}}}$$

5. CGA Algorithm

The study of algorithmic logic. The three steps of the genetic algorithm are responsible for determining the convergence rate. The present study aims to enhance the efficacy of the genetic algorithm by optimizing the selection, crossover, and mutation processes (Mora-Melià et al., 2017). This approach mimics the biological evolution process to generate a new population of individuals. Catastrophic events lead to increased individual diversity without a corresponding increase in population size . Additionally, the removal of an ideal local trap becomes a less challenging task. The algorithm flowchart is shown in flowing figure 2:

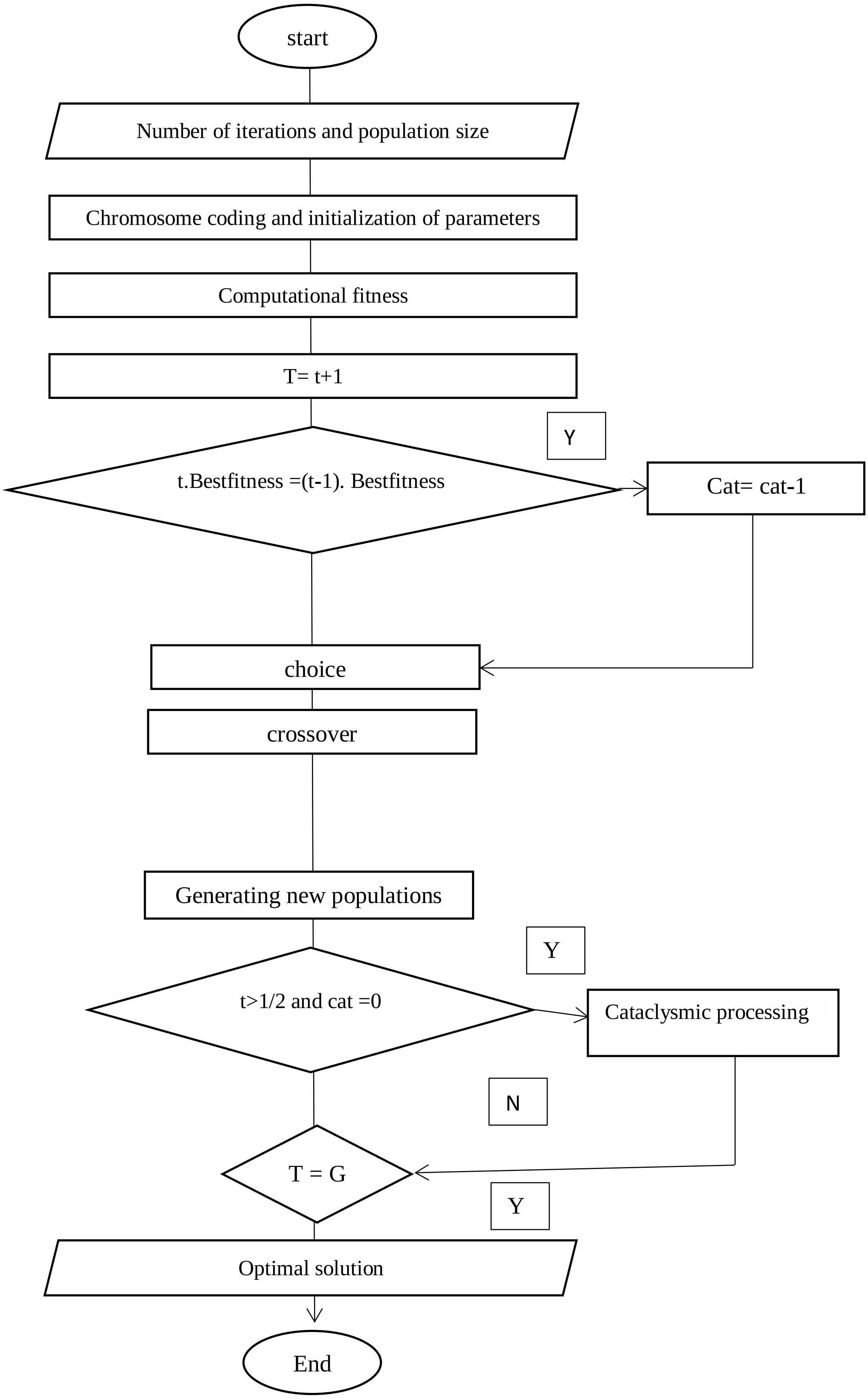

start
Number of iterations and population size
Chromosome coding and initialization of parameters
Computational fitness
T= t+1
Y
t.Bestfitness =(t-1). Bestfitness
Cat= cat-1
choice
crossover
Generating new populations
Y
t>1/2 and cat =0
Cataclysmic processing
N
T = G
Y
Optimal solution
End

Fig 2. The Flowchart of Compact Genetic (CGA) Algorithmic.

5.1 Basic Algorithm Operations

Encoding. Typically, multi-to-one mapping coupling encoding, also known as accurate coding, encodes cloud computing scheduling problem solutions (Harik et al., 1999). This paper uses the mapping pairing method to implement both the virtual machine and the task (Kamalinia & Ghaffari, 2017). For example, if there are M, VMS, that is, {v1, v2, v3, . . ., vM}, and N tasks, that is, {Tak1, Task2, Task3, . . ., Task n}, Figure 3 illustrates that the value of each gene ranges from 1 to M When the code's length is N:

Task	t1	t2	t3	t4	t5	t6
VM	v1	v3	v2	v4	v5	v6
Code	1	3	2	4	5	5

Fig. 3. Encoding.

5.2 Fitness Element.

The fitness function is a measure of an individual's level of fitness across their evolutionary past. In the evolutionary process, fitness increases with ease of retention (Supasil et al., 2021). The fitness function will directly impact the algorithm's performance and ability to achieve the objective (Ahn et al., 2004). In this paper, the influence of execution time and time delay on physical endurance must be considered. The delay between each task's execution and deadline is as follows:

$$\sum_{i\in[1,i]} a_{i,j}ECT_{i,j} + T_{tran}^{i} - \exp T_i$$

If $\sum_{i\in[1,i]} a_{i,j}\, ECT_{i,j} + T_{tran}^{i} - \exp T_i \leq 0,$

$$punish = \begin{cases} 0, & if\ \sum_{i\in[1,i]} a_{i,j}\, ECT_{i,j} + T_{tran}^{i} - \exp T_i \leq 0 \\ \left|\sum_{i\in[1,i]} ba_{i,j}ECT_{i,j} + T_{tran}^{i} - \exp T_i\right|, & if\ \sum_{i\in[1,i]} a_{i,j}\, ECT_{i,j} + T_{tran}^{i} - \exp T_i \leq 0 \end{cases}$$

Because reducing the overall task execution time while meeting task deadlines is the goal of this paper, the fitness function was designed as follows.

$$fitness = \frac{1}{\text{AIINTime} + \sum_{\text{i}=1}^{\text{Nitsk}} \text{punish}}$$

5.3 Optimize Roulette

The roulette selection approach is also known as proportionate selection. Selection is easier for flexible people (Chmiel & Kwiecien, 2018). This study uses roulette to pass on the fittest people to the next generation without crossover or mutation operations (Siddiqi et al., 2013). The remaining individuals choose the progeny population using standard roulette. the likelihood Ps(j) of a single choice in conventional roulette is

$$Ps(j) = \frac{fitness(j)}{\sum_{i=1}^{N} fitness(i)}$$

5.4 Crossover

The conventional genetic algorithm incorporates a crossover operation that entails selecting the crossover rate to establish the number of individuals to be crossed, utilizing rand (1, n) to generate a crossover operation for every intersecting individual, and mapping the two chromosomes to the segments following the displacement of the placement points (Joseph et al., 2014). The present research establishes a threshold for determining statistical significance across different groups or conditions. In the absence of a match, no further action will be taken. The cutoff size denotes the proportion of associated genes within the gene (Tian et al., 2019). This process effectively mitigates the adverse effects of inbreeding and enhances the quality of offspring in the context of human evolution. The article above established a threshold of 0.8 and a crossover probability exceeding 0.7 to prevent the cessation of cross operation due to excessive similarity, which may impede the convergence process (da S. Medeiros et al., 2020). The crossover process is depicted in Figure 4. The delay penalty refers to the negative consequence or punishment resulting from a delay in completing a task or meeting a deadline.

$$if \sum_{i\in[1,i]} a_{i,j}ECT_{i,j} + T_{tran}^{i} - \exp T_i \leq 0,$$

$$\sum_{i\in[1,i]} a_{i,j}ECT_{i,j} + T_{tran}^{i} - \exp T_i > 0$$

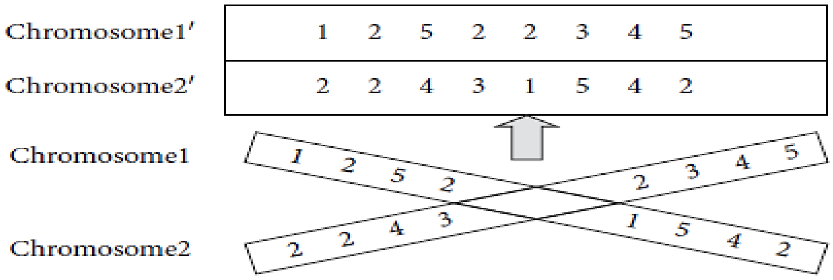


Fig 4. A Crossover Operation Of The Conventional Genetic Algorithm That Entails Selecting The Crossover Rate To Establish The Number Of Individuals To Be Crossed

5.5 Variation

The mutation is crucial. Mutations in genetic systems serve two purposes. First, local random scan (Abd Elaziz et al., 2019). This work sets two variability values. Over two-thirds of repeats reduce mutation chance by 0.02. Determine the number of mutations needed, choose two random chromosomal regions and swap gene values. Since the chance of the variation procedure being carried out is already low, it is enhanced (Narayanan & Moore, 1996). The data presented in Figure 5 suggests that the genic value may have remained constant after the mutation process (Sharma, 2020).

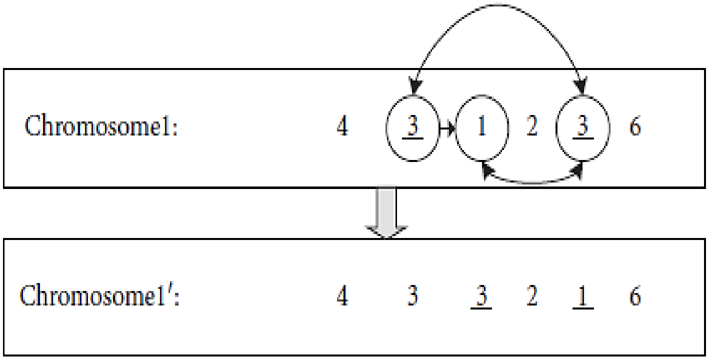


Fig. 5. The Mechanical Of Variation In Genetic Systems.

5.6 Task Scheduling and Classification Description

The steps of the genetic algorithm are responsible for determining the convergence rate. This approach mimics the biological evolution process to generate a new population of individuals (Harik et al., 1999). Additionally, the removal of an ideal local trap becomes a less challenging task. We review the steps for scheduling tasks and describing the classification in the form of steps as follows:

1. Is to categorize every task performed on every device.
2. Sort jobs by sensitivity, the initial coding is optimized for the virtual machines processing power (Shor, 1997).
3. Coding and parameter initialization and then calculating fitness.

4. Algebras of supersession plus one, the catastrophe threshold is lowered by one of the optimal individual fitness of the (t1) the generation is equal to that of the generation; otherwise, it will remain the same (Al-Tamimi & Mohammed Ali, 2023).
5. Execute the selection, cross, and mutation operations.
6. Create the descendant population and check to see if the catastrophe threshold cat equals 0. Continue the catastrophe operation if it equals 0.
7. If the maximum number of iterations has been reached, output; if not move on to step 3.

6. Evaluation

This experiment represented a standard terminal-to-fog server device. The simulation includes (Jyoti & Chauhan, 2022) based on catastrophic algorithms. Data about computing capacity for resources and calculations are extracted from randomly generated data in MATLAB. This study aims to analyze and contrast the iteration durations of Berger's theory and the time-based differential simple genetic algorithm while utilizing the same dataset. Utilizing blockchain technology to construct a framework for task scheduling that promotes trust and assurance (Cao et al., 2016). The model being proposed integrates a decentralized ledger system. The block of transactions effectively and securely stores the factual transactions of the system. Trust management comprises three fundamental elements: establishment, decision-making, and upkeep (Brakerski et al., 2021).

Furthermore, the reliable block is responsible for monitoring and recording the trustworthiness ratings of the individual nodes. The creation of a confidence assessment allows enterprises to choose reliable trade associates effectively (Grover, 1996). When job opportunities are limited, determining the most favorable result may not be readily discernible. As additional responsibilities are incorporated, the algorithms gain greater transparency. As the quantity of tasks escalates, the duration of execution for each task also increases, reducing the number of tasks uploaded to the cloud (Baritompa et al., 2005). The results in Figures 7-9 demonstrate that the proposed method exhibits a faster convergence rate and greater time efficiency as evolutionary algebra progresses.

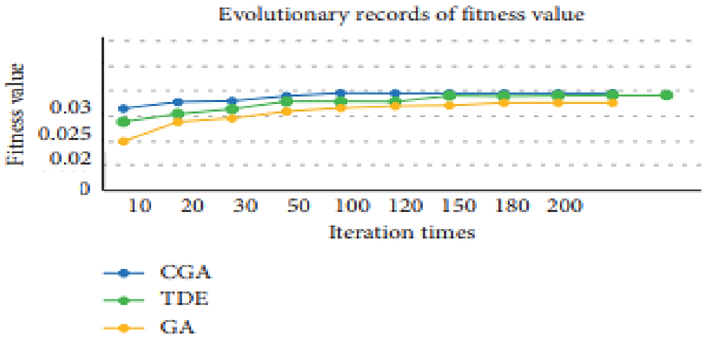


Fig. 6. The Evolutionary Records Of Fitness Value And The Times Of Iteration

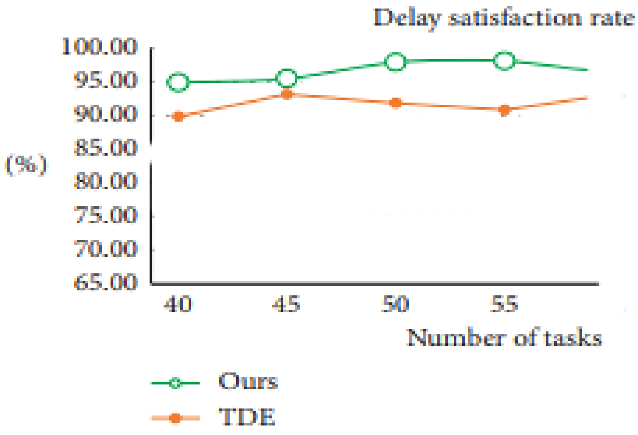


Fig. 7. The Delay satisfaction rate for the number of tasks.

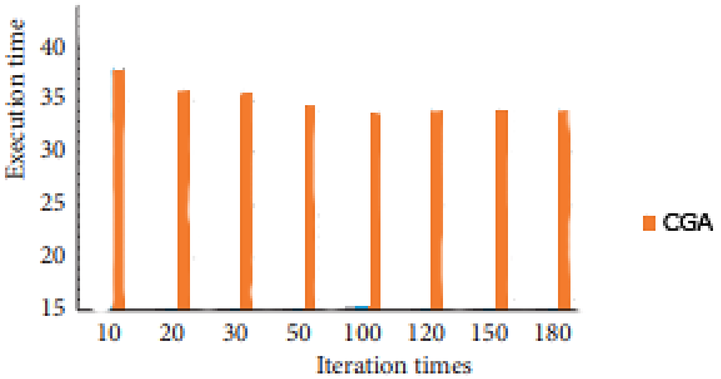


Fig 8. The Execution Time For The Iterations Using CGA Algorithm

The experimental findings suggest that the satisfaction rate with delay was over 97%, meeting the required demand. The CGA approach exhibits superior performance in task completion time and convergence speed of the evolutionary process (Fürer, 2008). As the number of iterations increases, the CGA algorithm can better locate the optimal solution and accelerate convergence. The study employed a calamitous technique that did not demonstrate any impact on the convergence rate or the optimal course. Given the challenge of achieving the local optimum, a viable alternative is to facilitate the operation in its ongoing efforts to improve the population (G. Sun et al., 2014).

7. Conclusion

Work schedule is a crucial concern within the context of collaborative edge-cloud environments. It is imperative to consider whether the task is executed in the cloud and how it is scheduled therein. Although blockchain presents significant challenges, genetic algorithms offer unique benefits compared to traditional methods for addressing intricate issues. Integrating blockchain technology can establish a structure for interactions based on trust within a cloud-fog environment. The current study introduces a novel approach that combines cloud, fog, and edge computing to create a hybrid environment. The proposed system also incorporates a trust-enhanced location-aware fair labor scheduling paradigm. The new model comprises three architectural layers: IoT, fog, and cloud. Utilizing the fog layer enables the facilitation of cloud-fog or fog-edge resource coordination and unified scheduling. The approach enabled edge devices to opt for either on-site or cloud-based execution venues for their tasks. The objective is to diminish the duration needed to accomplish each undertaking, and this information cannot be owned, regulated, or manipulated by an individual entity.

Enhancing the reliability and confirmability of task execution while minimizing administrative expenses. The CGA algorithm represents a supplementary methodology for addressing the issue of optimizing task scheduling. Berger's concept has been implemented to address the matter of impartiality in task scheduling. The CGA algorithm will probably undergo enhancements in the foreseeable future, enabling its application in dynamic and real-time task scheduling for edge-cloud collaboration under more realistic conditions. Besides memory consumption, the surge in demand, and system overloads, the task scheduling methodology can incorporate many supplementary variables. Furthermore, a parallel computing framework can be integrated to incorporate the Markov chain into our model. Data availability is limited to time and static duties as this work solely focuses on these aspects

References

Abd Elaziz, M., Xiong, S., Jayasena, K. P. N., & Li, L. (2019). Task scheduling in cloud computing based on hybrid moth search algorithm and differential evolution. *Knowledge-Based Systems*, 169, 39-52. https://doi.org/10.1016/j.knosys.2019.01.023

Abdul Kareem, E. I ., & Hussein, S. A. (2022). Optimal CPU Jobs Scheduling Method Based on Simulated Annealing Algorithm. *Iraqi Journal of Science*, 63(8), 3640–3651. https://doi.org/10.24996/ijs.2022.63.8.38

Abdullah, S. K., & Jabir, A. J. (2022). A Multi-Objective Task Offloading Optimization for Vehicular Fog Computing. *Iraqi Journal of Science*, 36(2), 785-800. https://doi.org/10.24996/ijs.2022.63.2.33

Acampora, G., Chiatto, A., & Vitiello, A. (2023). Genetic algorithms as classical optimizer for the Quantum Approximate Optimization Algorithm. *Applied Soft Computing*, 142, 110296. https://doi.org/10.1016/j.asoc.2023.110296

Ahn, C. W., Kim, K. P., & Ramakrishna, R. S. (2004). A memory-efficient elitist genetic algorithm. *In Parallel Processing and Applied Mathematics: 5th International Conference, PPAM 2003*, Czestochowa, Poland, September 7-10, 2003. Revised Papers 5 (pp. 552-559). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-24669-5_72

Al-Tamimi, M. S., & Mohammed Ali, F. A. (2023). Face mask detection based on algorithm YOLOv5s. *International Journal of Nonlinear Analysis and Applications*, 14(1), 679-697. https://doi.org/10.22075/ijnaa.2022.28178.3824

Baritompa, W. P., Bulger, D. W., & Wood, G. R. (2005). Grover's quantum algorithm applied to global optimization. *SIAM Journal on Optimization*, 15(4), 1170-1184. https://doi.org/10.1137/040605072

Brakerski, Z., Christiano, P., Mahadev, U., Vazirani, U., & Vidick, T. (2021). A cryptographic test of quantumness and certifiable randomness from a single quantum device. *Journal of the ACM (JACM)*, 68(5), 1-47. https://doi.org/10.1145/3441309

Cao, Z., Zhou, H., Yuan, X., & Ma, X. (2016). Source-independent quantum random number generation. *Physical Review X*, 6(1), 011020. https://doi.org/10.1103/PhysRevX.6.011020

Chavda, A. D. (2021). Multi-stage CNN architecture for face mask detection. *International Conference for Convergence in Technology*, (pp. 1-8). IEEE. https://doi.org/10.1109/I2CT51068.2021.9418207

Chmiel, W., & Kwiecien, J. (2018). Quantum-inspired evolutionary approach for the quadratic assignment problem. *Entropy*, 20(10), 781. https://doi.org/10.3390/e20100781

da S. Medeiros, D. R., Torquato, M. F., & Fernandes, M. A. (2020). Embedded genetic algorithm for low-power, low-cost, and low-size-memory devices. *Engineering Reports*, 2(9), e12231. https://doi.org/10.1002/eng2.12231

DAR, A. R., Ravindran, D., & Islam, S. (2020). Fog-based spider web algorithm to overcome latency in cloud computing. *Iraqi Journal of Science*, 61(7), 1781-1790. https://doi.org/10.24996/ijs.2020.61.7.27

Dubuque, I. A. Adomavicius, G., A. Tuzhilin. (2005). Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering, 17*(6), 734–749. https://doi.org/10.1109/TKDE.2005.99

Fürer, M. (2008). Solving NP-complete problems with quantum search. *In Latin American Symposium on Theoretical Informatics (pp. 784-792)*. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-78773-0_67

Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. *In Proceedings of the twenty-eighth annual ACM symposium on Theory of computing (pp. 212-219)*. https://doi.org/10.1145/237814.237866

Harik, G. R., Lobo, F. G., & Goldberg, D. E. (1999). The compact genetic algorithm. *IEEE transactions on evolutionary computation*, 3(4), 287-297. https://doi.org/10.1109/4235.797971

Harik, G., Cantú-Paz, E., Goldberg, D. E., & Miller, B. L. (1999). The gambler's ruin problem, genetic algorithms, and the sizing of populations. *Evolutionary computation*, 7(3), 231-253. https://doi.org/10.1162/evco.1999.7.3.231

Jiang, J., Han, G., Shu, L., Chan, S., & Wang, K. (2015). A trust model based on cloud theory in underwater acoustic sensor networks. *IEEE Transactions on Industrial Informatics*, 13(1), 342-350. https://doi.org/10.1109/TII.2015.2510226

Joseph, C. T., Chandrasekaran, K., & Cyriac, R. (2014). Improving the efficiency of genetic algorithm approach to virtual machine allocation. *In 2014 International Conference on Computer and Communication Technology (ICCCT) (pp. 111-116)*. IEEE. https://doi.org/10.1109/ICCCT.2014.7001477

Jyoti, A., & Chauhan, R. K. (2022). A blockchain and smart contract-based data provenance collection and storing in cloud environment. *Wireless Networks*, 28(4), 1541-1562. https://doi.org/10.1007/s11276-022-02924-y

Kamalinia, A., & Ghaffari, A. (2017). Hybrid task scheduling method for cloud computing by genetic and DE algorithms. *Wireless personal communications*, 97, 6301-6323. https://doi.org/10.1007/s11277-017-4839-2

Kumar, A. K. (2021). A hybrid tiny YOLO v4-SPP module based improved face mask detection vision system. J. Ambient Intell. Humaniz. *Comput*, p1–14. https://doi.org/10.1007/s12652-021-03541-x

Li, C. C. (2020). *Proceedings of the 2nd International Conference on Artificial Intelligence and Advanced Manufacture*. Association for Computing Machinery, pp. 74–77. https://doi.org/10.1145/3421766.3421768.

Li, Y., Wang, X., Gan, X., Jin, H., Fu, L., & Wang, X. (2019). Learning-aided computation offloading for trusted collaborative mobile edge computing. *IEEE Transactions on Mobile Computing*, 19(12), 2833-2849. https://doi.org/10.1109/TMC.2019.2934103

Lucas, A. (2014). Ising formulations of many NP problems. *Frontiers in physics*, PP2, 5. https://doi.org/10.3389/fphy.2014.00005

Mohamed, N. A., & Al-Tamimi, M. S. H. (2020). Image fusion using a convolutional neural network. *Solid State Technol*, 63(6), 13149-13162.

Mohammed Ali, F. A., & Al-Tamimi, M. S. (2022). Face mask detection methods and techniques: A review. *International Journal of Nonlinear Analysis and Applications*, 13(1), 3811-3823. https://doi.org/10.22075/ijnaa.2022.6166

Mora-Melià, D., Martínez-Solano, F. J., Iglesias-Rey, P. L., & Gutiérrez-Bahamondes, J. H. (2017). Population size influence on the efficiency of evolutionary algorithms to design water networks. *Procedia Engineering*, 186, 341-348. https://doi.org/10.1016/j.proeng.2017.03.209

Murthy, C. V. B., Shri, M. L., Kadry, S., & Lim, S. (2020). Blockchain based cloud computing: Architecture and research challenges. *IEEE access*, 8, 205190-205205. https://doi.org/10.1109/ACCESS.2020.3036812

Narayanan, A., & Moore, M. (1996). Quantum-inspired genetic algorithms. *In Proceedings of IEEE international conference on evolutionary computation (pp. 61-66)*. IEEE. https://doi.org/10.1109/ICEC.1996.542334

Patil, S. (2022). Block Chain-Based System: Applications and Challenges. International Journal of Research and Analysis in Science and Engineering, pp.2(2), 9-9.

Perumalla, S. C. (2021). Block chain-based access control and intrusion detection system in iod. International Conference on Communication and Electronics Systems (ICCES) IEEE, pp. 511-518. https://doi.org/10.1109/ICCES51350.2021.9488948

Rodionova, A., Antonov, K., Buzdalova, A., & Doerr, C. (2019). Offspring population size matters when comparing evolutionary algorithms with self-adjusting mutation rates. *In Proceedings of the Genetic and Evolutionary Computation Conference (pp. 855-863)*. https://doi.org/10.1145/3321707.3321827

Rylander, B., Soule, T., Foster, J. A., & Alves-Foss, J. (2000). Quantum Genetic Algorithms. *In GECCO (p. 373)*.

Sharma, V. (2020). *Face mask detection using yolov5 for COVID-19*. Doctoral dissertation, California State University San Marcos.

Shor, P. W. (1999). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 41(2), 303-332. https://doi.org/10.1137/S0036144598347011.

Siddiqi, U. F., Shiraishi, Y., & Sait, S. M. (2013). Memory-efficient genetic algorithm for path optimization in embedded systems. *IPSJ Online Transactions*, 6, 28-36. https://doi.org/10.2197/ipsjtrans.6.28

Sohail, A. (2023). Genetic algorithms in the fields of artificial intelligence and data sciences. *Annals of Data Science*, 10(4), 1007-1018. https://doi.org/10.1007/s40745-021-00354-9

Sun, G., Su, S., & Xu, M. (2014). Quantum algorithm for polynomial root finding problem. *In 2014 Tenth International Conference on Computational Intelligence and Security (pp. 469-473).* IEEE. https://doi.org/10.1109/CIS.2014.40

Supasil, J., Pathumsoot, P., & Suwanna, S. (2021). Simulation of implementable quantum-assisted genetic algorithm. *In Journal of Physics: Conference Series* (Vol. 1719, No. 1, p. 012102). IOP Publishing. https://doi.org/10.1088/1742-6596/1719/1/012102.

Tian, Z., Chu, X., Wang, X., Wei, X., & Shen, C. (2022). Fully convolutional one-stage 3d object detection on lidar range images. *Advances in Neural Information Processing Systems*, 35, 34899-34911. https://doi.org/10.1109/ICCV.2019.00972.

Wang, T., Zhang, G. X., Cai, S. B., Jia, W. J., & Wang, G. J. (2018). Survey on trust evaluation mechanism in sensor-cloud. *Journal on Communications*, 39(6), 37-51. http://dx.doi.org/10.11959/j.issn.1000-436x.2018098

Xu, Y., Li, K., Hu, J., & Li, K. (2014). A genetic algorithm for task scheduling on heterogeneous computing systems using multiple priority queues. *Information Sciences*, 270, 255-287. https://doi.org/10.1016/j.ins.2014.02.122

Yang, H., Cho, J. H., Son, H., & Lee, D. (2020). Context-aware trust estimation for realtime crowdsensing services in vehicular edge networks. *In 2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC) (pp. 1-6)*. IEEE. https://doi.org/10.1109/CCNC46108.2020.9045221

Yu, J., & Zhang, W. (2021). Face mask wearing detection algorithm based on improved YOLO-v4. *Sensors*, 21(9), 3263.https://doi.org/10.3390/s21093263