ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20230404000503
Original scientific paper

Block Chain Technology Assisted Privacy Preserving Resource Allocation Scheme for Internet of Things Based Cloud Computing

Raja RAMAMOORTHI, Arunprakash RAMASAMY*

Abstract: Resource scheduling in cloud environments is a complex task, as it involves allocating suitable resources based on Quality of Service (QoS) requirements. Existing resource allocation policies face challenges due to resource dispersion, heterogeneity, and uncertainty. In this research, the authors propose a novel approach called Quasi-Oppositional Artificial Jellyfish Optimization Algorithm (QO-AJFOA) for resource scheduling in cloud computing (CC) environments. The QO-AJFOA model aims to optimize the allocation of computing power and bandwidth resources in servers, with the goal of maximizing long-term utility. The technique combines quasi-oppositional based learning (QOBL) with traditional AJFOA. Additionally, a blockchain-assisted Smart Contract protocol is used to distribute resource allocation, ensuring agreement on wireless channel utilization. Experimental validation of the QO-AJFOA technique demonstrates its promising performance compared to recent models, as tested with varying numbers of tasks and iterations. The proposed approach addresses the challenges of resource scheduling in cloud environments and contributes to the existing literature on resource allocation policies.

Keywords: block chain; cloud computing; internet of things (IoT); resource scheduling; metaheuristics; MEC server

1 INTRODUCTION

Cloud Computing (CC) refers to a pattern which has been utilized over years presently, executing tasks for users in an efficient, and reliable way. CC model offers distinct kinds of services, namely Platform-as-a-Service (PaaS), Infrastructure-as-a-Service (IaaS), and Software-as-a-Service (SaaS). It further presents distinct kinds of tasks executed for addressing communication, computation, and storage necessities of users [1]. Cloud services mostly depend upon the usage of virtual machines ideology that splits the resources logically in a cloud, as individual machines, for alleviating the user to receive services from Pay-as-you-Go way [2, 3]. It says that the less VMs or resources were utilized, the less payment for the services which took place. Next, in large applications which demand a big number of Virtual Resources (VRs), (that can be used for the VMs generalization), optimization of the VRs deployment should be utilized for the minimum time that might save users money [4]. Moreover, the reduced use of VRs results in advanced stages of power consumption and effectiveness.

Public clouds grant distinct choices of services which are adopted in complicated smart Internet of Things (IoT) applications, namely industrial IoT and Smart City, for improving mostly every Quality of Service (QoS) measure of such applications [5]. These services indulge infrastructure levels, platforms, and software; it alleviates data outsourcing or information and storage scrutiny. Continuous data streams and information were anticipated to increase a problem in these complicated smart IoT-Cloud scenarios [6]. Additionally, such streams might carry more delicate and confidential data which should be protected after and at the time of their storage and processing in cloud. Since an extension of a cloud at network edge. Applications of the IoT patterns become the inspiration of the FC outline since many applications need short response time and real-time services [7]. The reason behind this was Things were resource limited (that is computation power, energy, and storage,). Also, the cloud fog layer was expected to control the data streams and information dynamically amongst every network party. For allowing the incorporation of IoT, CC, and FC, are currently ready, namely Security-as-a-Service, Backend-as-a-Service, and Process-as-a-Service.

Blockchain (BC) refers to a base technology of numerous existing cryptocurrency algorithms. BC is an undergoing technology which ensures a trusty, fully distributed, digital monetary mechanism [8]. From that time, BC was suggested to enhance trust and safety in several applications, through deployment for handling identity, data, reputation, payments, or other reasons. Additionally, the capability of offering de-centralized decisions through BC networks inspired several authors for deploying it in the de-centralized, dispersed Cloud servers at the network edge [9]. Then, Blockchain-as-a-Service (BaaS) has been suggested and scrutinized in the article, as a service method same as Process-as-a-Service systems. So, distinct BaaS methods were powerfully anticipated for supporting CC at the time of life cycle of management and information processing in tough cases [10]. Therefore, BC offering immutable storage abilities and computational power must reduce the cloud burden.

This study develops a novel quasi-oppositional artificial jellyfish optimization algorithm (QO-AJFOA) for RS in the CC environment. The presented QO-AJFOA model majorly aims to optimally allocate the resources related to computing power (CP) and bandwidth of every server for maximizing the long-term utility of the devices. In addition, the QO-AJFOA technique contains the combination of quasi-oppositional based learning (QOBL) method with the traditional AJFOA. Moreover, a BC assisted Smart Contract protocol is used to distribute resource allocation, enforcing the clients to reach an agreement on the wireless channel utilization. The experimental validation of the QO-AJFOA technique is tested under varying number of tasks and iterations.

2 RELATED WORKS

Gai et al. [11] suggest a new technique which complies IoT with BC and edge computing that is known as BC-related Internet of Edge model. The suggested method, devised for a flexible and manageable IoT mechanism,

adequately abuses benefits of BC and edge computing for establishing a privacy-preserving system while taking other restrains, like energy cost. Gai et al. [12] suggest a model permission-BC-edge-model for smart grid network (PBEM-SGN) for addressing the 2 vital problems in smart grid, energy security and privacy protections, by compiling edge computing and BC approaches. The researchers utilize group signatures and transform channel authorization methods for guaranteeing validity of users. The best security-aware strategy was made through smart contracts which run on the BC.

In [13], the authors suggest cloud mining and agent mining techniques for solving the above-mentioned issue in the BC enabled IoT. In particular, miners serve as mining mediators for sensors in IoT, offload mining errands to CC servers, and employ network resources forcefully. Besides, for improvising the outcome, networking resources allocation and computing resources allocation, the access selection of users, were framed as a joint optimizing issue. Li et al. [14] suggest a BC-enabled secure storage policy for logistics data. Specifically, in brief, the scheme could be classified into 2 parts. The first one includes aggregation and data generation, session accomplishing, records encoding and memory, where a BC network was utilized for assisting the cloud server having data storage, and smart contracts were arranged for presenting dependable storage interfaces. Next, a potential consensus system was launched for enhancing efficacy of the consensus process. Similarly, the saved reports should be audited securely by using the deployed network of BC.

A BC enabled distributed security structure leveraging software-defined networking (SDN) and edge cloud was launched in [15]. The security attack identification was attained at the cloud layer, and security assaults were consequently minimized at the IoTs edge layer. The SDN assisted gateway suggestions dynamic network traffic flow administration, that makes a contribution to the security attack identification via determination of suspected network traffic flows and decreases security assaults by impeding doubtful flows. In [16], a de-centralized and privacy-preserving charging scheme for EVs was suggested, that depended upon fog computing and BC. In this, fog computing can be launched for offering local computing having low latency. To be specific, a fog computing system that was made up of fog computing nodes (FCNs) was utilized for granting localized services. Also, a flexible consortium BC architecture was suggested. The BC mechanism was deployed over the distributed FCNs, granting a de-centralized and secure storage ambience.

The term "Internet of Things" (IoT) refers to a network of networked physical devices, automobiles, appliances, and other items that are equipped with sensors, software, and connection to gather and share data. This network may also include people. The term "cloud computing" refers to the distribution of computer resources and services through the internet in a manner that is both scalable and on-demand. The Internet of Things generates enormous volumes of data, which necessitates the use of cloud computing resources for the purposes of data storage, processing, and analysis [23].

IoT devices depend on the resources of cloud computing to offload data processing and storage. Resource Allocation in IoT-based Cloud Computing IoT devices rely on the resources of cloud computing to offload data processing and storage. In order to guarantee the most effective usage of cloud resources and to satisfy the requirements of IoT applications, efficient resource allocation is essential. When sensitive data from IoT devices is shared with cloud service providers for the purpose of resource allocation, however, privacy problems emerge. Devices that are part of the Internet of Things will often gather and communicate sensitive data, such as personal and location information. It is essential for Internet of Things (IoT) systems to maintain users' privacy in order to prevent illegal access to user data, maintain data integrity, and comply with privacy standards [24]. During data transfer, storage, and processing, strategies for the protection of privacy strive to restrict the exposure of sensitive information to the absolute minimum. Transactions may be carried out in a way that is both private and open thanks to blockchain technology, which consists of a decentralized and unchangeable ledger. Through the use of cryptographic algorithms and distributed consensus procedures, it guarantees the data's privacy, as well as its integrity and immutability. Due to the fact that Blockchain is decentralized, there is no longer a need for a centralized authority. As a result, the dangers associated with data modification and unlawful access have been significantly reduced. Using the capabilities of blockchain technology, it is possible to design resource allocation protocols that protect users' anonymity in IoT-based cloud computing environments. Blockchain technology offers a decentralized and tamper-proof architecture that makes it possible for Internet of Things devices to safely communicate data with cloud service providers while still retaining control over their private information. The use of smart contracts on the blockchain makes it possible to make automatic resource distribution choices while protecting users' privacy. These decisions are based on specified rules and agreements [25].

3 THE PROPOSED MODEL

In this study, an effective QO-AJFOA technique was established for RS in the CC environment. The presented QO-AJFOA model majorly aims to optimally allocate the resources related to CP and bandwidth of every server for maximizing the long-term utility of the devices. Moreover, a BC assisted Smart Contract protocol is used to distribute resource allocation, enforcing the clients to reach an agreement on the wireless channel utilization. Fig. 1 depicts the overall process of proposed method.

3.1 Design of QO-AJFOA

In this study, the QO-AJFOA technique involves the integration of the QOBL concept into the traditional AJFOA. Chou and Truong [17] introduced an AJFOA that was based on the search and movement behaviors of jellyfish (JF) in the ocean. Based on the subsequent three methodologies, the JS is implemented:

The JF obeys one dominating condition (that is, internal movement or ocean current of a group) based on a time-control process.

The JF desired to be placed nearby food quantity.

Food is distributed to JF through a predetermined fitness function.

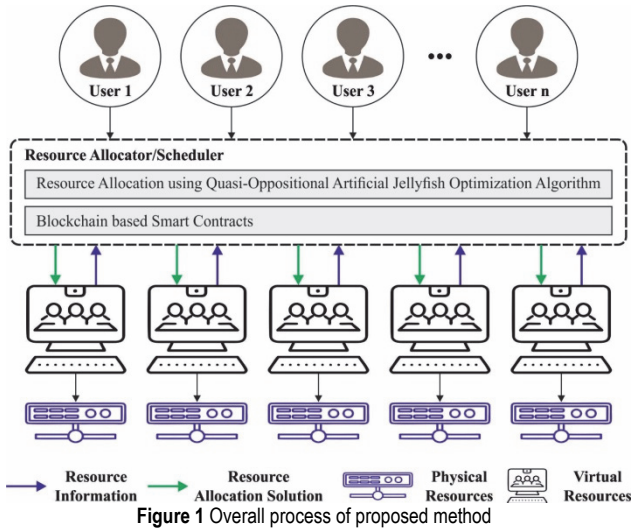

Figure 1 Overall process of proposed method

Once the JF moves inside a swarm, a bloom was generated due to active or passive movement. Food quantity varies with JF motion alongside a food-search direction. Afterward, the comparison among food quantities, the optimal values of fitness functions estimate the optimal position. The multiple phases from the JS optimizer are defined below. The population is initialized by using a logistic map:

$$\vec{P}_{i+1} = \theta \cdot \vec{P}_i\left(1-\vec{P}_i\right), 0 \le P_0 < 1 \tag{1}$$

In Eq. (1), the logistic value of i^{th} JF location is represented as $\vec{P}$. Effective implementation can be attained when the ∂ value is equivalent to four. The mathematical expression of the ocean current is expressed in the following,

$$\vec{P}_i(s+1) = \vec{P}_i(s) + \overrightarrow{rand}_1 * \left(\vec{P}^* - \gamma * rand_2 * \mu\right) \tag{2}$$

Eq. (2), $\overrightarrow{rand}_1$ represents the trajectory random integer lies within [0, 1], * denotes the vector multiplication operator, γ indicates the distributed coefficient, $rand_2$ indicates a random integer in the interval of zero and one, μ indicates the average population. The movement of the JF is controlled using active and passive movements. If the JF moves within the current, a movement is selected as passive. Therefore, the novel location is defined as follows:

$$\vec{P}_i(s+1) = \vec{P}_i(s) + rand_3 * \rho * \left(X_b - Y_b\right) \tag{3}$$

In Eq. (3), $rand_3$ indicates a random integer lies in the interval [0, 1], ρ denotes the moving distance from the existing location, X_b and Y_b denote the upper and lower limits of the searching space, correspondingly [18]. The novel location is existing in consecutive forms. Active movement can be described using Eq. (4):

$$\vec{P}_i(s+1) = \vec{P}_i(s) + \overrightarrow{rand}_1 * \vec{M} \tag{4}$$

Let $\vec{M}$ be the direction of movements formulated as follows:

$$\vec{M} = \begin{cases} \vec{P}_i(s) - \vec{P}_j(s), \\ \text{if fitness function}\left(\vec{P}_i\right) < \text{fitness function}\left(\vec{P}_j\right), \\ \overrightarrow{P_j}(s) - \vec{P}_i(s), \text{ or else.} \end{cases} \tag{5}$$

The ocean current, active and passive movements, are alternated by the time-control process, (s). In the following, the mathematical process is given in detail

$$C(s) = \left(1 - \frac{s}{S_{\max}}\right) * \left(2 * rand - 1\right) \tag{6}$$

It is noted that as time proceed, every JF continue to move inside the swarm to search for optimal food position. The major step of the AJFOA technique is demonstrated in Algorithm 1.

Algorithm 1:Pseudocode of AJFOA

Input ← objective function $f(P)$, population size (N_{pop}) searching space $[X_b : Y_b]$ Max count of iterations $\left(Max_{\text{int}}\right)$

Output ← an optimum outcome and visualization (JF bloom)

Begin
Determine objective function $F(P)$
Set the searching space, population size N_{pop}
Max count of iterations Max_{int}
Initializing population of JFs x_i
Compute the food at all the locations
DefineJF with optimum place
Initializing time: s = 1
while s < $s$max
Fitness estimation of all the iterations (solutions)
For i = 1: N_{pop}
Estimate the time control, $C(s)$, utilizing in Eq. (6)
If $c(t)$ >= 0.5
JF follow ocean current
else
JF moved inside a swarm
If Rand [0:1] >= $C(s)$
JF moved passively
else
JF moved actively intheir direction
End if
End if
End for
upgradea novel place to JF
Verify novel bound condition
Verify end criteria
Output the optimum outcomes and visualization (JF bloom)
End while
End

Tizhoosh developed Opposition-based learning (OBL). The discoverer of OBL says that a number opposite was possibly nearer than an arbitrary number to solutions [19]. By incorporating OBL in traditional evolutionary approaches, one might raise the coverage of solution space important to faster convergence and increased accuracy. Here, the researchers integrated QOBL in AJFOA technique that offered additional possibilities for finding solutions nearby global optimal. Firstly, the idea of QOBL is presented and later it is applied to accelerate convergence of AJFOA technique. The concept of QOBL is utilized in generation jumping and initialized populations. There exist some definitions utilized in OBL, as follows:

Opposite number: consider $x\varepsilon[a,b]$ represents a real number. The opposite number x^* can be determined as follows:

$$X^* = a + b - x \tag{7}$$

Opposite point: Let $X = (x_1, x_2, \ldots, x_s)$ refers to a point in S dimension space, in which $x_1, x_2, \ldots, x_s \varepsilon Y$ and $x_j \varepsilon [a_j, b_j], j\varepsilon 1,2,\ldots,S.$ The opposite point $X^* = (x_1^*, x_2^*, \ldots, x_s^*)$ is determined as follows:

$$x_i^* = a_j + b_j - x_j \tag{8}$$

Quasi-opposite number: consider x to represent real numbers within $[a,b]$. Its quasi-opposite number, x_{qo} is determined using Eq. (9):

$$\chi_{qo} = rand\left(c, x^*\right) \tag{9}$$

where c is represented as:

$$c = \frac{a+b}{2}$$

Quasi-opposite point: where x represents real numbers lies within $[a,b]$. Then, quasi-opposite point x_i^{q0} is determined by:

$$\chi_i^{qo} = rand\left(c_i, x_i^0\right) \tag{10}$$

where $c_i = \frac{a_i + b_i}{2}, x_i \varepsilon [a_i, b_i; i = \{1,2,\ldots,d\}.$

3.2 Application of QO-AJFOA for Resource Scheduling

In this study, the presented QO-AJFOA model majorly aims to optimally allocate the resources related to CP and bandwidth of every server for maximizing the long-term utility of the devices [20]. In this architecture, the objective is to improve the CP and width of MEC server to exploit the long-term utilities of each mobile device as follows,

$$P: \max_{B_{N,M}, F_{N,M}} \lim_{K\to+\infty} \frac{1}{K} \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{n=1}^{N_m} U_{n,m}(k) \tag{11}$$

$$\text{s.t. } C1: \sum_{n=1}^{N_m} (f_{n,m}(k) t_{n,m}^{\text{comp}}(t)) \le F_m, \forall m \in M,$$

$$C2: \sum_{n=1}^{N_m} b_{n,m}(k) \le B_m, \forall m \in M,$$

$$C3: C_{n,m}(k) \le G_n, \forall m \in M, n \in N_m,$$

$$C4: \frac{D_n / l}{\min_{\forall n} \{t_{n,m}^{\text{span}} + t_{n,m}^{\min e}(k) + t_{n,m}^{\text{prop}}\}} \ge \Omega,$$

$$C5: d_{n,m}^2 + (v_n t_{n,m}^{\text{mine}}(k))^2 - 2 d_{n,m} v_n t_{n,m}^{\text{mine}}(k) \cos\rho_{n,m} \le \omega^2,$$

$$\forall m \in M, n \in N_m,$$

where $B_{N,M} = \left\{ b_{n,m}(k) \mid b_{n,m}(k) \in [b_{\min}, b_{\max}] \right\}$ $k = 1,\ldots,K, m = 1,\ldots,M, n = 1,\ldots,N_m$, and $F_{N,M} = \left\{ f_{n,m}(k) \mid f_{n,m}(k) \in (f_{\min}, f_{\max}] \right\} k = 1,\ldots,K,$ $m = 1,\ldots,M, n = 1,\ldots,N_m$ are decision parameters that are the CP and allocation bandwidth of each device at all decision epochs, where $b_{\min}, b_{\max}, f_{\min}$ and $f_{\max}$ denote the lower and upper limits of CP and allocation bandwidth for mobile devices, correspondingly. Cl Constraint restricts the CP and allocation bandwidth of all the devices from single MEC server could not exceed their overall CP at all the decision epochs, and C2 constraint limits the allocation bandwidth of devices from single MEC server could not exceed their overall bandwidth. As assured in constraint, the mining costs of all the device is just mining budget, C3. C4 Constraint guarantees the BC throughput, that is, the amount of processing transactions for each second, in which l represents the average data size of transaction, $t_{n,m}^{\text{span}}$ denotes the timespan from the latter effectively mined blocks to the time once device n begins mining, and Ω indicates the lower limit of the BC throughput. $d_{n,m}$ indicates the distance among MEC server m and device $n, \rho_{n,m}$ denotes the angle between the moving device direction and that of devices n to MEC server m, ω indicates the radius of transmission range of SBS. C5 Constraint guarantees the mining tasks of all the devices are completed by single MEC server; that is, there is no transmission of MEC server to resolve the mining tasks of every device.

3.3 Block Chain Based Smart Contracts

In this work, a BC assisted Smart Contract protocol is used to distribute resource allocation, enforcing the clients to reach an agreement on the wireless channel utilization. BC is a series of pre-determined data types that poses a number of encrypted transactions in a distributed ledger [21]. To receive or send messages that comprise transactions (or other kinds of dataset), a user must be linked to the BC and need to download local copies of BC. From those measures, the robustness against an overall breakdown of single node is exploited. Alternatively, bare copy will be easier for manipulating, for example, by

automatically changing the entries of individual blocks. Because of the resultant disagreements among individual users, no consensus will be accomplished. Therefore, a BC has to present a mechanism that guarantees consensus-based legitimization and the authorized initiation of transactions within the network.

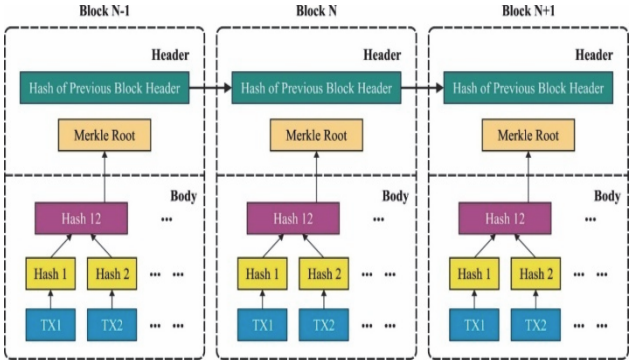

Figure 2 Structure of block chain

Furthermore, in the network to permit the secure management of function calls or complex conditions, a sophisticated protocol should be considered. Excepting a number of transactions, all the blocks possess a block header comprising obligatory information. At present, a considerable amount of header applications are presented. Nonetheless, a smaller number of header entries is recognized that is necessary to function the BC. Especially, for organizing the BC in consecutive order, it is mandatory to add a reference address on the header pointing at the preceding block. To prevent consensus and timing problems, a timestamp and a version number are also taken into account. To guarantee the trustworthiness of each transaction, a digital signature is generally produced by an accumulated concatenation of hash function, for example, a Merkle (Patricia) tree; also store the resultant hash inside the header. Fig. 2 demonstrates the framework of BC.

Generally, the concept of Smart Contract referred to an expanded function of BC, which forces every user to implement an equivalent number of code fragments, for example, the data exchange as function parameter. To run and invoke this code in detail, an additional type of message is taken into account. For Ethereum, the field "data" and "init" are utilized for exchanging input data and initially running code, correspondingly. For exchanging code specific values or datatypes, a specific kind of SmartContract- message is considered, and beneficial to transmit by running code segments of other users. Therefore, this type of message is equal to function calls. Out of the defined behaviors of Smart Contract, the mining tasks have to encompass the execution of code and the generation of blocks. Thus, further mechanisms are needed considering a payment that the transaction originator has to be paid to the miner, for example, Gas in the Ethereum BC. To obtain a consensus on resource sharing without central control for concurrently functioned WPN, we present a Smart-Contract-based technique. When registering to the BC, all the participants get a reasonable share of the available resource. When this amount is scarce, for example, long-term changed channel condition. Other users who possess surplus resources are capable of sharing part of the resource as a potential result of direct negotiation.

4 PERFORMANCE VALIDATION

In this section, the experimental validation of the QO-AJFOA algorithm is tested using a series of simulations taking place under varying number of tasks. Tab. 1 and Fig. 3 offer a comparative response time (REST) inspection of the QO-AJFOA model with existing models under different numbers of tasks [22]. The outcomes assured that the QO-AJFOA system has depicted proficient results with minimal values of REST. For instance, on 100 tasks, the QO-AJFOA model has offered reduced REST of 114 ms whereas the ICSA, CSRSA, and EERS-CEPO models have resulted in increased REST of 573 ms, 515 ms, and 271ms respectively. Along with that, on 300 tasks, the QO-AJFOA model has reached lower REST of 826ms whereas the ICSA, CSRSA, and EERS-CEPO models have accomplished higher REST of 1258 ms, 1189 ms, and 1004 ms respectively. Moreover, on 600 tasks, the QO-AJFOA model has demonstrated minimal REST of 1797 ms whereas the ICSA, CSRSA, and EERS-CEPO models have exhibited maximum REST of 2271 ms, 2106 ms, and 1893 ms respectively.

Table 1 REST analysis of QO-AJFOA algorithm with existing methodologies under distinct count of tasks

Response Time / ms				
Number of Tasks	ICSA	CSRSA	EERS-CEPO	QO-AJFOA
100	573	515	271	114
200	1128	1017	631	550
300	1258	1189	1004	826
400	1563	1432	1295	1097
500	1957	1777	1526	1363
600	2271	2106	1893	1797

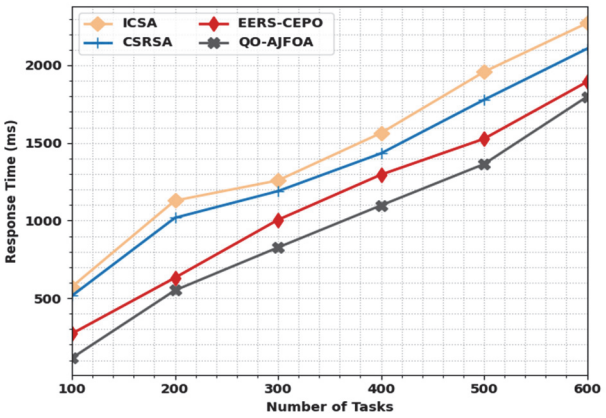

Figure 3 REST analysis of QO-AJFOA approach with distinct count of tasks

Tab. 2 and Fig. 4 present a comparative REST inspection of the QO-AJFOA system with recent models under distinct number of iterations. The outcomes assured that the QO-AJFOA system has shown proficient results with minimal values of REST. For instance, on 50 iterations, the QO-AJFOA approach has rendered reduced REST of 1414 ms whereas the ICSA, CSRSA, and EERS-CEPO algorithms have resulted in increased REST of 2160 ms, 1825 ms, and 1584 ms correspondingly. Additionally, on 150 iterations, the QO-AJFOA method has gained lower REST of 586 ms whereas the ICSA, CSRSA, and EERS-CEPO approaches have accomplished higher REST of 881 ms, 748 ms, and 673 ms correspondingly. Also, on 250 iterations, the QO-AJFOA method has demonstrated minimal REST of 494 ms whereas the ICSA, CSRSA, and

EERS-CEPO methodologies have exhibited maximum REST of 840 ms, 812 ms, and 621 ms correspondingly.

Table 2 REST analysis of QO-AJFOA algorithm with existing methodologies under distinct count of iterations

Response Time / ms				
Number of Iterations	ICSA	CSRSA	EERS-CEPO	QO-AJFOA
50	2160	1825	1584	1414
100	1632	1113	931	802
150	881	748	673	586
200	801	761	669	545
250	840	812	621	494

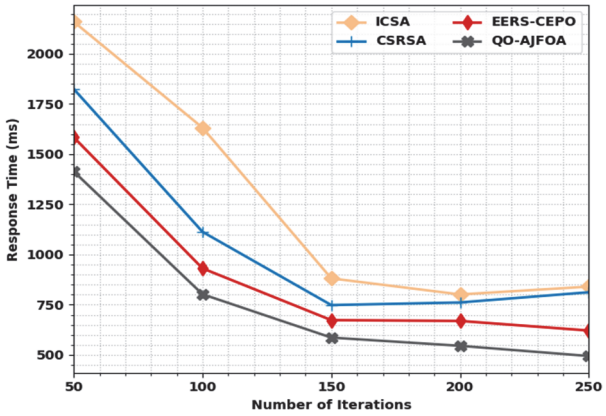

Figure 4 REST analysis of QO-AJFOA approach with distinct count of iterations

Table 3 Execution time analysis of QO-AJFOA approach with recent methods under distinct count of tasks

Execution Time / ms				
Number of Tasks	ICSA	CSRSA	EERS-CEPO	QO-AJFOA
100	2890	2785	2190	2011
200	3694	3535	3252	3060
300	4892	4376	3950	3818
400	5142	5030	4393	3941
500	5642	5458	4967	4857
600	6316	5963	5406	5128

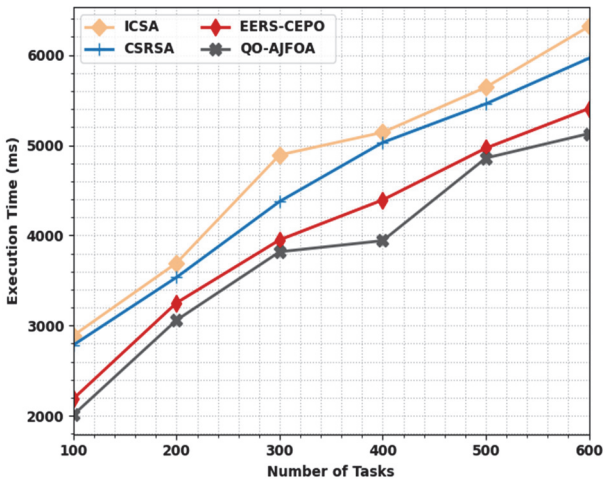

Figure 5 EXT analysis of QO-AJFOA approach with distinct count of tasks

Tab. 3 and Fig. 5 provide a comparative execution time (EXT) analysis of the QO-AJFOA approach with existing models under different numbers of tasks. The outcomes assured that the QO-AJFOA system has shown proficient results with minimal values of EXT. For instance, on 100 tasks, the QO-AJFOA algorithm has offered reduced EXT of 2011 ms whereas the ICSA, CSRSA, and EERS-CEPO models have resulted in increased EXT of 2890 ms, 2785 ms, and 2190 ms correspondingly. Also, on 300 tasks, the QO-AJFOA approach has reached lower EXT of 3818 ms whereas the ICSA, CSRSA, and EERS-CEPO models have accomplished higher EXT of 4892 ms, 4376 ms, and 3950 ms correspondingly. Furthermore, on 600 tasks, the QO-AJFOA methodology has demonstrated minimal EXT of 5128 ms whereas the ICSA, CSRSA, and EERS-CEPO models have exhibited maximum EXT of 6316 ms, 5963 ms, and 5406 ms correspondingly.

Tab. 4 and Fig. 6 present a comparative energy consumption (ECM) scrutiny of the QO-AJFOA approach with existing models under distinct number of scheduling cycles (SCs). The results assured that the QO-AJFOA algorithm has shown proficient results with minimal values of ECM. For instance, on 1 SC, the QO-AJFOA approach has rendered reduced ECM of 1.16 KWh whereas the ICSA, CSRSA, and EERS-CEPO techniques have resulted in increased ECM of 1.94 KWh, 1.62 KWh, and 1.62 KWh correspondingly. Also, on 3 SCs, the QO-AJFOA model has reached lower ECM of 1.13 KWh whereas the ICSA, CSRSA, and EERS-CEPO approaches have accomplished higher ECM of 2.21 KWh, 2.02 KWh, and 1.14 KWh respectively. Moreover, on 5 SCs, the QO-AJFOA method has demonstrated minimal ECM of 0.98KWh whereas the ICSA, CSRSA, and EERS-CEPO models have exhibited maximum ECM of 1.7 KWh, 1.35 KWh, and 2.06 KWh correspondingly.

Table 4 ECM analysis of QO-AJFOA approach with recent methods under distinct count of scheduling cycles

Energy Consumption / KWh				
Scheduling Cycle	ICSA	CSRSA	EERS-CEPO	QO-AJFOA
1	1.94	1.62	1.62	1.16
2	2.27	1.86	1.80	1.34
3	2.21	2.02	1.44	1.13
4	2.03	2.27	2.22	1.72
5	1.7	1.35	2.06	0.98

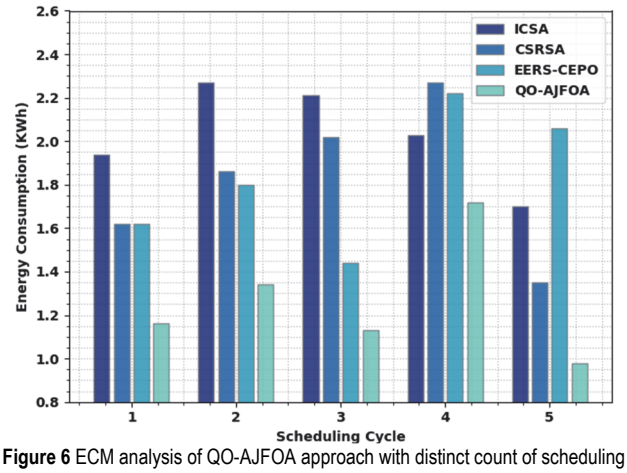

Figure 6 ECM analysis of QO-AJFOA approach with distinct count of scheduling cycles

Tab. 5 and Fig. 7 grant a comparative average ECM (AECM) review of the QO-AJFOA algorithm with existing models under different numbers of tasks. The results assured that the QO-AJFOA technique has shown proficient results with minimal values of AECM.

For instance, on 100 tasks, the QO-AJFOA methodology has provided reduced AECM of 1.55kJ whereas the ICSA, CSRSA, and EERS-CEPO models have resulted in increased AECM of 2.79 kJ, 2.43 kJ, and 1.99

kJ correspondingly. Also, on 300 tasks, the QO-AJFOA method has reached lower AECM of 3.44 kJ whereas the ICSA, CSRSA, and EERS-CEPO methods have accomplished higher AECM of 4.48 kJ, 4.23 kJ, and 3.78 kJ correspondingly. Besides, on 600 tasks, the QO-AJFOA model has demonstrated minimal AECM of 7.65 kJ whereas the ICSA, CSRSA, and EERS-CEPO techniques have exhibited maximum AECM of 9.99 kJ, 8.09 kJ, and 7.95 kJ correspondingly.

Table 5 AECM analysis of QO-AJFOA approach with recent methods under distinct count of tasks

Average Energy Consumption / kJ				
Number of Tasks	ICSA	CSRSA	EERS-CEPO	QO-AJFOA
100	2.79	2.43	1.99	1.55
200	3.34	3.23	2.87	2.47
300	4.48	4.23	3.78	3.44
400	6.29	5.30	4.84	4.46
500	8.50	6.81	6.19	5.76
600	9.99	8.09	7.95	7.65

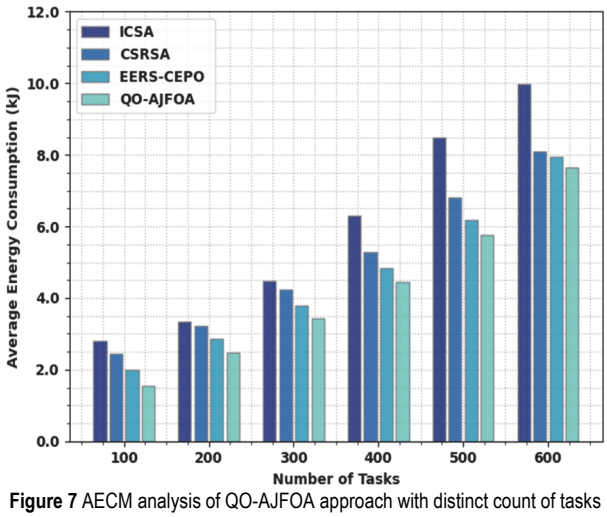

Figure 7 AECM analysis of QO-AJFOA approach with distinct count of tasks

Table 6 AEP analysis of QO-AJFOA approach with recent methods under distinct count of tasks

Average Executive Power / W				
Number of Tasks	ICSA	CSRSA	EERS-CEPO	QO-AJFOA
100	1971	1752	1446	1358
200	2379	2197	1993	1815
300	3163	2955	2753	2668
400	3867	3622	3386	3242
500	4588	4253	3842	3690
600	5783	4751	4365	4258

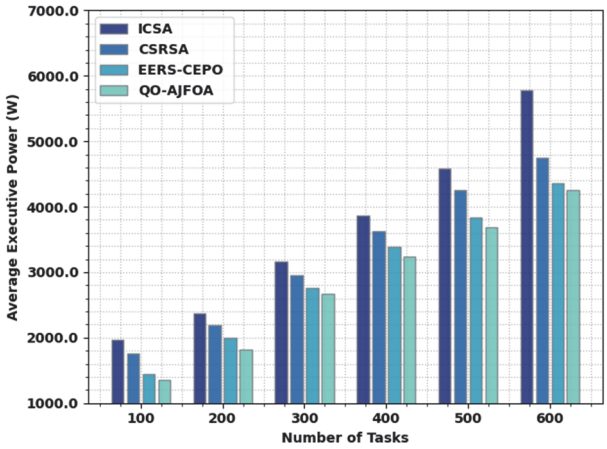

Figure 8 AEP analysis of QO-AJFOA approach with distinct count of tasks

Tab. 6 and Fig. 8 provide a brief average executive power (AEP) check of the QO-AJFOA methodology with existing models under different numbers of tasks. The outcomes assured that the QO-AJFOA method has shown proficient results with minimal values of AEP. For instance, on 100 tasks, the QO-AJFOA model has presented decreased AEP of 1358 W while the ICSA, CSRSA, and EERS-CEPO models have resulted in superior AEP of 1971 W, 17525 W, and 1446 W correspondingly. Moreover, on 300 tasks, the QO-AJFOA approach has gained decreased AEP of 2668 W while the ICSA, CSRSA, and EERS-CEPO methods have accomplished higher AEP of 3163 W, 2955 W, and 2753 W correspondingly. Moreover, on 600 tasks, the QO-AJFOA approach has demonstrated minimal AEP of 4258 W whereas the ICSA, CSRSA, and EERS-CEPO models have exhibited maximum AEP of 5783 W, 4751 W, and 4365 W correspondingly.

5 CONCLUSION

In this research, an effective QO-AJFOA technique was established for RS in the CC environment. This research focuses on the development of a novel quasi-oppositional artificial jellyfish optimization algorithm (QO-AJFOA) for resource scheduling (RS) in the cloud computing (CC) environment. The proposed QO-AJFOA aims to optimize the allocation of computing power and bandwidth resources of servers to maximize the long-term utility of devices. The technique combines quasi-oppositional based learning (QOBL) with the traditional AJFOA.The presented QO-AJFOA model majorly aims to optimally allocate the resources related to CP and bandwidth of every server for maximizing the long-term utility of the devices. In addition, the QO-AJFOA technique involves the integration of the QOBL concept into the traditional AJFOA. Moreover, a BC assisted Smart Contract protocol is used to distribute resource allocation, enforcing the clients to reach an agreement on the wireless channel utilization. The experimental validation of the QO-AJFOA technique is tested under varying number of tasks and iterations. The wide-ranging result analysis shows the promising performance of the QO-AJFOA technique over recent models. Thus, the QO-AJFOA technique is exploited to effectually schedule the resources in the CC environment.

6 REFERENCES

[1] Zhang, H., Tong, L., Yu, J., & Lin, J. (2021). Block chain-Aided Privacy-Preserving Outsourcing Algorithms of Bilinear Pairings for Internet of Things Devices. *IEEE Internet of Things Journal*, *8*(20), 15596-15607. https://doi.org/10.1109/JIOT.2021.3073500

[2] Wei, S., Dunbing, T., & Ping, Z. (2021). Research on Cloud Enterprise Resource Integration and Scheduling Technology Based on Mixed Set Programming. *Technical Gazette*, *28*(6), 2027-2035. https://doi.org/10.17559/TV-20210718091658

[3] Baranwal, G., Kumar, D., & Vidyarthi, D. P. (2022). BARA: A blockchain-aided auction-based resource allocation in edge computing enabled industrial internet of things. *Future Generation Computer Systems*, *135*, 333-347. https://doi.org/10.1016/j.future.2022.05.007

[4] Li, Y. & Zhanyong W. (2022). A Cloud Based Network Intrusion Detection System. *Technical Gazette*, *29*(3), 987-992. https://doi.org/10.17559/TV-20211130024245

[5] Prabadevi, B., Deepa, N., Pham, Q. V., Nguyen, D. C., Reddy, T., Pathirana, P. N., & Dobre, O. (2021). Toward block chain for edge-of-things: a new paradigm, opportunities, and future directions. *IEEE Internet of Things Magazine*, *4*(2), 102-108. https://doi.org/10.1109/IOTM.0001.2000191

[6] Zhu, K., Huang, L., Nie, J., Zhang, Y., Xiong, Z., Dai, H. N., & Jin, J. (2022). Privacy-Aware Double Auction with Time-Dependent Valuation for Block chain-based Dynamic Spectrum Sharing in IoT Systems. *IEEE Internet of Things Journal*, 1-1. https://doi.org/10.1109/JIOT.2022.3165819

[7] Gopinath, S., Vinoth Kumar, K., Elayaraja, P., Parameswari, A., Balakrishnan, S., & Thiruppathi, M. (2021). SCEER: Secure cluster based efficient energy routing scheme for wireless sensor networks. *Materials Today: Proceedings*, 45(2), 3579-3584. https://doi.org/10.1016/j.matpr.2020.12.1096

[8] Saad, A. & Sooyong, P. (2023). A Decentralized Lightweight Blockchain Nodes Architecture Based on a Secure OpenFlow Protocol Controller Channel. *Technical Gazette*, 30(1), 114-121. https://doi.org/10.17559/TV-20220427051644

[9] Yin, J., Xiao, Y., Pei, Q., Ju, Y., Liu, L., Xiao, M., & Wu, C. (2022). Smart DID: A Novel Privacy-preserving Identity based on Blockchain for IoT. *IEEE Internet of Things Journal*, 1-1. https://doi.org/10.1109/JIOT.2022.3145089

[10] Almagrabi, A. O. & Bashir, A. K. (2021). A classification-based privacy-preserving decision-making for secure data sharing in Internet of Things assisted applications. *Digital Communications and Networks*, *8*(4), 436-445. https://doi.org/10.1016/j.dcan.2021.09.003

[11] Firoozjaei, M. D., Lu, R., & Ghorbani, A. A. (2020). An evaluation framework for privacy-preserving solutions applicable for block chain-based internet-of-things platforms. *Security and Privacy*, *3*(6), 131. https://doi.org/10.1002/spy2.131

[12] Gai, K., Wu, Y., Zhu, L., Zhang, Z., & Qiu, M. (2019). Differential privacy-based blockchain for industrial internet-of-things. *IEEE Transactions on Industrial Informatics*, *16*(6), 4156-4165. https://doi.org/10.1109/TII.2019.2948094

[13] Gai, K., Wu, Y., Zhu, L., Xu, L., & Zhang, Y. (2019). Permissioned blockchain and edge computing empowered privacy-preserving smart grid networks. *IEEE Internet of Things Journal*, *6*(5), 7992-8004. https://doi.org/10.1109/JIOT.2019.2904303

[14] Qiu, C., Yao, H., Jiang, C., Guo, S., & Xu, F. (2019). Cloud computing assisted blockchain-enabled Internet of Things. *IEEE Transactions on Cloud Computing*, *10*(1), 247-257. https://doi.org/10.1109/TCC.2019.2930259

[15] Li, H., Han, D., & Tang, M. (2021). A Privacy-Preserving Storage Scheme for Logistics Data with Assistance of Blockchain. *IEEE Internet of Things Journal*, *9*(6), 4704-4720. https://doi.org/10.1109/JIOT.2021.3107846

[16] Medhane, D. V., Sangaiah, A. K., Hossain, M. S., Muhammad, G., & Wang, J. (2020). Blockchain-enabled distributed security framework for next-generation IoT: an edge cloud and software-defined network-integrated approach. *IEEE Internet of Things Journal*, *7*(7), 6143-6149. https://doi.org/10.1109/JIOT.2020.2977196

[17] Li, H., Han, D., & Tang, M. (2020). A privacy-preserving charging scheme for electric vehicles using blockchain and fog computing. *IEEE Systems Journal*, *15*(3), 3189-3200. https://doi.org/10.1109/JSYST.2020.3009447

[18] Chou, J. S. & Truong, D. N. (2021). A novel metaheuristic optimizer inspired by behavior of jellyfish in ocean. *Applied Mathematics and Computation*, *389*, 125535. https://doi.org/10.1016/j.amc.2020.125535

[19] Elkabbash, E. T., Mostafa, R. R., & Barakat, S. I. (2021). Android malware classification based on random vector functional link and artificial Jellyfish Search optimizer. *PloS one*, *16*(11), 0260232. https://doi.org/10.1371/journal.pone.0260232

[20] Houssein, E. H., Mahdy, M. A., Fathy, A., & Rezk, H. (2021). A modified Marine Predator Algorithm based on opposition based learning for tracking the global MPP of shaded PV system. *Expert Systems with Applications*, *183*, p.115253. https://doi.org/10.1016/j.eswa.2021.115253

[21] Gopinath, S., Vinoth Kumar, K., Elayaraja, P., Parameswari, A., Balakrishnan, S., & Thiruppathi, M. (2021). SCEER: Secure cluster based efficient energy routing scheme for wireless sensor networks. *Materials Today: Proceedings*, *45*(2), 3579-3584. https://doi.org/10.1016/j.matpr.2020.12.1096

[22] Ning, Z., Sun, S., Wang, X., Guo, L., Wang, G., Gao, X., & Kwok, R. Y. (2021). Intelligent resource allocation in mobile blockchain for privacy and security transactions: a deep reinforcement learning based approach. *Science China Information Sciences*, *64*(6), 1-16. https://doi.org/10.1007/s11432-020-3125-y

[23] Haoyang, D. & Junhui C. (2023). An Improved Ant Colony Algorithm for New energy Industry Resource Allocation in Cloud Environment. *Technical Gazette*, *30*(1), 153-157. https://doi.org/10.17559/TV-20220712164019

[24] Vinoth Kumar, K. & Balaganesh, D. (2022). Efficient Privacy-Preserving Red Deer Optimization Algorithm with Blockchain Technology for Clustered VANET. *Technical Gazette*, *29*(3), 813-817. https://doi.org/10.17559/TV-202112161156350

[25] Mansour, R. F., Alhumyani, H., Khalek, S. A., Saeed, R. A., & Gupta, D. (2022). Design of cultural emperor penguin optimizer for energy-efficient resource scheduling in green cloud computing environment. *Cluster Computing*, *26*(1), 575-586. https://doi.org/10.1007%2Fs10586-022-03608-0

Contact information:

Mr. Raja RAMAMOORTHI, Assistant Professor
Department of Computer Applications,
Anjalai Ammal Mahalingam Engineering College,
Kovilvenni, Chennai
E-mail: miltonraja@gmail.com

Dr. Arunprakash RAMASAMY, Assistant Professor (Sr. Gr)
(Corresponding Author)
Department of Computer Science and Engineering,
University College of Engineering - Ariyalur,
Tamilnadu, India
E-mail: arunitvijay@gmail.com