Received: November 29, 2021. Revised: December 28, 2021.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Effective Load Balancing and Security in Cloud using Modified Particle Swarm Optimization Technique and Enhanced Elliptic Curve Cryptography Algorithm

Santhosh Kumar Gorva[1]* **Latha Channagiri Anandachar[2]**

[1]*Don Bosco Institute of Technology, Bengaluru, India*
[2]*RV Institute of Technology and Management, Bengaluru, India*
* Corresponding author's Email: gsantosh905@gmail.com

Abstract: In recent decades, the rapid growth of cloud computing has led to increasing concerns about the security and energy requirement of cloud data centers. To overcome the concerns of load balancing and data security, a new hybrid model is proposed in this article. Firstly, a Modified Particle Swarm Optimization (MPSO) technique is proposed for balancing the tasks between heavy-loaded Virtual Machine (VMs) and light-loaded VMs. An effective load balancing improves the performance and availability of website, database, application and other computing resources. In the MPSO, a linear decreasing inertia weight is included to achieve optimal solutions in both VM migrations and load balancing. Secondly, an Enhanced Elliptic Curve Cryptography (EECC) algorithm is proposed for effective data security. In the proposed EECC algorithm, a new pseudo-random key is combined with a public key to improve the security of cloud data centers. In this work, the effectiveness of the proposed MPSO-EECC model is validated in terms of Service Level Agreement (SLA), execution time, energy consumption, energy SLA violation, encryption, and decryption comparison time. As represented in the experimental section, the proposed MPSO technique almost reduced 30%-70% of energy consumption compared to the existing optimization techniques. Similarly, the proposed EECC algorithm reduced 10ms to 30ms of decryption comparison time related to the comparative cryptography algorithms.

Keywords: Cloud computing, Enhanced elliptic curve cryptography, Modified particle swarm optimization, Security, Virtual machine.

1. Introduction

In recent times, cloud computing is an effective computing archetype that delivers several services on demand at a minimum cost [1]. The cloud computing environment gave a new direction to information technology in terms of remote data sharing, multi-tenancy, and resource sharing [2]. The primary objective of a cloud computing platform is to provide more data storage, fast and easy use the computing services [3, 4]. However, the unpredictable and unstable rate of client requests and alternating resource usages of VMs leads to an unbalanced load that results in SLA violation and performance degradation [5, 6]. Additionally, the unbalanced cloud servers cause increased power consumption, poor resource utilization, increased response time, and decreased throughput [7]. So, an effective load balancing methodology is required for balancing the load of cloud servers, which improves the overall performance of the cloud servers [8]. In addition to this, the outsourced data are the crucial entities in the cloud computing environment that suffers from numerous attacks and threats by exploring the vulnerabilities available in the cloud modules [9, 10]. The attackers can be an outsider (mischievous hackers) or insider (cloud service providers), who wants to access the owner's data that is used for some gain [11, 12]. To protect the outsourced cloud data, a significant cryptography or encryption method is required in this application [13, 14]. To overcome the issues of load balancing and outsourced cloud data security, a new model is developed in this article. The main contributions of this study are listed as follows:

- Developed MPSO technique to effectively balance the heavy-loaded VMs and light-loaded VMs for creating a load balance among VMs. The developed MPSO technique superiorly reduces the power consumption and increases the system resource usages.
- Developed EECC algorithm generates pseudo-random keys with a public key for outsourced data encryption and decryption, which effectively ensures the privacy and integrity of cloud data.
- The efficiency of the proposed MPSO-EECC model is investigated in light of SLA, energy consumption, energy SLA violations, encryption comparison time, decryption comparison time, and execution time on two types of scenarios.

This paper is organized as follows: Some existing research papers related to “load balancing and security in cloud computing” are surveyed in Section 2. The detailed explanation and the experimental results of the proposed MPSO-EECC model are denoted in Sections 3 and 4. The conclusion of the present work is stated in Section 5.

2. Related works

Ibrahim [15] presented a new Power Aware-Particle Swarm Optimization (PAPSO) technique to find the optimal placement of VMs in the cloud. The PAPSO technique was developed based on decimal encoding that maps the migrated VMs to the appropriate Physical Machines (PMs). Additionally, an effective fitness function was used in the PAPSO technique that reduces energy consumption without violating Service Level Agreement (SLA). Balaji [16] developed an adaptive cat swarm optimization technique for load balancing in the cloud. In this literature, the effectiveness of the presented technique was investigated utilizing different performance measures and compared to the state-of-the-art techniques. Traditional population based stochastic techniques, PSO, and cat swarm optimization techniques cannot always produce the optimal solutions in load balancing and VM migration. Mapetu [17] presented a dynamic technique for reducing the trade-off between VM migrations, SLA violations, and energy consumption. In this study, the binary PSO technique was applied for load balancing that impact a number of host shutdown and energy consumption. Additionally, the Pearson correlation coefficient was used for VM migration, and the experimental outcomes showed that the developed technique effectively outperformed the benchmark and existing techniques using random workloads and PlanetLab. However, the optimal solutions cannot be guaranteed in a binary PSO technique, and also the number of VM migrations and the number of host showdowns was ineffective in this study.

Jena [18] combined an improved Q-learning algorithm and modified PSO technique for load balancing among the VMs in the cloud computing platform. Further, an intelligent load balancing technique was required for improving the execution time of assigned tasks. Namasudra [19] used identity based time release encryption, distributed hash table, and attribute based encryption techniques for secure and efficient access control in the cloud computing environment. Initially, the resources/data were encrypted by utilizing the user attributes, and then the encrypted data were divided into two types such as extracted cipher-text, and encapsulated cipher-text. In addition, the decryption key was encrypted using identity based time release encryption technique, and then combine the extracted cipher-text with the key cipher-text to create the cipher-text shares. Lastly, the cipher-text shares were distributed in the network, and then the encapsulated cipher-text were stored in the cloud servers. As stated in the resulting section, the computational complexity of the developed model was comparably higher related to the state of the art encryption techniques. Tahir [20] introduced a novel model: Crypto Genetic Algorithm (GA) to cope with privacy issues and data integrity. In this literature study, the GA was utilized to generate keys for data decryption and encryption that ensure the integrity and privacy of cloud data. The presented CryptoGA obtained better encryption performance by means of throughput and execution time. However, some of the major security attacks in the cloud were not discussed in this study.

Soltanshahi [21] applied krill herd optimization algorithm for VM allocation to the physical hosts in the cloud data centers. In this literature, the CloudSim simulator was used to analyze the effectiveness of the developed optimization algorithm and the experimental results showed that the developed algorithm effectively reduced energy consumption by 35%. However, this literature study didn’t concentrate on a major concern of data security in the cloud computing environment. El Makkaoui [22] used two fast variants of the cloud Paillier method for fast decryption. The 1st variant uses Chinese remainder theorem, and the 2nd variant uses the modified Paillier method for decryption. The theoretical results showed that the developed method delivers better decryption speed when preserving a security level. However, the

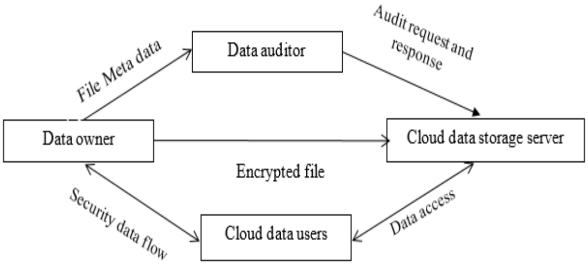

Figure. 1 Flowchart of a general cloud computing environment

Paillier method was insecure under the issue of decisional composite residuosity. To address the concerns of load balancing and security in the cloud, a new model named MPSO-EECC is introduced in this article.

3. Methodology

The main aim of the MPSO-EECC model is to minimize energy consumption, maximize the overall SLA and energy SLA percentages, and to improve the cloud data security by means of execution time. The flowchart of a cloud computing environment is graphically depicted in Fig. 1, and it includes four major components such as data auditor, data owner, cloud data users, and cloud data storage server.

- Data auditor evaluates the services provided by the cloud provider in light of privacy impact, security controls, etc.
- Data owner has the right to access and edit the data in cloud.
- Cloud data users, who use cloud computing services for accessing, storing and managing the computing services.
- Cloud data storage server is a cloud computing model that stores on the internet through a cloud computing provider.

3.1 Load balancing using MPSO technique

The two main objectives of this research article are load balancing using the MPSO technique and security using the EECC algorithm. In this work, the MPSO technique is used to allocate resources to the physical hosts by reducing energy consumption and computational complexity. The population based optimization algorithms like PSO deliver better solutions to the NP-hard problems such as VM allocation to the physical hosts. Related to other optimization algorithms, the proposed MPSO technique delivers a convenient solution with an efficient function. In the MPSO technique, the velocity vector indicates VMs, and the position vector represents PMs. The conventional PSO technique is a population based stochastic technique, which maintains two main populations such as a population of particles best position, and a population of particles current position [23, 24]. In the conventional PSO technique, every particle is related to two characteristics such as position vector X and velocity vector V. The particle moves in the dynamic search space with a velocity, which is adjusted based on the particle's companions experience and particles experience. The position and velocity of the particles are updated based on Eqs. (1) and (2).

$$v_{id}(t+1) = w \times v_{id}(t) + c_1 \times r_1 \times [p_{id}(t) - x_{id}(t)] + c_2 \times r_2 \times [p_{gd}(t) - x_{id}(t)] \quad (1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id} \quad (2)$$

where, $r_1\ and\ r_2$ indicates two random numbers, which are distributed between the range of (0, 1), and c_1 and c_2 states acceleration coefficients. The w indicates inertia weight, which is utilized to balance the local and global search. In the MPSO technique, a linear decreasing inertia weight is proposed for enhancing the fine-tuning properties of the conventional PSO technique. In this scenario, the inertia weight value w linearly decreased from w_{max} initial value to w_{min} the final value, while the iteration increases. The linear decreasing inertia weight is mathematically expressed in Eq. (3).

$$w(t) = \frac{t_{max}-t}{t_{max}}(w_{max} - w_{min}) + w_{min} \quad (3)$$

In a D dimensional vector space, the particle's position is denoted as $X_i = [x_{i1}, x_{i2}, . x_{ij} . x_{iD}]$, where, $x_{ij} \epsilon [x_{min}, x_{max}]$ denotes the i^{th} particle position in j^{th} dimension. Similarly, the particle's velocity is specified as $V_i = [v_{i1}, v_{i2}, . v_{ij} . v_{iD}]$, where $v_{ij} \epsilon [v_{min}, v_{max}]$ represents the i^{th} particle velocity in j^{th} dimension. The particle's best position, and the particle's global best position are defined as $P_i = (p_{i1}, p_{i2}, \dots, p_{ij}, \dots, p_{iD})$ and $P_g = (p_{g1}, p_{g2}, \dots, p_{gj}, \dots, p_{gD})$. The pseudocode of the MPSO technique is depicted below.

Pseudocode of MPSO technique

```
Initialization
    Initialize the particle swarms randomly
    While
    Calculate the fitness value of every particle swarm
    For (number of particles $n = 1$)
        Determine particle's current best position
        Determine particle's global best position
        For (number of particle dimensions $d = 1$)
            Update the particle's velocity using Eq. (1)
            Update the particle's position using Eq. (2)
        Next $d$
    Next $n$
    Update the linear decreasing inertia weight $w$ using Eq. (3)
  Next-generation will be processed after the stopping criterion met
End
```

3.2 Cloud data security using EECC algorithm

As mentioned earlier, the cloud computing platform provides computing services and storage to the users with low cost of internet, elasticity and high data availability without the need for local storage. In the cloud computing platform, the security related to outsourced data becomes a major issue for cloud users. In recent times, several encryption algorithms are developed to ensure outsourced data security, but the concern is that the cloud providers should have access to the out-sourced data to respond to the user's request. In this scenario, outsourced data security is a major concern, especially if the data is more sensitive and private. Computing on encrypted data is considered to be an efficient way to overcome the aforementioned concerns. In this article, an EECC algorithm is proposed for effective data security in the cloud computing environment. The conventional ECC algorithm is a modern crypto-system, which works based on algebraic structures (elliptic curve) over finite fields. The ECC algorithm implements all asymmetric crypto-system's capabilities such as key exchange, signatures, and encryption. On the other hand, the ECC algorithm is a modified version of Rivest, Shamir, Adleman (RSA) algorithm, where it uses small signatures and keys that provide fast signatures, fast key generation and fast key agreement [25, 26]. In the ECC algorithm, the private key is indicated as H and the prime number is stated as n_{pr}. The general formula of elliptic curve cubic is mathematically depicted in Eq. (4).

$$E = pr(i)^3 + m \times pr(i) + s \tag{4}$$

where, m and s are denoted as constant values and it is $m = s = 2$. The best point is selected from the elliptic curve, if the condition $A = B$ is satisfied. The values A and B are mathematically stated in Eqs. (5) and (6).

$$A = mod\ (E,\ n_{pr}) \tag{5}$$

$$B = mod\ ((pr(j))^2,\ n_{pr}) \tag{6}$$

where, n_{pr} states number of a prime number and $pr(i,j)$ states the elliptic curve points. In the ECC algorithm, the doubling mechanism is utilized for identifying the $A\ and\ B$ values. The public key p_f and the best point $p_e(k,l)$ are mathematically related in Eq. (7).

$$p_f = H \times p_e \tag{7}$$

In the EECC algorithm, a pseudo random key is combined with a public key p_f to further improve the security in the cloud platform. The pseudo random key generates and assigns a name for every block after data encryption, here the keys are generated from the encrypted data files. The assigned names cannot be determined by the attackers, cloud servers and users. The advance calculation on the blocks is performed by the users to avoid privacy leakage of data without affecting the features of third party auditors. The EECC algorithm significantly prevents privacy leakage and provides efficient privacy protection of public auditing compared to the ECC algorithm. In the key generation phase, the EECC algorithm generates random parameters and uses blind metadata that consists of several data blocks. In this scenario, the user transmits parameters to the servers, while third party auditor gives a request to the server for data authentication. The server estimates the proof by utilizing data blocks, random parameters, and metadata for better security.

3.2.1. Data encryption

In this scenario, each share has blocks and each block is encrypted using the EECC algorithm. The number of blocks is indicated as $b(i,j)$, where, i denotes number of rows in the blocks, and j denotes number of columns in the blocks. Here, two parts of the data $U_x(i,j)$ and $U_y(i+1,j)$ are given as the input for encryption mechanism, which is mathematically stated in Eqs. (8) and (9).

$$Z_1 = H \times p_s \times p_e \tag{8}$$

$$Z_2 = \left(U_x(i,j), U_y(i+1,j)\right) + Z_1 \tag{9}$$

where, p_s indicates generated pseudo random key.

3.2.2. Data decryption

In the decryption mechanism, the generated pseudo random key p_s and the private key H is used for decrypting the data. In this scenario, Z_{11} is applied for decrypting the data point, which is mathematically depicted in Eq. (10), and the final result of the decryption process is indicated in Eq. (11).

$$Z_{11} = H \times p_s \times Z_1 \tag{10}$$

$$Z_{ij} = Z_2 - Z_{11} \tag{11}$$

The quantitative performance of the proposed MPSO-EECC model is stated in the upcoming section in terms of load balancing and cloud security.

4. Experimental results

The proposed MPSO-EECC model performance is simulated using CloudSim tool, where it is utilized to model the system and to investigate the behavior of cloud computing elements like VMs, data centers, application delivery approaches and resource policies [27]. In this scenario, the effectiveness of the proposed MPSO-EECC model is validated by comparing its performance with the benchmark models like Genetic Algorithm (GA), Modified Best Fit Decreasing (MBFD) algorithm, PSO, Krill Herd optimization Algorithm (KHA) [21], cloud Paillier method [22], fast cloud Paillier method [22] and ECC algorithm. Hence, the efficiency of the proposed MPSO-EECC model is validated using SLA, energy consumption, energy SLA violation, execution time, encryption comparison time, and decryption comparison time. The performance metric: SLA is used for analyzing the level of service quality between the cloud service providers and the users. Usually, the SLA relies on how fast the responses are done to the requests or response time [28].

The energy SLA violation is used to measure the usage of VM allocation to the physical hosts. The overall behavior of energy SLA violation depends on the number of SLA contract violations and energy consumption [29] that is mathematically defined in Eq. (12). The energy consumption by PMs is precisely described by a linear relation of CPU usage. In this scenario, the PM consumes 70% of energy, while it is completely utilized.

$$Energy\ SLA\ violation = Energy\ \left(\frac{kw}{h}\right) \times SLA \tag{12}$$

4.1 Quantitative investigation related to loading balancing

In load balancing, the effectiveness of the proposed MPSO technique is validated by comparing its performance with GA, MBFD, KHA, and PSO using overall SLA, energy consumption, and overall energy SLA violation. Here, the quantitative investigation is done in two scenarios that are denoted in Table 1. The undertaken optimization techniques: GA, MBFD, KHA, PSO, and MPSO performance is investigated under over-loaded host detection techniques like Inter Quartile Range (IQR) and Median Absolute Deviation (MAD) and other VM selection techniques such as Random Selection (RS), and Minimum Migration Time (MMT).

*Million Instructions per Seconds (MIPS), Random Access Memory (RAM)

By investigating Table 2, the proposed MPSO technique obtained significant performance in load balancing by means of energy consumption on both scenarios. In Table 2, the proposed MPSO technique almost reduced 30% of energy consumption in scenario I related to comparative techniques: GA, MBFD, KHA [21] and PSO. In addition, the MPSO technique averagely reduced 30%-70% of energy consumption in scenario II related to other optimization techniques on the four cases: MMT/MAD, RS/MAD, MMT/IQR, and RS/IQR. The graphical presentation of the MPSO technique in light of energy consumption is depicted in Fig. 2.

As depicted in the Table 3 and 4, the proposed MPSO technique achieved high overall SLA, and

Table 1. Data specifications of scenario I and II

Scenario	**VM specifications**		
	MIPS	**CPU**	**RAM (MB)**
Scenario I	250-1000	1	128
Scenario II	500-2500	1	613-3840
Scenario	Data center specifications		
	Data center	Physical hosts	VMs/tasks
Scenario I	1	100	290
Scenario II	1	800	1175
Scenario	Physical hosts specifications		
	MIPS	CPU	RAM (MB)
Scenario I	1000-3000	1	8192
Scenario II	1860-2660	2	4096

Received: November 29, 2021. Revised: December 28, 2021.

Table 2. Performance investigation of MPSO technique in light of energy consumption

Energy consumption (W/H)								
	Scenario I				Scenario II			
Techniques	MMT/MAD	RS/MAD	MMT/IQR	RS/IQR	MMT/MAD	RS/MAD	MMT/IQR	RS/IQR
GA	88	90	70	63	444	508	1160	464
MBFD	61	51	79	47	452	460	354	424
KHA [21]	50	40	64	43	441	368	345	385
PSO	58	46	75	43	448	455	348	419
MPSO	49	38	56	41	438	362	342	371

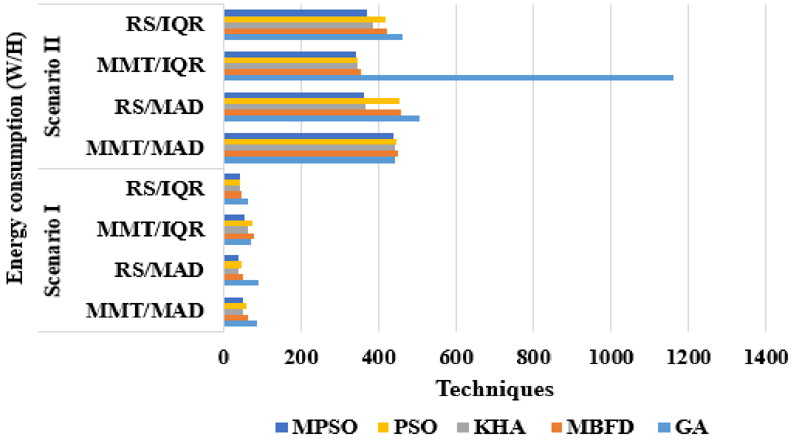

Figure. 2 Graphical presentation of MPSO technique in light of energy consumption

Table 3. Performance investigation of MPSO technique in light of overall SLA

Overall SLA (%)								
	Scenario I				Scenario II			
Techniques	MMT/MAD	RS/MAD	MMT/IQR	RS/IQR	MMT/MAD	RS/MAD	MMT/IQR	RS/IQR
GA	1.19	1.18	1.12	1.51	3.92	3.68	4.27	4.02
MBFD	2.08	2.38	2.31	1.8	3.57	3.64	4.51	3.7
KHA [21]	2.74	1.98	2.38	1.93	3.98	4.57	5.46	4.68
PSO	1.73	1.98	1.65	1.07	3.54	3.61	4.48	3.68
MPSO	3.32	3.48	3.12	3.03	4.1	4.58	5.52	4.7

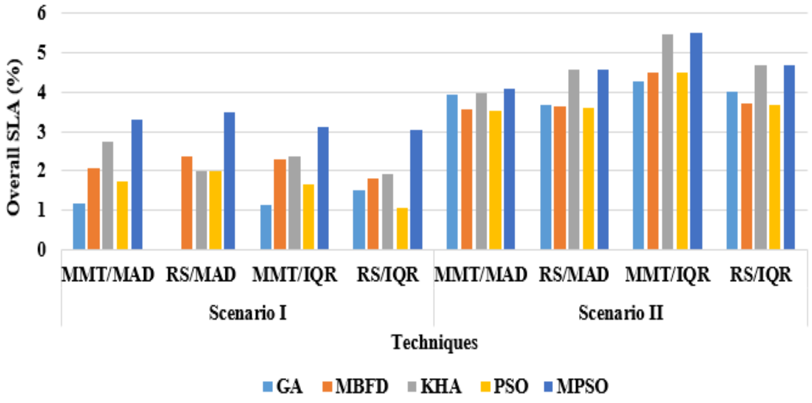

Figure. 3 Graphical presentation of MPSO technique in light of overall SLA

overall energy SLA violation related to the comparative techniques such as GA, MBFD, KHA [21] and PSO on both the scenarios. The MPSO technique effectively trade-off the imbalance among loads across 290 VMs in scenario I, and 1175 VMs in scenario II. Once the system is found imbalance, the loads are migrated from heavy-loaded VMs to light-loaded VMs to create a balance among VMs. The graphical presentation of MPSO technique by means of overall SLA, and overall energy SLA violation are indicated in the Fig. 3 and 4.

Received: November 29, 2021. Revised: December 28, 2021.

Table 4. Performance investigation of MPSO technique in light of overall energy SLA violation

Overall Energy SLA violation (%)				
Scenario I				
Techniques	MMT/MAD	RS/MAD	MMT/IQR	RS/IQR
GA	0.07	0.06	0.05	0.04
MBFD	0.11	0.12	0.16	0.07
KHA [21]	0.1	0.08	0.13	0.08
PSO	0.08	0.10	0.13	0.05
MPSO	0.09	0.09	0.1	0.07

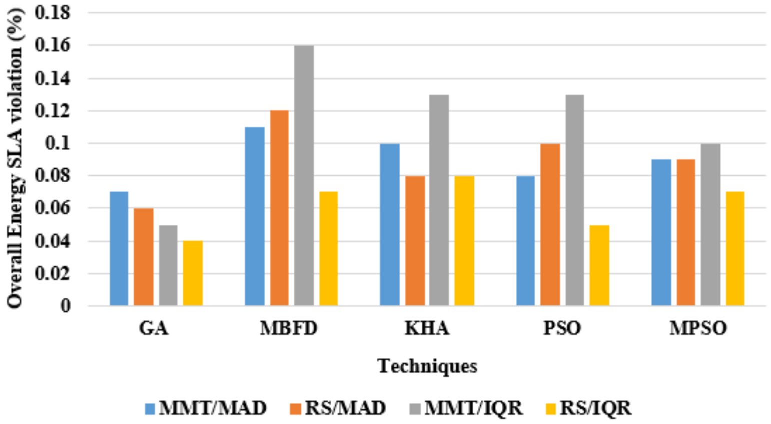

Figure. 4 Graphical presentation of MPSO technique in light of overall energy SLA violation for scenario I

4.2 Quantitative investigation related to cloud data security

In cloud data security, the effectiveness of EECC algorithm is investigated by comparing its performance with cloud Paillier [22], fast cloud Paillier [22], and ECC by means of execution time, encryption comparison time, and decryption comparison time. By inspecting table 5, the proposed EECC algorithm obtained a minimum encryption comparison time compared to cloud Paillier [22], fast cloud Paillier [22], and ECC algorithm by varying the data size from 100 to 1000 bits. Hence, the EECC algorithm almost reduced 100ms to 700ms of encryption comparison time related to the comparative algorithms. Graphical depiction of EECC algorithm in light of encryption comparison time is stated in Fig. 5.

Similarly in Table 6, the proposed EECC algorithm almost reduced 10ms to 30ms of decryption comparison time related to the comparative algorithms like cloud Paillier, fast cloud Paillier, and conventional ECC. Graphical presentation of EECC algorithm by means of decryption comparison time is denoted in Fig. 6. The proposed EECC algorithm includes the advantages like moderately fast encryption and decryption, fast signature, fast key generation, comparatively have smaller key, signature and cipher-text than the traditional cryptography algorithms.

In addition, the effectiveness of the proposed EECC algorithm is analyzed in terms of execution time. As stated in Table 7, the EECC algorithm obtained a minimum execution time by varying the number of bits from 100 to 1000. Additionally, the obtained results state that the proposed MPSO technique produces optimal solutions in load balancing and VM migration, and is also computationally effective. The experimental analysis confirmed that the proposed MPSO-EECC model effectively overcomes the problems listed in the literature [15, 16, 18, 19, 21].

Table 5. Performance investigation of EECC algorithm in light of encryption comparison time

Data size (bits)	Encryption comparison time (ms)			
	Cloud Paillier [22]	Fast Cloud Paillier [22]	ECC	EECC
100	1150	1300	1230	1050
200	1210	1380	1330	1070
300	1250	1420	1330	1100
400	1300	1500	1450	1150
500	1310	1580	1480	1200
600	1360	1620	1570	1250
700	1400	1800	1750	1300
800	1420	1900	1840	1350
900	1450	2180	2100	1400
1000	1500	2210	2160	1420

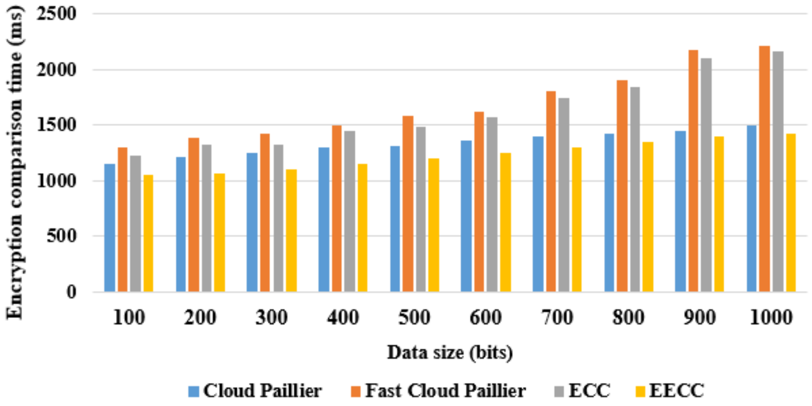

Figure. 5 Graphical presentation of EECC algorithm in light of encryption comparison time

Table 6. Performance investigation of EECC algorithm in light of decryption comparison time

Data size (bits)	Decryption comparison time (ms)			
	Cloud Paillier [22]	Fast Cloud Paillier [22]	ECC	EECC
100	220	250	210	180
200	200	250	210	190
300	220	250	200	195
400	220	250	210	205
500	210	250	220	215
600	200	250	230	220
700	210	250	220	230
800	210	250	220	235
900	220	250	220	240
1000	230	260	240	240

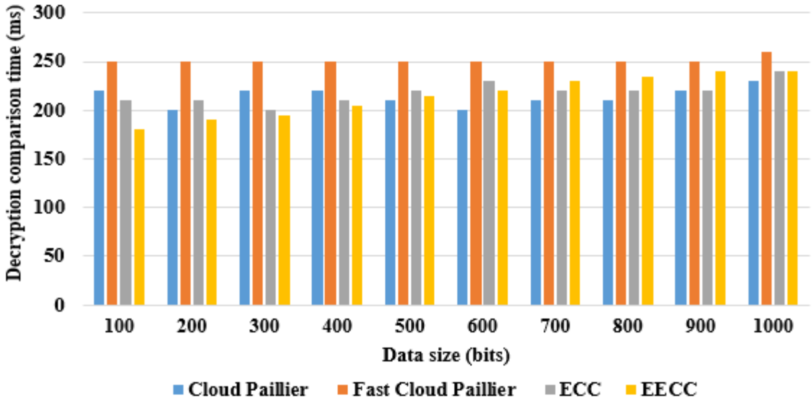

Figure. 6 Graphical presentation of EECC algorithm in light of decryption comparison time

Table 7. Performance investigation of EECC algorithm in light of execution time

EECC algorithm	
Data size (Bits)	Execution time (ms)
100	1250
200	1290
300	1310
400	1370
500	1430
600	1500
700	1530
800	1600
900	1670
1000	1690

5. Conclusion

In this article, a new model: MPSO-EECC is proposed in the cloud computing platform for effective load balancing and data security. The MPSO technique achieves a set of non-dominated solutions that help in maximizing the system's reliability and minimizing resource wastage and energy consumption. Correspondingly, the EECC algorithm is a pseudo-random based public key encryption algorithm that works based on elliptic curve theory, which creates smaller, faster, and efficient cryptographic keys. Further, the EECC algorithm servers as a secure tool for modeling a secure platform for cloud applications. The proposed MPSO-EECC model achieved effective performance in load balancing and data security compared to existing optimization techniques and cryptography algorithms by means of energy consumption, energy SLA violation, execution time, SLA, encryption comparison time, and decryption comparison time. As seen in the resulting phase, the proposed MPSO-EECC model almost minimized 30%-70% of energy consumption, and10ms to 30ms of decryption comparison time, which are better related to the comparative models. As a future extension, a new deep learning model can be included in the proposed MPSO-EECC model to further reduce the execution time and energy consumption in the cloud computing environment.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

The paper conceptualization, methodology, software, validation, formal analysis, investigation, resources, data curation, writing—original draft preparation, writing—review and editing, visualization, have been done by 1st author. The supervision and project administration, have been done by 2nd author.

References

[1] A. Mosa and N. W. Paton, “Optimizing virtual machine placement for energy and SLA in clouds using utility functions”, *Journal of Cloud Computing*, Vol. 5, No. 1, pp. 1-17, 2016.

[2] H. Qiu, H. Noura, M. Qiu, Z. Ming, and G. Memmi, “A user-centric data protection method for cloud storage based on invertible DWT”, *IEEE Transactions on Cloud Computing*, Vol. 9, No. 4, pp. 1293-, 1304, 2019.

[3] G. Tian, H. Ma, Y. Xie, and Z. Liu, “Randomized deduplication with ownership management and data sharing in cloud storage”, *Journal of Information Security and Applications*, Vol. 51, p. 102432, 2020.

[4] R. K. Devi, G. Murugaboopathi, and M. Muthukannan, “Load monitoring and system-traffic-aware live VM migration-based load balancing in cloud data center using graph theoretic solutions”, *Cluster Computing*, Vol. 21, No. 3, pp. 1623-1638, 2018.

[5] K. V. S. Reddy, J. Srinivas, and A. A. M. Qyser, “A dynamic hierarchical load balancing service architecture for cloud data centre virtual machine migration”, In: *Proc. of the Smart Intelligent Computing and Applications, Springer, Singapore*, pp. 9-17, 2019.

[6] S. Mohanty, P. K. Patra, M. Ray, and S. Mohapatra, “A novel meta-heuristic approach for load balancing in cloud computing”, In: *Proc. of Research Anthology on Architectures, Frameworks, and Integration Strategies for Distributed and Cloud Computing, IGI Global*, pp. 504-526, 2021.

[7] N. H. Shahapure, P. M. Rekha, and N. Poornima, “Threshold Compare and Load Balancing Algorithm to for Resource Optimization in a Green Cloud”, *Revista Geintec-Gestao Inovacao E Tecnologias*, Vol. 11, No. 4, pp. 4465-4481, 2021.

[8] S. Kaushik and C. Gandhi, “Ensure hierarchal identity based data security in cloud environment”, *International Journal of Cloud Applications and Computing (IJCAC)*, Vol. 9, No. 4, pp. 21-36, 2019.

[9] S. Namasudra, D. Devi, S. Kadry, R. Sundarasekar, and A. Shanthini, “Towards DNA based data security in the cloud computing environment”, *Computer Communications*, Vol. 151, pp. 539-547, 2020.

[10] P. Gupta, D. K. Verma, and A. K. Singh, “Improving RSA algorithm using multi-threading model for outsourced data security in cloud storage”, In: *Proc. of 8th International Conference on Cloud Computing, Data Science & Engineering*, pp. 14-15, 2018.

[11] Y. Zhang, X. Chen, J. Li, D. S. Wong, H. Li, and I. You, “Ensuring attribute privacy protection and fast decryption for outsourced data security in mobile cloud computing”, *Information Sciences*, Vol. 379, pp. 42-61, 2017.

[12] A. T. Lo'ai and G. Saldamli, “Reconsidering big data security and privacy in cloud and mobile cloud systems”, *Journal of King Saud University-Computer and Information Sciences*, 2019.

[13] L. K. Ramachandrappa and P. C. Renukaradhya, “Data Security using Proxy Based Methods in Cloud Computing”, *International Journal of Advanced Scientific Innovation*, Vol. 2, No. 3, pp. 9-13, 2021.

[14] V. Goyal and C. Kant, “An effective hybrid encryption algorithm for ensuring cloud data security”, *Big Data Analytics, Springer, Singapore*, pp. 195-210, 2018.

[15] A. Ibrahim, M. Noshy, H. A. Ali, and M. Badawy, “PAPSO: A power-aware VM placement technique based on particle swarm

optimization", *IEEE Access*, Vol. 8, pp. 81747-81764, 2020.

[16] K. Balaji, P. S. Kiran, and M. S. Kumar, "An energy efficient load balancing on cloud computing using adaptive cat swarm optimization", *Materials Today: Proceedings*, 2021.

[17] J. P. B. Mapetu, L. Kong, and Z. Chen, "A dynamic VM consolidation approach based on load balancing using Pearson correlation in cloud computing", *The Journal of Supercomputing*, Vol. 77, No. 6, pp. 5840-5881, 2021.

[18] U. K. Jena, P. K. Das, and M. R. Kabat, "Hybridization of meta-heuristic algorithm for load balancing in cloud computing environment", *Journal of King Saud University-Computer and Information Sciences*, 2020.

[19] S. Namasudra, "An improved attribute-based encryption technique towards the data security in cloud computing", *Concurrency and Computation: Practice and Experience*, Vol. 31, No. 3, p. e4364, 2019.

[20] M. Tahir, M. Sardaraz, Z. Mehmood, and S. Muhammad, "CryptoGA: a cryptosystem based on genetic algorithm for cloud data security", *Cluster Computing*, Vol. 24, No. 2, pp. 739-752, 2021.

[21] M. Soltanshahi, R. Asemi, and N. Shafiei, "Energy-aware virtual machines allocation by krill herd algorithm in cloud data centers", *Heliyon*, Vol. 5, No. 7, p. e02066, 2019.

[22] K. E. Makkaoui, A. Ezzati, A. B. Hssane, and S. Ouhmad, "Fast Cloud-Paillier homomorphic schemes for protecting confidentiality of sensitive data in cloud computing", *Journal of Ambient Intelligence and Humanized Computing*, Vol. 11, No. 6, pp. 2205-2214, 2020.

[23] J. C. Bansal, "Particle swarm optimization", *In Evolutionary and Swarm Intelligence Algorithms, Springer, Cham*, pp. 11-23, 2019.

[24] B. Chopard and M. Tomassini, "Particle swarm optimization", *An Introduction to Metaheuristics for Optimization, Springer, Cham*, pp. 97-102, 2018.

[25] S. Kumari, M. Karuppiah, A. K. Das, X. Li, F. Wu, and N. Kumar, "A secure authentication scheme based on elliptic curve cryptography for IoT and cloud servers", *The Journal of Supercomputing*, Vol. 74, No. 12, pp. 6428-6453, 2018.

[26] S. Ibrahim and A. Alharbi, "Efficient image encryption scheme using Henon map, dynamic S-boxes and elliptic curve cryptography", *IEEE Access*, Vol. 8, pp. 194289-194302, 2020.

[27] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. Derose, and R. Buyya, "CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms", *Software Practice and Experience*, Vol. 41, pp. 23-50, 2011.

[28] R. Yadav, W. Zhang, O. Kaiwartya, P. R. Singh, I. A. Elgendy, and Y. C. Tian, "Adaptive energy-aware algorithms for minimizing energy consumption and SLA violation in cloud computing", *IEEE Access*, Vol. 6, pp. 55923-55936, 2018.

[29] A. M. Ammar, J. Luo, Z. Tang, and O. Wajdy, "Intra-balance virtual machine placement for effective reduction in energy consumption and SLA violation", *IEEE Access*, Vol. 7, pp. 72387-72402, 2019.