Received July 17, 2020, accepted July 23, 2020, date of publication July 31, 2020, date of current version August 13, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3013282

EC-ElGamal and Genetic Algorithm-Based Enhancement for Lightweight Scalable Blockchain in IoT Domain

J. GURUPRAKASH AND SRINIVAS KOPPU
SITE, Vellore Institute of Technology, Vellore 632014, India

Corresponding author: Srinivas Koppu (srinukoppu@vit.ac.in)

ABSTRACT Security, data privacy and decentralization are significant challenges in the Internet of Things (IoT) domain. These challenges are inherited attribute of another emerging technology, Blockchain. This enforced convergence of IoT and Blockchain, attracting researchers to study on the effective use of Blockchain's strength to solve the challenges of IoT. Rapid IoT adoption requires standardization and mature solution on security, data protection for compliance and performance for commercialization. These demands made a surge in variant blockchain flavours and combinations catering to different problems, and one such is Lightweight Scalable Blockchain (LSB). LSB had considerable caveats that require improvement for better adoption in the IoT domain. This paper focuses on encrypting transaction transmission, improving transaction flow, block validation, hash quality, hash rate and storage cost to improvise security and performance. The experimental evaluation is demonstrated on data from the temperature sensor to showcase superior applicability of the proposed work in the IoT domain. Implementation and result comparison with conventional LSB proves, the following achievements 1) An additional layer of transaction encryption using hybrid Elliptic Curve ElGamal (EC-ElGamal) method increases the security of the transmitted transaction for security enhancement. 2) Obtained 20% reduction in transaction processing time, 22% reduction on block validation processing time, 53% improvement on the hash operation and quality with an overall 7% saving on the storage cost thereby increased the overall performance.

INDEX TERMS IoT blockchain, LSB, EC-ElGamal encryption, genetic algorithm, IoT.

I. INTRODUCTION

The Internet of Things (IoT) is a developing technology that connects the self-configurable devices to create a dynamic and efficient platform for association and communication [1]. IoT devices are usually low-power heterogeneous devices that are resource-constrained in terms of storage and computation [2]. IoT systems communicate with each other to provide services and exchange information. In a distributed network, communication should operate with minimum delay. Distributed networks are also susceptible to security threats [3] arising the need for high-security mechanisms to secure IoT systems in the network [4]. However, traditional mechanisms and centralized security authorization are inadequate due to scalability problems [5], [6] and there is thus a need for decentralized access control and delay-sensitive authentication for IoT devices to operate optimally [7]. Scalability and latency are other important problems. Fog nodes address these problems by allowing limited communication, storage, control, and supervision at the network edge, serving as dedicated channel technologies to replace a centralized remote environment [8].

Blockchain is an emerging technology whose attributes can address various challenges in IoT. For example security, authentication, access control, and the distributed nature of blockchain technology can be applied in IoT [9]. Blockchain's core cryptographic features and decentralized nature [10], [11] have also made it a favourable option for applications in which high security and decentralization are required. Initially, blockchain was used in Bitcoin a peer-to-peer currency, as illustrated in Fig. 1. Subsequently, experts examined fundamental blockchain techniques in IoT implementation to strengthen security and authentication. Blockchain's characteristics of reliability, fault tolerance, and unforgeability was demonstrated to be useful additions to IoT. In addition, a smart contract can be incorporated in IoT implementation for autonomous decision-making and fine-grained access [12].

The associate editor coordinating the review of this manuscript and approving it for publication was Yassine Maleh.

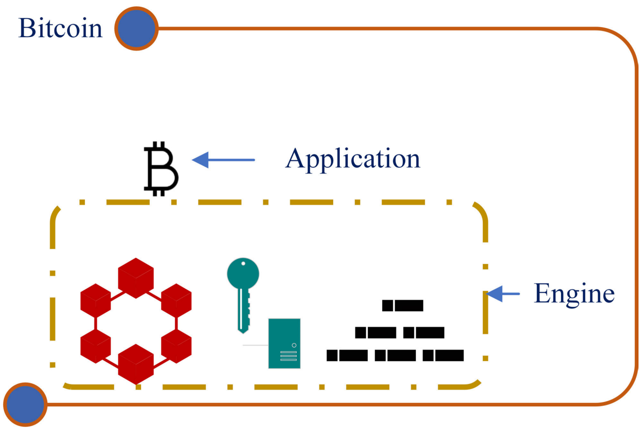

FIGURE 1. Bitcoin blockchain.

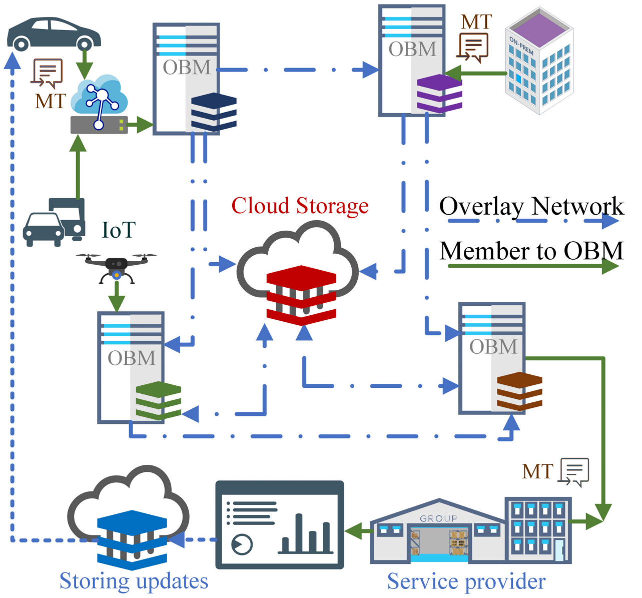

FIGURE 2. Overview of Lightweight Scalable Blockchain.

Existing blockchain has several limitations, such as overhead when the node count in the network increases, and occasional untraceability. In addition, blockchain involves complex consensus algorithms and has limited throughput caused by increased transactions [13], [14]. These challenges are overcome by Lightweight Scalable Blockchain (LSB), which is illustrated in Fig. 2. LSB uses overlay networks and clusters on IoT devices and service providers. The selected cluster head performs the blockchain management [15]. The Blocks formed by transactions are hashed and stored in the blockchain as a distributed ledger [16], [17].

The main objectives of this proposed work are as follows,

- Improve the encryption and decryption process using Elliptic curve ElGamal (EC-ElGamal).
- Develop high-performance hashing using SHA-384 with the Genetic Algorithm.

The remainder of this paper is organized as follows. Section II provides an overview of existing literature, while Section III describes the proposed method. Section IV presents the results and discussion, followed by Section V that concludes the paper and discusses future directions.

II. LITERATURE REVIEW

This section presents a review of the literature concerning to Lightweight blockchain's, Blockchain-based IoT solutions, challenges on blockchain approaches and cryptography. The conclusions drawn aid in formulating the proposed methodology.

In [18], the authors studied LSB for optimization, using smart home applications as an example. An overview of LSB is presented in Fig. 2. The figure illustrates the centralized manager in a smart-home-enabled environment with shared keys in low-resource devices for using incoming and outgoing requests for communication. An overlay network is used to achieve this decentralization in LSB and generic blockchain, which are managed by high-resource devices with ensured privacy and end-to-end security. Cluster head formation on the overlay network decreases the overhead [19].

LSB integrates various optimized algorithms for trust distribution and throughput management. LSB is also secure against many security attacks discussed by qualitative arguments. The simulation result in [18] demonstrated that LSB reduced the packet overhead and delay and increased scalability when compared to other methods. The LSB framework also reduced the processing time compared to existing methods without additional delay for smart home services. LSB ensures the privacy and security of IoT applications and users at a high level. A distributed time-based consensus algorithm was proposed in [18] that was used to decrease the block delay and overhead of the mining process by cluster heads. Distributed throughput management was also used to increase throughput efficiency and avoid transaction load in network deviation.

In [20], the authors focused on resource-constrained IoT sensor devices and proposed a sensor-chain lightweight scalable blockchain framework. Conventional blockchain was reduced to lightweight blockchain in the following three stages. In the first stage, in the spatial domain, small local blockchains were formed in a disjoint manner and were held in a storage space for the IoT device. In the second stage, a size limitation in the temporal domain was imposed by a temporal constraint on the local blockchain lifespan. In the third stage,in memory local blockchain was retained on a temporal basis at the sensor node. The sensor chain was then evaluated based on long-term performance and scalability. The results indicated that the storage space was superior to that of existing methods. However, validation was fast on the local blockchain, whereas the proposed method resulted in additional block validation overhead and delay.

In [21], the authors designed a privacy-preserving blockchain. The study also minimized the need for a centralized entity, as blockchain itself guarantees built-in integrity, trust and immutability of security of information. The main problems examined were the lack of transaction

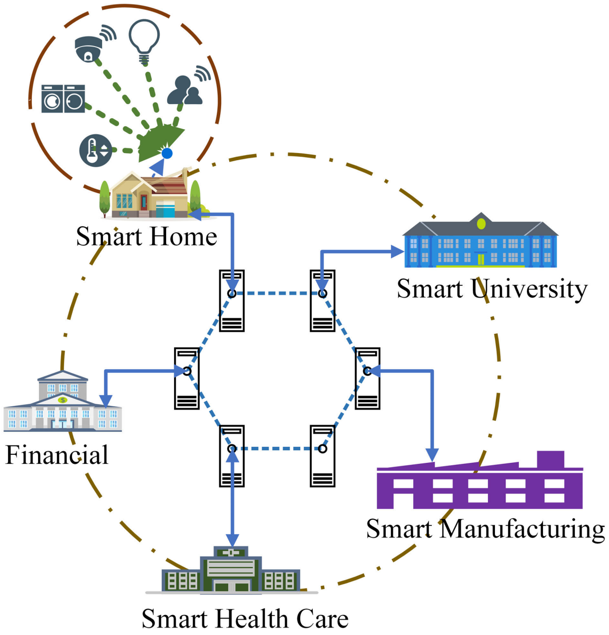

FIGURE 3. Smart Home IoT Architecture with Blockchain.

privacy, slow transaction rate, and high-resource consumption. Although directed acyclic graph improved transactions in blockchain, other problems remained. The higher transaction rate achieved by the method proposed in [21] involved the hash graph process of miner-free transaction validation. The authors called this method PrivLiteChain, a blockchain implemented with temporal constraints to make the blockchain lightweight and enforce transaction privacy with local differential privacy.

In [22], the authors used LSB in a smart home setting to ensure security and privacy. Their study involved three tiers consisting of cloud storage, overlay, and a smart home. Fig. 3 presents a diagram illustrating a smart home with the three-tier smart city set-up. A miner is employed on high-resource devices to handle all communications of internal and external sensors, while the blockchain backbone ensures security for communication, control, and audit. The authors also studied various attacks on LSB and demonstrated that a denial-of-service attack, distributed denial-of-service attack, dropping attack, chain modification, compromising time attack, Sybil attack, consensus attack, and 51% attack were unlikely to occur in LBS. The results demonstrated increased privacy and security but a considerable increase in the block overhead, processing time, resource consumption, and network traffic.

In [23], [24], the authors focused on developing lightweight and hash-based blockchain for Industrial IoT (IIoT). The blockchain hash function flexibly changed based on the transactions rate to enhance the blockchain availability in the network. The authors used a lightweight hash function, such as QUARK, PHOTON, and SPONGENT, to increase the performance, and examined the throughput and resource consumption. The use of a lightweight hash aimed to reduce the resource constraints of the devices used in the implementation and to ensure cryptographic security. The generated blocks were connected with the flexible hash generated; therefore, the latency and computational complexity were reduced. The process could be monitored, supervised, and controlled directly by fields and cell nodes; thus, the scalability was improved. Additional lightweight hash functions [25] have been developed, such as parallel hash functions, that were proposed for various IoT applications with less latency.

In [26], the authors examined an LSB trust mechanism in the vehicular IoT domain. An on-off attack pursued by a malicious participant present in blockchain architecture can degrade the integrity of the distributed ledger. In the study [27], violation of the system was monitored and controlled by remote software. The actor-based language Rebeca was used to analyse the attack and model a system under such an attack. This model explored possible attacks, especially on-off attacks, and the effectiveness of LSB against them. The Rebeca model used an evaluation/testing tool for verifying the properties of security in a distributed environment.

In [28], the authors proposed a multi-layer hierarchical architecture for monitoring and managing underwater IoT (IoUT) on cloud data using blockchain. In the proposed framework, cluster-based sensors were formed and organized depending on the selected residual energy cluster head, and data from nodes were routed to the higher layer. The authors used the Bloom filter for cluster head and node tracking. The standard secret key was used by the IoUT protocol for gateway communication, and another secret key was used by the cluster head. The routed data were then stored in the blockchain. The scalability problem as well as transaction preparation and routing to miners were solved by the fog layer's smart gateway, which was incorporated into the IoUT blockchain [29]. A lightweight consensus mechanism was used to add blocks in the blockchain in which the IoUT data were stored. The feasibility of the architecture was evaluated by performance and security analysis.

In [30], the authors proposed cyber security for IoT environments based on lightweight blockchain. Managing local and public transactions on IoT devices requires separate blockchain for transactions to reduce attack size. LBC solved consensus algorithm uniquely by reducing the waiting period of transactions to a larger extent than existing approaches.

In [31], scalability problems in access control were addressed, as performance is often degraded when a centralized access control system cannot handle increased load. The authors proposed a fully decentralized blockchain method to overcome these problems, and the proposed method was applicable to many constrained IoT devices. The proof of concept used ensured scalable and easy-to-manage IoT access control system features. Fig. 4 presents this decentralized network, displaying management hub nodes and IoT device nodes connected to the blockchain, ensuring high flexibility [32].

In [33], the authors proposed a multilevel blockchain system to improve privacy and data security in IoT applications.

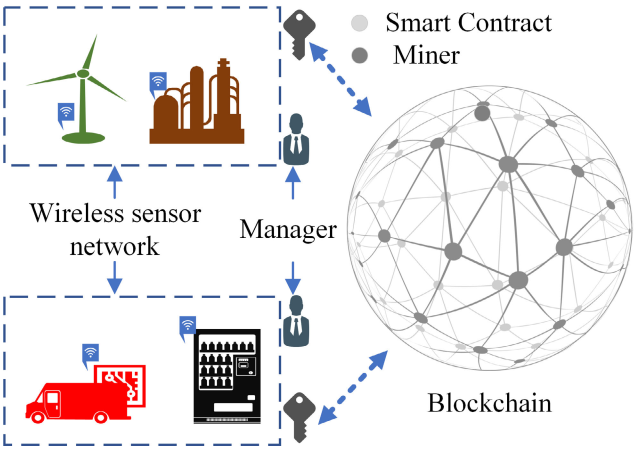

FIGURE 4. Decentralized access control system.

The authors focused on improving resource consumption and response time. The study mainly used mobile agents that transferred between blockchain and IoT to execute encryption, hash functions, aggregation, and decryption functions, as required, and the simulation results were satisfactory.

In [34], the focus was on the rapid development of IIoT with a topology for edge computing and resource-constrained devices. The proposed topology allowed new challenges in security, data transmission, and data storage to be solved. For distributed IoT, trust distribution and ledger-based blockchain technology were well suited for edge computing.

In [35], a resource-constrained layered lightweight blockchain framework was proposed for IIoT, containing a resource-constrained layer and resource-extended layer with a dynamic trust algorithm and lightweight consensus algorithm. This framework was used to enhance security and decrease the transaction rate in new blocks [36]. Transaction load balance was achieved in blockchain, and the simulation results yielded better performance for IIoT compared with baseline.

In [37], [38], LightChain was proposed as blockchain and discussed for secure service provisioning. The features of public and private consortium blockchain were used, and authorized users maintained the blockchain. The data were read in the blockchain only by public users and evaluated the service codes. The proof-of-authority consensus mechanism was used for blockchain consensus. The Keccak256 hashing algorithm was used to convert arbitrary data into a fixed-size hash because it exhibits lower gas consumption than other algorithms. By using proof-of-authority, the gas consumption decreased to 17 gas units compared to using proof-of-work.

In [39], the authors focused on data integrity in surveillance cameras using lightweight blockchain. Videos are used for criminal investigations, and video footage is thus substantial evidence. Video evidence can be obtained from both trusted and untrusted surveillance systems. From untrusted sources, the integrity of the information is questionable. In [39], an airport ecosystem was used as an example, and a variety of video sources with different trust levels were used to produce video surveillance information. The blockchain-based implementation ensured tampered proof and authentic video storage. Lightweight blockchain technology was used to save the video meta data as blockchain transactions to support video integrity. The study ensured non-repudiation and audit. The paper also discussed the latency overhead introduced by blockchain.

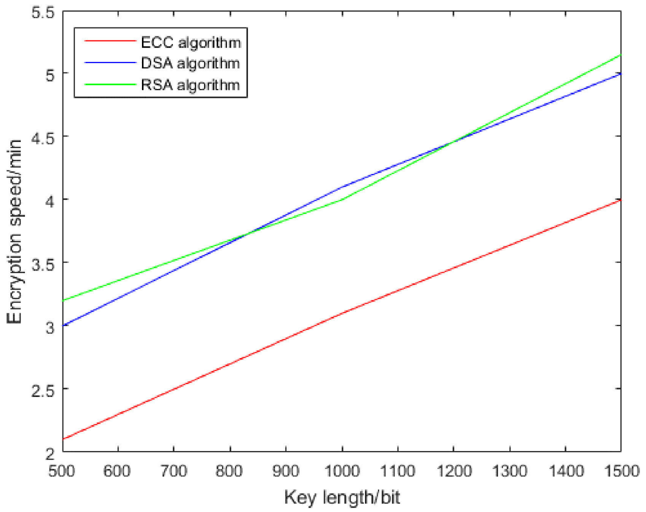

FIGURE 5. Speed of ECC compared to DSA and RSA algorithm, as evaluated in [42].

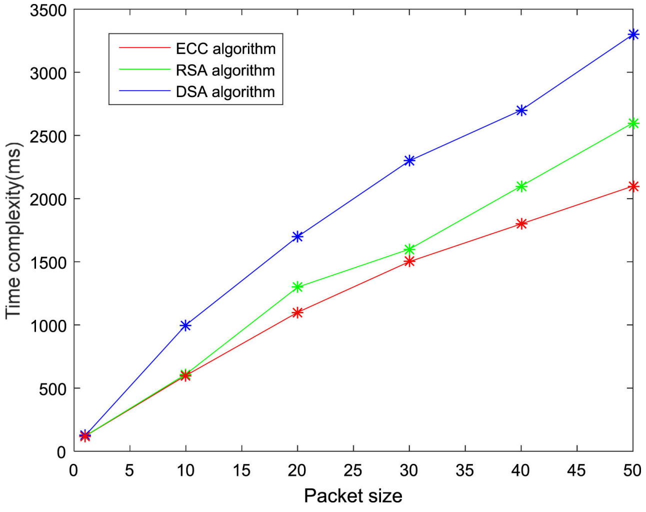

FIGURE 6. Time Complexity of ECC compared to RSA and DSA, as evaluated in [42].

Authors in [40] discussed the guarantee of information security by employing Elliptic Curve Cryptography (ECC) algorithm. In [41], the authors examined image encryption. Comparative analysis of ECC was performed with Digital Signature Algorithm (DSA) and Rivest–Shamir–Adleman (RSA) algorithms in terms of speed see Fig. 5 and in terms of time complexity see Fig. 6. The results revealed that the encryption accuracy of ECC was 43% and 30% higher than that of DSA and RSA.

The study also demonstrated that ECC with 165 bits resulted in the same security as RSA and DSA with 1,203 bits, and ECC with 210 bits demonstrated performance matching

TABLE 1. Summary of RSA and ElGamal.

Attributes	RSA	ElGamal
Key length	>1024 bits	1024 bits
Algorithm Type	Asymmetric	Asymmetric
Simulation	Fast	Fast
Scalability	No	Good
Encrypt	Different	Different
Decrypt Keys		
Power	High	Low
Implementation	Not efficient	Faster and Efficient

TABLE 2. Comparison of RSA and EC.

Security Bit level	RSA	EC
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	512

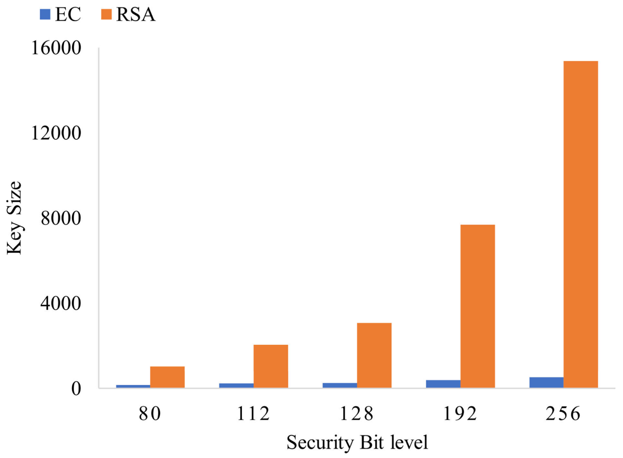

FIGURE 7. Comparison of RSA and EC.

RSA and DSA with 2,105 bits. This comparison indicated that ECC had high-security performance.

The authors in [43] compared Elliptic curve and ElGamal with RSA, on various attributes as represented in Table 1 and Table 2. RSA and ElGamal share similarities in algorithm type, simulation and keys, ElGamal standout in key length, scalability, power consumption and implementation. In Fig. 7 the comparison is based on the key size that affect the run time. EC has become the de facto algorithm for cryptography based on key length and efficiency.

In [44] the authors discussed the relevance of EC based on the algebraic structure of a curve over a finite field in modern applications.

In [45], the authors suggested that Secure Hash Algorithms (SHA) require optimization and a hybrid of multiple techniques based on the underlying application. A comparison of the SHA family of algorithm as displayed in Table 3 based on security standards reveals the functional, security level, and block size. As derived from [45], the use of SHA in various application as displayed in Table 4 highlights the importance of SHA algorithms for use in IoT and blockchain.

In this section, a literature review of LSB, Blockchain-based IoT applications, traditional IoT security measures, lightweight blockchain, attacks on LSB, and a comparison study of EC and ElGamal with the RSA and SHA family is presented. The key conclusion is that blockchain's inherent features solve most IoT challenges; however, the limitations of generic blockchain create the need for lightweight blockchain. Lightweight blockchain has been proposed by various researchers based on their respective challenges and focused areas of improvement. In general, for the IoT domain, lightweight blockchain is the best option, as it helps overcome the high-resource utilization of generic blockchain. However, the majority of existing studies do not focus on improving security or optimizing performance to improve lightweight blockchain for IoT. The conclusions drawn from this literature review can be summarized as follows: 1) EC displays superior results to those of RSA, 2) ElGamal presents a better outcome than RSA, 3) most of the conventional attacks are unlikely to occur in LSB, and 4) Improving security and performance can help increased industry adoption of Blockchain-based IoT solution. The literature review also identifies factors that can be improved.

TABLE 3. Comparison of SHA Family.

Parameter	SHA-256	SHA-384	SHA-512
Hash size (bit)	256	384	512
Block size (bit)	512	1024	1024
Message size (bit)	$< 2^{64}$	$< 2^{128}$	$< 2^{128}$
Security bits	128	192	256
Cipher	AES-128	AES-192	AES-256
Word size (bit)	32	64	64
Number of Rounds	64	80	80

Note: Comparison taken from [45]

TABLE 4. Requirements of applications relying on SHA.

Application	Throughput	Power
Bitcoin Mining	Critical	Major
IoT Cryptography	Moderate	Minor
IoT Mining	Critical	Major

Note: Based on survey from [45]

III. PROPOSED SYSTEM

This study aims to improve the security and performance of LSB using EC-ElGamal and Genetic algorithm based key generation for SHA-384 as shown in Fig. 8. The following metrics are considered: transaction flow, block validation, hash rate, hash quality, and storage cost.

Encryption of transactions with EC-ElGamal helps add greater security. Increasing the transaction flow or decreasing the transaction overhead can help support performance improvement in LSB and can demonstrate suitability for IoT, where thousands of sensors are continuously pushing sensory data as transactions to the blockchain. Improvements in block validation demonstrate that LSB can handle IoT device transactions without a queue. Hashing, the core of block aggregation and block validation, can help achieve high performance and security with a high hash rate and quality. Storage utilization cost is another major factor in blockchain implementation; therefore, a superior solution to the existing model is necessary.

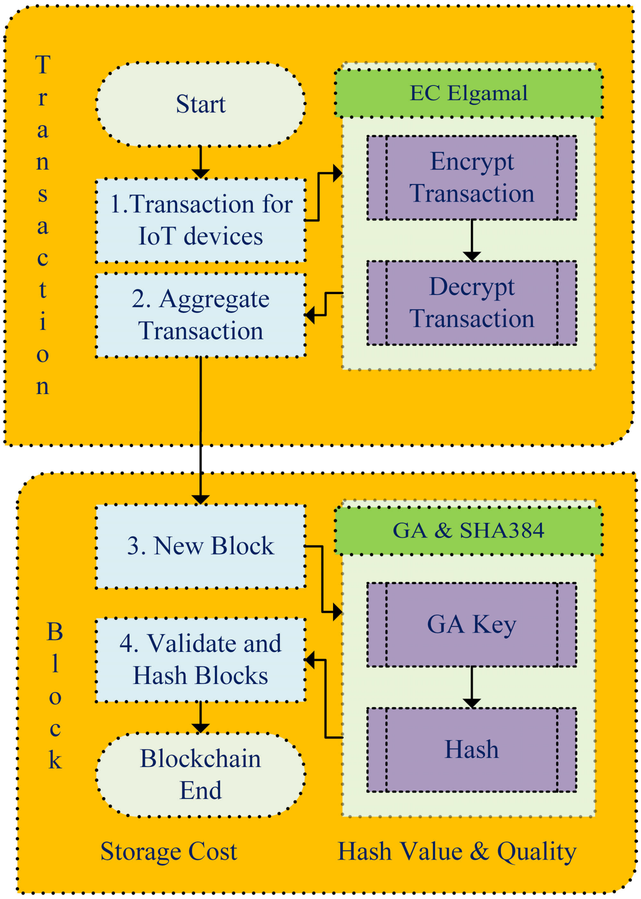

FIGURE 8. Proposed workflow.

A. METHODOLOGY: EXPERIMENTAL DESIGN

1) Temperature sensor data are considered as transaction data that originate from geographically distributed sensors.
2) The transactions are encrypted during transmission from a sensor and decrypted upon receipt at the gateway. (Algorithm 1)
3) The transactions are aggregated and await further transfer into the network. (Algorithm 1)
4) The aggregated transactions are further pushed to become blocks.
5) At the block manager and validation node, blocks are managed, and all unverified blocks are validated.
6) Hashing of the blocks is performed. (Algorithm 2)
7) Blocks are hashed and linked to the address of the previous hash. (Algorithm 2)
8) The peer-to-peer transfer of the blocks is represented.

Based on Algorithm 1, encryption and decryption of transaction is performed. Scenario 1 — Input is transactions generated from the IoT devices, and the output is an encrypted transaction. Scenario 2 — Input is an encrypted transaction, and the output is a decrypted transaction. Encryption and decryption happen only within the participating nodes inside the ecosystem, thereby eliminating unauthorized access on the transaction data. Algorithm 1 uses EC-Elgamal, which is not present in conventional LSB. Hence, providing an additional layer of enhanced security to transactions in our proposed method. Algorithm 2 describes block hashing. We have proposed a hybrid method using Genetic algorithm and SHA-384. The input to this algorithm is Key and Blocks, and the output is uniform length Hash digest.

Algorithm 1 EC-ElGamal Based Transaction Encryption and Decryption

```
EC ElGamal - Transaction Encryption Input:
Transaction
Output: Encrypted or Decryption
Step 1:
Input: Transaction
Output: Encrypted Transaction
for (Each T in Transaction) do
    Encrypt Transaction
Step 2:
Input: Encrypted Transaction
Output: Transaction
for (Each ET in Encrypted Transaction) do
    Decrypt Transaction
```

Algorithm 2 Genetic Algorithm (GA) and SHA-384 Based Bock Hashing

```
GA + SHA 384 - Block hashing
Input: Block
Output: Hash Digest
Step 1:
Input: Population
Output: Prime Keys
for (Each P in Population) do
    Generate Prime Key
Step 2:
Input: Block and Keys
Output: Hash Digest
for (Each B in Block) do
    Hash Digest = $Hash_{(Block,Keys)}$
```

B. REPRESENTATIONAL ARCHITECTURE OF LSB FOR IoT DOMAIN

In Fig. 9 the architecture flow of the experiment and evaluation are numbered as follows: 1) Temperature sensor push the temperatures reading continuously to the underlying backbone. 2) The transactions are moved to the respective gateway manager. 3) Transaction aggregation happens at the gateway manager. 4) The unverified block is moved further to the block manager. 5) Block validation is performed. 6) The validation is followed by hashing. 7) The hashed blocks are pushed to the overlay network. 8) Peer-to-peer exchange of the block occurs across the nodes.

- In path 3, Transaction flow is measured.
- In path 5, Block validation rate is calculated

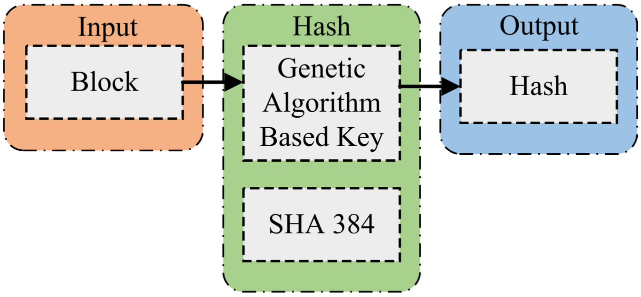

FIGURE 9. Representational Architecture of proposed work.

- In path 6, Hash rate is calculated
- In path 7, Hash Quality is measured
- And the end of Path 7, Storage cost is estimated

C. EC-ElGamal CRYPTOGRAPHY

An Asymmetric data encryption algorithm is proposed based on Elliptic curve ElGamal cryptography. Elliptic Curve Discrete Logarithmic Problem (ECDLP) defined as asymmetric cryptosystem developed based on Elliptic Curve Cryptosystem (ECC). Mixing the elliptic curve law of addition and discrete logarithmic operations based on elliptic curves, we establish a hybrid cryptosystem.

The ECDLP exhibits irrevocable solution. ElGamal cryptosystem generally called as EC-ElGamal Encryption, has faster speed and better performance with a short key length. The security of EC-ElGamal encryption is higher than that of traditional approaches [46]. The Encryption and decryption process with respect to EC-ElGamal techniques represented below:

Let E_p considered as an elliptic curve over finite field represented in the form of,

$$y^2 = x^3 + ax + b\,(mod\ (p)); \tag{1}$$

where a, b are two constants which satisfy

$$4a^3 + 27b^2 \neq 0, p - prime \tag{2}$$

The Abelian property represented as additive followed by elliptic curve co-ordinates Consider D as an infinite point from Elliptic curve E_p four points A, B, C, D.

$$D + A = A + D = A; \tag{3}$$

$$-D = D; \tag{4}$$

$$If\ A\,(x;\ y) \neq 0, \quad then\ -A = (x;\ -y); \tag{5}$$

$$If\ B = -A, \quad then\ A + B = D; \tag{6}$$

$$If\ A \neq B, \quad B \neq D, \ B \neq -A, \tag{7}$$

The straight-line intersection AB (if A≠ B) denoted by C or E_p at point of intersection of A (if A = B) with elliptic curve E_p another point, hence A + B = C. A (x_1, y_1) and B (x_2, y_2) are randomly taken on elliptic curve and at another point C straight line is made. Further, on y-axis over C to C' cross the parallel line, and A (x_1, y_1) +B (x_2, y_2) =C' (x_3, y_3), C' = -C = $(x_3, -y_3)$ defined. Hence, C' (x_3, y_3) which is the point addition result and given by

$$\begin{aligned} x_3 &= \tau^2 - x_1 - x_2 \\ y_3 &= \tau(x_1 - x_3) - y_1 \end{aligned} \tag{8}$$

where $\tau = slope$

$$\tau = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} & if\ A \neq B \\ \dfrac{3x_1^2 + a}{2y_1} & if\ A = B \end{cases} \tag{9}$$

Algorithm 3, shows step wise procedures of EC-ElGamal.

Algorithm 3 Elliptic Curve - Elgamal

```
EC - Elgamal
Step 1: For receiver Key Generation
Input: $E_p$
Output: $E_p, p, L, B$
(i) The equation of Elliptic curve, $E_p$: $y^2 = x^3 + ax + b$
where, p - prime and L - basic point are chosen.
(ii) By receiver, private key is set denoted by Pr, further, then
B = $P_r L$ is calculated.
where, B is the transformed private key for the receiver
(iii) Return the value of keys $E_p, p, L, B$.
  Step 2: Transmitter Encryption processes:
Input: Plain Text R
Output: Encrypted text $E_1$,$E_2$
(i) R is the plain text, R' is converted to point on field of elliptic curve. using (2)(3) equations
(ii) The transmitter set $P_{r1}$ which is a private key, then
$E_1 = P_{r1}$L and
$E_2$ = R' ⊗ $P_{r1}$B,
where addition operation is denoted by symbol of ⊗
(iii) To the receiver $E_1$,$E_2$ encrypted data transmitted.
  Step 3: Receiver Decryption process:
Input: $E_1$, $E_2$
Output: R Decrypted text
(i) According to Pr the receiver private key, R' is given by
R' = $E_2 \oslash P_r E_2$
(R' ⊗ $P_{r1}$B) ⊘ $P_r$ ($P_{r1}$ L)
(R' ⊗ ($P_{r1} P_r$ L ⊘ $P_r . P_{r1}$ L) = R
where Elliptic curve inverse addition operation denoted by "⊘".
(ii) From plain text R' to R, restored.
```

D. GENETIC ALGORITHM KEY GENERATION

In this method, an initial population is created using *population size* × *chromosome size* instead of using random integers. To generate a highly random key, a Genetic algorithm is applied, enhancing the existing method [47]. Fitness function synonymously called objective function helps to determine the suitability of chromosomes. As the algorithm iterates, fitness function helps to increase the best fit as

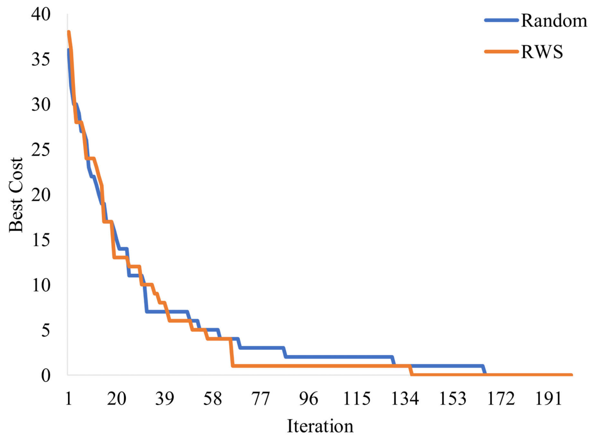

FIGURE 10. Comparison of various Crossover methods.

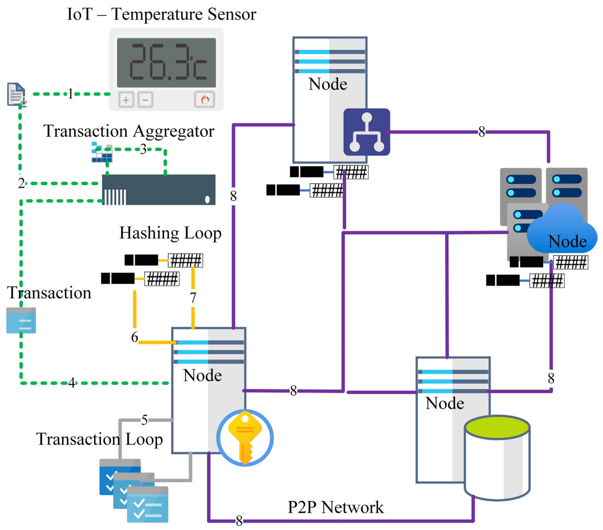

FIGURE 11. Comparison of Selection Methods.

a whole. The primes are determined based on the fitness function, *fit(x)*. This serves to make the key more robust. The process starts from the initial size of the population to select the key from the final new population. Furthermore, to produce an unpredictable and robust key, an effective selection process is applied based on Fig. 11, choice of crossover is based on Fig. 10 and Random Hybrid Crossover is represented in Algorithm 5.

The procedure for key generation using the Genetic algorithm is presented in Algorithm 4

E. SECURE HASH ALGORITHM – 384

SHA is defined from cryptographic hash functions [48]. In the proposed method, the powerful SHA-384 is used with the 384-bit block cipher algorithm, in which the intermediate hash value is encrypted using the message block with a key generated from the Genetic algorithm. The process of the hash function is illustrated in Fig. 12. The block and keys serve as input to the hash function, and the last hash values are linked with the previous hash values.

Algorithm 4 Genetic Algorithm Based Key Generation

```
Genetic Algorithm based Key Generation Input: $P_s$, $C_l$
Output: Key
Step 1: Initial population selection process
Population size = $P_s$
Chromosome length = $C_l$
We generate $P_s$x$C_l$ size of random prime numbers
Initial selection process
S= $\sum_{i,j}^{P_s,C_l} \frac{Ps_ij}{(P_s \times C_l)}$
Step 2: Fitness
$fit(x) = ((x+1)!+1)\% x)$
Step 3: Selection
RouletteWheelSelection
Step 4: Crossover process
Cross=[]
for each population do
    Population(i) = rand(0 to 1)
    if $p1 > S$ then
        break()
    else
        $Par_1$ = find(rand*sum(p)< Csum(p),1,'1')
        $Par_2$ = find(rand*sum(p)< Csum(p),1,'2')
        $Gene\_count$ = rand(0 to $P_s$(i))
        for $j=1$ to $Gene\_count$ do
            $g_1$,$g_2$=RandomHybridCrossover()
            Swap($Par_1[g_1]$, $Par_2[g_2]$)
    Update ($Par_1$, $Par_2$) in Cross[]
Step 5: Mutation process
Opt=rand(1,2)
Mutation=[]
if $opt==1$ then
    for $i$ in cross do
        R_v=rand(1,$C_l$)
        Mut = R_v
else if $opt==2$ then
    for $i$ in cross do
        R_v = rand(1,$P_s$)
        Mut = R_v
Update Mut in Mutation[]
Step 6: Offspring selection process
Selection based on fitness
Step 7: Key selection process:
Pick from new population
```

Algorithm 6, presents the step wise process of SHA-384.

The proposed methods, EC-ElGamal encryption and Genetic algorithm based key generation for SHA-384, are implemented and used in the experiments, and the results are discussed in Section IV.

IV. RESULT AND DISCUSSION

The proposed system to enhance security and performance was evaluated in terms of the transaction and block validation

Algorithm 5 Random Hybrid Crossover

```
Random Hybrid Crossover Input: $Par_1$, $Par_2$
Output: $g_1$, $g_2$
m = rand([1, 3]); switch m do
    case 1:
    g[] = SinglePointCrossover p[]
    case 2:
    g[] = DoublePointCrossover p[]
    otherwise:
    g[] = UniformCrossover p[]
```

Algorithm 6 SHA-384 Algorithm

```
Secure Hash Algorithm-384
Input: Block[],Keys[]
Output: Hash DIGEST
Step 1: Initialize
$H_{[0:7]} = Keys_{[P9:P16]}$ 9-16th primes
$K_{[0:63]} = Keys_{[P0:P79]}$ first 80 primes
Variable as $Var_{[a:h]}$
Hash Val as $H_{[0:7]}$
Step 2: Preprocessing
for i in 0:7 do
    W = W + ‘1’ append a single ’1’ bit
    for (len(h[i])) do
        W = W + ‘0’ append ‘0’ bit
    W = W + L append L as big-endian int
    Until(L+1+K+64%(384*2))==0
Step 3: Blocks
for each Block do
    W[64]
    W = W[0:15]+W[16:63]
    for i in 16:63 do
        $S_0 = (W[i-15] \circlearrowright 7) \oplus (W[i-15] \circlearrowright 18) \oplus (W[i-15] \ggg 3)$
        $S_1 = (W[i-2] \circlearrowright 17) \oplus (W[i-2] \circlearrowright 19) \oplus (W[i-2] \ggg 10)$
        w[i] = w[i-16] + $S_0$ + w[i-7] + $S_1$
Step 4: Compression loop
for i in 0:63 do
    $S_1 = (e \circlearrowright 14) \oplus (e \circlearrowright 18) \ggg (e \circlearrowright 41)$
    $ch = (e \wedge f) \oplus ((\neg e) \wedge g)$
    $temp_1 = H + S_1 + ch + k_{[i]} + w_{[i]}$
    $S_0 = (a \circlearrowright 28) \oplus (a \circlearrowright 34) \oplus (a \circlearrowright 39)$
    $maj = (a \wedge b) \oplus (a \wedge c) \oplus (b \wedge c)$
    $temp_2 = S_0 + maj$
Step 5: Hash Digest
for loop in LEN(Var) do
    $Var_{[a-h]} == Var_{[h-a]}$
    if $Var_{[loop]}$==‘e’ then
        $Var_{[loop]} = Var_{[loop]} + temp_1$
    if $Var_{[loop]}$==‘a’ then
        $Var_{[loop]} = Var_{[loop]} + temp_1 + temp_2$
    $H_{[1-7]} = H_{[1-7]} + Var_{[a-g]}$
    $digest = hash = hash + h_{0:5}$
```

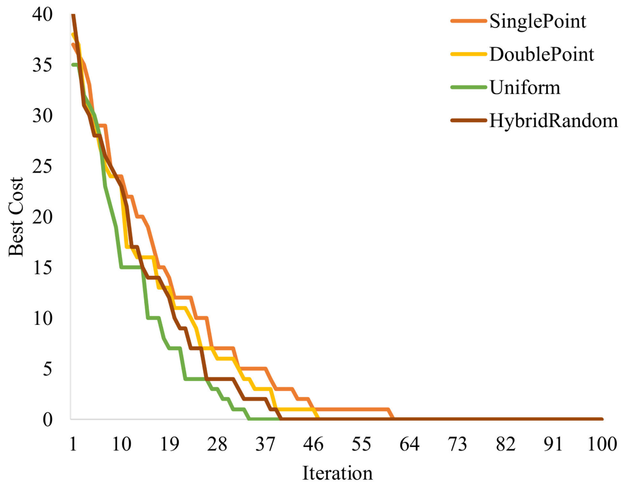

FIGURE 12. Block hashing architecture with GA Key and SHA-384.

processing time, hash rate and quality, and cost. The results are discussed in the subsequent subsections.

The experiment and simulation were performed using Python 3 on Windows (64-bit) platform, Intel Core i5 processor, 8 GB RAM. The dataset used in this process was from Kaggle and contain temperature readings from IoT devices installed outside and inside of anonymous buildings. The readings were captured during the alpha testing phase of the devices [49]. Thus, the devices were uninstalled or shut off several times during the entire reading period (07-28-2018 to 12-08-2018). We used this stream of data, which contained features and 97,605 records for data transmission.

Security enhancement in the proposed work is by adding an extra layer of encryption using EC-ElGamal. Transaction from inception to block hashing is encrypted and can only be decrypted by the participant node within the ecosystem.

A. METRICS EVALUATION

We performed an experiment to evaluate the following metrics: 1) transaction processing time, 2) block validation processing time, 3) hash rate, 4) hash quality, and 5) storage cost of the proposed method, and compared these metrics with existing approaches.

1) COMPARISON OF TRANSACTION FLOW

Transaction processing overhead is anticipated at the aggregator with an increase in IoT devices. An increase in transaction processing time eventually creates a bottleneck and causes the system to degrade. Our proposed method aims to increase transaction flow with a reduced processing time.

A transaction consists of IoT device data that were continuously pushed to the blockchain ecosystem. The size of the transaction packets were uniform, and we measured them as the cumulative sum of all transaction sizes in kB. In this study, we took 25 kB of data for transactions from a total of 18 transaction simulation points that were incrementally added and transferred.

We simulated a real-world scenario of transaction flow from devices through the transaction aggregator to the

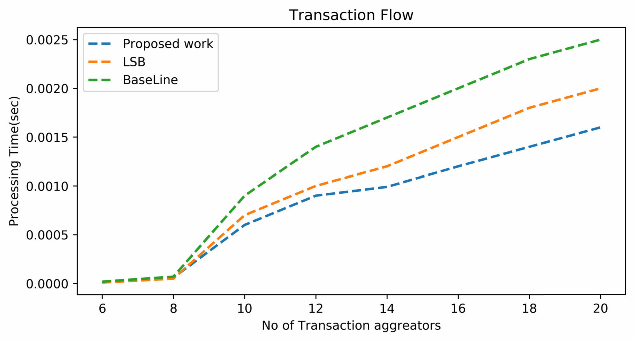

FIGURE 13. Comparison of Transaction Flow processing time between proposed and existing methods.

block managers. Processing time is calculated using (10).

$$\begin{aligned} &T_P = T_E - T_S \\ &\textit{where } T_P \textit{ is Transaction Processing time} \\ &T_E \textit{ is Process completion time} \\ &\textit{and } T_S \textit{ is Process start time} \end{aligned} \tag{10}$$

Existing methods demonstrated an increase in processing time when the transactions increased, whereas our proposed method displayed reduced processing time as in Fig. 13.

2) COMPARISON OF BLOCK VALIDATION

We measured the time taken by the proposed method to validate a new block, and plotted the average processing time of successfully verified blocks affixed to the blockchain in our simulation. The processing time was compared with that of other approaches, such as LSB and baseline approach. With our proposed system, it took 0.01 s for the creation of 10 blocks. Processing time is calculated using (11).

$$\begin{aligned} &BV_P = BV_E - BV_S \\ &\textit{where } BV_P \textit{ is Block validation Processing time} \\ &BV_E \textit{ is Block validation completion time} \\ &\textit{and } BV_S \textit{ is Block validation start time} \end{aligned} \tag{11}$$

The proposed method's turnaround time was significantly lower despite the increase in blocks from block managers, as illustrated in Fig. 14. It is thus evident that with our proposed method, the processing time for block validation was lower than for previous LSB methods. Hence, proving that the proposed method turn around time would be significantly less despite increase block flows from block managers.

3) COMPARISON OF HASH OPERATION

The measurement of the hash operation performed on an equal number of block cycles was performed using (12).

$$\begin{aligned} &H_O = \left(\frac{S_B}{PT_B}\right) \times No_H \\ &\textit{where, } H_O \textit{ is Hash operation} \\ &S_B \textit{ is Block size} \\ &PT_B \textit{ is Block processing time} \\ &\textit{and } No_H \textit{ is Number of Hash} \end{aligned} \tag{12}$$

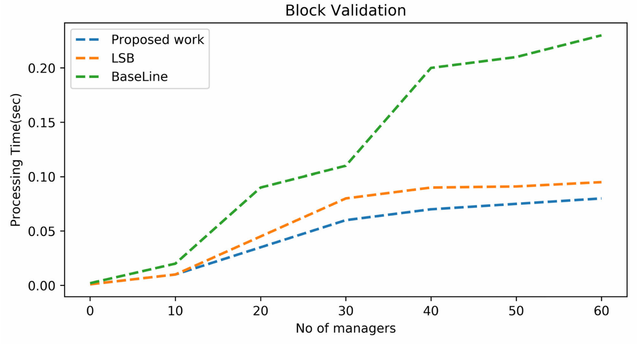

FIGURE 14. Comparison of Block Validation processing time between proposed and existing methods.

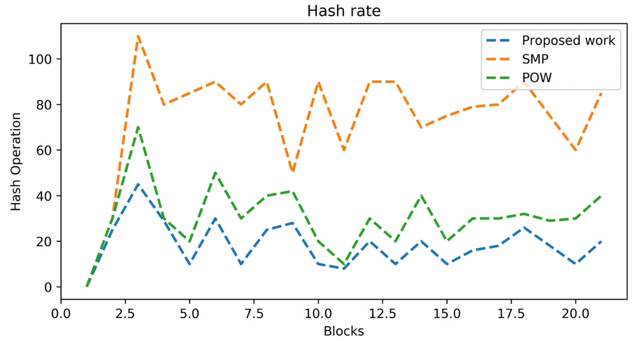

FIGURE 15. Comparison of Hash Operations between proposed and existing methods (SMP: synergistic multiple proof, POW: proof-of-work).

The proposed hash operation was compared with the synergistic multiple proof and proof-of-work, as illustrated in Fig. 15. The hash operation of the proposed method produced a better result than that of existing methods. The hashes were created using a combination of the Genetic algorithm and SHA-384 to make the operation faster and stronger.

4) COMPARISON OF HASH QUALITY

Hash quality for a block cycle is an indicator of the computation required in the ecosystem. The minimum hash quality was determined based on (13).

$$\begin{aligned} &H_Q = min\left(\frac{No_H}{C_H}\right) \\ &\textit{where, } H_Q \textit{ is Hash quality} \\ &No_H \textit{ is Number of hashes created and} \\ &C_H \textit{ is Hash creation } \cos t \end{aligned} \tag{13}$$

As illustrated in Fig. 16, the hash quality of the proposed method significantly increased in comparison to existing methods (synergistic multiple proof and proof-of-work). The proposed method of a combined hash thus provides better quality than existing hashing, which is based only on hexadecimal representation. This signifies a reduction in computing power in the peer-to-peer network.

5) COMPARISON OF STORAGE COST

The storage cost was computed by measuring the size of data stored in the memory layer. The storage cost per MB

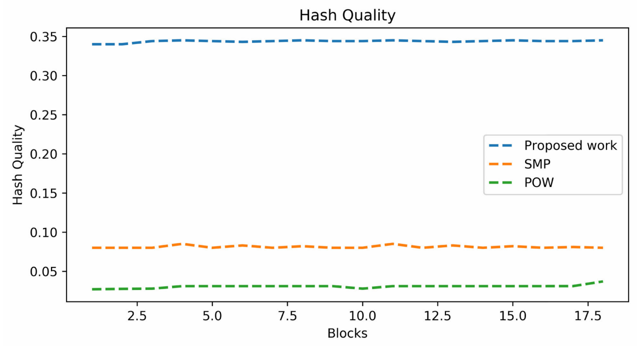

FIGURE 16. Comparison of Hash quality between proposed and existing methods (SMP: synergistic multiple proof, POW: proof-of-work).

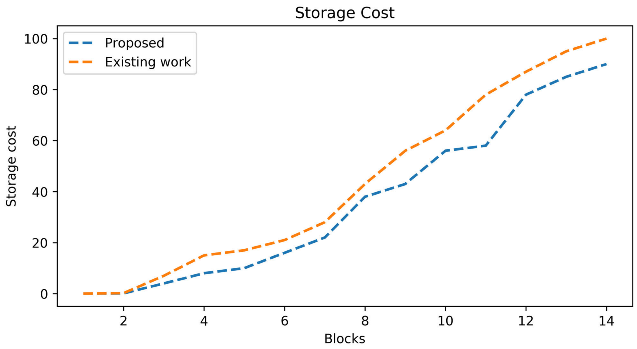

FIGURE 17. Comparison of Storage cost between proposed method and existing method.

for each block cycle was measured by (14) with a transaction of a 1 MB file and at a simulated 500 Transaction Per Second (TPS)

$$\begin{aligned} &C_s = C_B \times TC_B \\ &where,\ C_S\ is\ Cost\ of\ Storage \\ &C_B\ is\ Block\ cycle\ and \\ &TC_B\ is\ Time\ to\ create\ Blocks \end{aligned} \tag{14}$$

Fig. 17 presents the experimental results of a comparison between the proposed method and a baseline method. The proposed method showed reduction in storage cost.

The results indicate that increasing transaction flow with a shorter processing time, performing more block validation, and using a high hash rate all improve performance. The encryption of transactions using EC-ElGamal and the improvement of hash quality using a hybrid Genetic algorithm with SHA-384 thus support security enhancement.

V. CONCLUSION AND FUTURE WORKS

Based on the metrics from our experiments, the use of EC-ElGamal and Genetic algorithm based key for SHA-384 improved security and enhanced performance. The proposed system of transaction encryption using EC-ElGamal and block hashing with enhanced SHA-384 can thus be used to improve LSB for better adoption in blockchain-based IoT applications. Compared with the baseline method, the proposed technique implements enhanced technique for encryption and competitive achievements by 20% decrease in transaction processing time, 22% decrease in block validation processing time, 53% increase in hash rate, hash quality and optimized 7% saving on storage consumption.

TABLE 5. Future work that can be considered for improvising LSB.

Area	Focus	Outcome
Block Confirmation	Lightweight Consensus	Block validation time
Block Sync	Lightweight Consensus	Faster block sync
Security	Blockchain framework	Adaptable security
Hardware	Specialized hardware	Performance improvement
Network	Network latency	Reduce delays
Security	Physical Identification	Reduce computation

As displayed in Table 5, future work should focus on consensus, a security framework, hardware, networks, and physical identification mechanisms, leading to the following improvements. First, a lightweight consensus mechanism can further reduce the block validation time, while the use of a physical identification mechanism can reduce the computation. In addition, an adaptable security framework for a blockchain network can be considered a potential overall enhancement. Specialized hardware can be used to improve the processing time, and a new mechanism can be used to reduce the network delay. Innovations to implement these improvements can lead to ultra-lightweight high-performance blockchain suitable for all domains.

REFERENCES

[1] S. Cho and S. Lee, "Survey on the application of BlockChain to IoT," in *Proc. Int. Conf. Electron., Inf., Commun. (ICEIC)*, Jan. 2019, pp. 1–2, doi: 10.23919/ELINFOCOM.2019.8706369.

[2] T. Golomb, Y. Mirsky, and Y. Elovici, "CIoTA: Collaborative IoT anomaly detection via blockchain," 2018, *arXiv:1803.03807*. [Online]. Available: http://arxiv.org/abs/1803.03807

[3] M. A. Rashid and H. H. Pajooh, "A security framework for IoT authentication and authorization based on blockchain technology," in *Proc. 18th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun./13th IEEE Int. Conf. Big Data Sci. Eng. (TrustCom/BigDataSE)*, Aug. 2019, pp. 264–271, doi: 10.1109/TrustCom/BigDataSE.2019.00043.

[4] D. Puthal, S. P. Mohanty, P. Nanda, E. Kougianos, and G. Das, "Proof-of-Authentication for scalable blockchain in resource-constrained distributed systems," in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2019, pp. 1–5, doi: 10.1109/ICCE.2019.8662009.

[5] B. S. Balaji, P. V. Raja, A. Nayyar, P. Sanjeevikumar, and S. Pandiyan, "Enhancement of security and handling the inconspicuousness in IoT using a simple size extensible blockchain," *Energies*, vol. 13, no. 7, p. 1795, Apr. 2020. [Online]. Available: https://www.mdpi.com/1996-1073/13/7/1795

[6] X. Wu, F. Kong, J. Shi, L. Bao, F. Gao, and J. Li, "A blockchain Internet of Things data integrity detection model," in *Proc. Int. Conf. Adv. Inf. Sci. Syst. (AISS)*. New York, NY, USA: Association for Computing Machinery, Nov. 2019, pp. 1–7, doi: 10.1145/3373477.3373498.

[7] G. Sankar Ramachandran and B. Krishnamachari, "Blockchain for the IoT: Opportunities and challenges," 2018, *arXiv:1805.02818*. [Online]. Available: http://arxiv.org/abs/1805.02818

[8] S. Tuli, R. Mahmud, S. Tuli, and R. Buyya, "FogBus: A blockchain-based lightweight framework for edge and fog computing," *J. Syst. Softw.*, vol. 154, pp. 22–36, Aug. 2019.

[9] V. Rakovic, J. Karamachoski, V. Atanasovski, and L. Gavrilovska, "Blockchain paradigm and Internet of Things," *Wireless Pers. Commun.*, vol. 106, no. 1, pp. 219–235, May 2019, doi: 10.1007/s11277-019-06270-9.

[10] W. J. Buchanan, S. Li, and R. Asif, "Lightweight cryptography methods," *J. Cyber Secur. Technol.*, vol. 1, nos. 3–4, pp. 187–201, Oct. 2017, doi: 10.1080/23742917.2017.1384917.

[11] S. Biswas, K. Sharif, F. Li, B. Nour, and Y. Wang, "A scalable blockchain framework for secure transactions in IoT," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4650–4659, Jun. 2019, doi: 10.1109/JIOT.2018.2874095.

[12] O. Novo, "Scalable access management in IoT using blockchain: A performance evaluation," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4694–4701, Jun. 2019, doi: 10.1109/JIOT.2018.2879679.

[13] H. Moudoud, S. Cherkaoui, and L. Khoukhi, "An IoT blockchain architecture using oracles and smart contracts: The use-case of a food supply chain," in *Proc. IEEE 30th Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2019, pp. 1–6, doi: 10.1109/PIMRC.2019.8904404.

[14] S. K. T. Mehedi, A. A. M. Shamim, and M. B. A. Miah, "Blockchain-based security management of IoT infrastructure with ethereum transactions," *Iran J. Comput. Sci.*, vol. 2, no. 3, pp. 189–195, Sep. 2019, doi: 10.1007/s42044-019-00044-z.

[15] C. Ehmke, F. Wessling, and C. M. Friedrich, "Proof-of-property: A lightweight and scalable blockchain protocol," in *Proc. 1st Int. Workshop Emerg. Trends Softw. Eng. Blockchain (WETSEB)*. New York, NY, USA: Association for Computing Machinery, May 2018, pp. 48–51, doi: 10.1145/3194113.3194122.

[16] E. R. Naru, H. Saini, and M. Sharma, "A recent review on lightweight cryptography in IoT," in *Proc. Int. Conf. I-SMAC (IoT Social, Mobile, Analytics Cloud) (I-SMAC)*, Feb. 2017, pp. 887–890, doi: 10.1109/I-SMAC.2017.8058307.

[17] T. Bhattasali, "Licrypt: Lightweight cryptography technique for securing smart objects in Internet of Things environment," *CSI Commun.*, vol. CSIC-May(2013), pp. 26–36, May 2013.

[18] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "LSB: A lightweight scalable blockchain for IoT security and anonymity," *J. Parallel Distrib. Comput.*, vol. 134, pp. 180–197, Dec. 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0743731518307688

[19] H. Qiu, M. Qiu, G. Memmi, Z. Ming, and M. Liu, "A dynamic scalable blockchain based communication architecture for IoT," in *Smart Blockchain*. Cham, Switzerland: Springer, 2018, pp. 159–166, doi: 10.1007/978-3-030-05764-0_17.

[20] A. R. Shahid, N. Pissinou, C. Staier, and R. Kwan, "Sensor-chain: A lightweight scalable blockchain framework for Internet of Things," in *Proc. Int. Conf. Internet Things (iThings) IEEE Green Comput. Commun. (GreenCom) IEEE Cyber, Phys. Social Comput. (CPSCom) IEEE Smart Data (SmartData)*, Jul. 2019, pp. 1154–1161, doi: 10.1109/iThings/GreenCom/CPSCom/SmartData.2019.00195.

[21] A. R. Shahid, N. Pissinou, L. Njilla, E. Aguilar, and E. Perez, "Demo: Towards the development of a differentially private lightweight and scalable blockchain for IoT," in *Proc. IEEE 16th Int. Conf. Mobile Ad Hoc Sensor Syst. Workshops (MASSW)*, Nov. 2019, pp. 172–173, doi: 10.1109/MASSW.2019.00045.

[22] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for IoT security and privacy: The case study of a smart home," in *Proc. IEEE Int. Conf. Pervas. Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2017, pp. 618–623, doi: 10.1109/PERCOMW.2017.7917634.

[23] B. Seok, J. Park, and J. H. Park, "A lightweight hash-based blockchain architecture for industrial IoT," *Appl. Sci.*, vol. 9, no. 18, p. 3740, Sep. 2019. [Online]. Available: https://www.mdpi.com/2076-3417/9/18/3740

[24] Y. Liu, K. Wang, Y. Lin, and W. Xu, "LightChain: A lightweight blockchain system for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3571–3581, Jun. 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8664132/

[25] F. H. Pohrmen, R. K. Das, W. Khongbuh, and G. Saha, "Blockchain-based security aspects in Internet of Things network," in *Advanced Informatics for Computing Research*. Singapore: Springer, 2019, pp. 346–357, doi: 10.1007/978-981-13-3143-5_29.

[26] F. Moradi, A. Sedaghatbaf, S. A. Asadollah, A. Causevic, and M. Sirjani, "On-off attack on a blockchain-based IoT system," in *Proc. 24th IEEE Int. Conf. Emerg. Technol. Factory Autom. (ETFA)*, Sep. 2019, pp. 1768–1773, doi: 10.1109/ETFA.2019.8868238.

[27] R. Xu, Y. Chen, E. Blasch, and G. Chen, "Microchain: A hybrid consensus mechanism for lightweight distributed ledger for IoT," 2019, *arXiv:1909.10948*. [Online]. Available: http://arxiv.org/abs/1909.10948

[28] M. A. Uddin, A. Stranieri, I. Gondal, and V. Balasurbramanian, "A lightweight blockchain based framework for underwater IoT," *Electronics*, vol. 8, no. 12, p. 1552, Dec. 2019. [Online]. Available: https://www.mdpi.com/2079-9292/8/12/1552

[29] M. Y. Ary Saputro and R. F. Sari, "Securing IoT network using lightweight multi-fog (LMF) blockchain model," in *Proc. 6th Int. Conf. Electr. Eng., Comput. Sci. Informat. (EECSI)*, Sep. 2019, pp. 183–188, doi: 10.23919/EECSI48112.2019.8976914.

[30] O. Abdulkader, A. M. Bamhdi, V. Thayananthan, F. Elbouraey, and B. Al-Ghamdi, "A lightweight blockchain based cybersecurity for IoT environments," in *Proc. 6th IEEE Int. Conf. Cyber Secur. Cloud Comput. (CSCloud)/5th IEEE Int. Conf. Edge Comput. Scalable Cloud (EdgeCom)*, Jun. 2019, pp. 139–144, doi: 10.1109/CSCloud/EdgeCom.2019.000-5.

[31] O. Novo, "Blockchain meets IoT: An architecture for scalable access management in IoT," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1184–1195, Apr. 2018, doi: 10.1109/JIOT.2018.2812239.

[32] R. Xu, Y. Chen, E. Blasch, and G. Chen, "BlendCAC: A blockchain-enabled decentralized capability-based access control for IoTs," in *Proc. IEEE Int. Conf. Internet Things (iThings) IEEE Green Comput. Commun. (GreenCom) IEEE Cyber, Phys. Social Comput. (CPSCom) IEEE Smart Data (SmartData)*, Jul. 2018, pp. 1027–1034, doi: 10.1109/Cybermatics_2018.2018.00191.

[33] B. Mbarek, N. Jabeur, T. Pitner, and A.-U.-H. Yasar, "MBS: Multilevel blockchain system for IoT," *Pers. Ubiquitous Comput.*, pp. 1–8, Nov. 2019, doi: 10.1007/s00779-019-01339-5.

[34] X. Xu, Z. Zeng, S. Yang, and H. Shao, "A novel blockchain framework for industrial IoT edge computing," *Sensors*, vol. 20, no. 7, p. 2061, Apr. 2020, doi: 10.3390/s20072061.

[35] V. Dedeoglu, R. Jurdak, G. D. Putra, A. Dorri, and S. S. Kanhere, "A trust architecture for blockchain in IoT," in *Proc. 16th EAI Int. Conf. Mobile Ubiquitous Systems: Comput., Netw. Services*. New York, NY, USA: Association for Computing Machinery, Nov. 2019, pp. 190–199, doi: 10.1145/3360774.3360822.

[36] Y. Yu, S. Zhang, C. Chen, and X. Zhong, "LVChain: A lightweight and vote-based blockchain for access control in the IoT," in *Proc. IEEE 4th Int. Conf. Comput. Commun. (ICCC)*, Dec. 2018, pp. 870–874, doi: 10.1109/CompComm.2018.8780687.

[37] I. Ali, R. J. ul Hussen Khan, Z. Noshad, A. Javaid, M. Zahid, and N. Javaid, "Secure service provisioning scheme for lightweight clients with incentive mechanism based on blockchain," in *Advances on P2P, Parallel, Grid, Cloud and Internet Computing*. Cham, Switzerland: Springer, 2020, pp. 82–93, doi: 10.1007/978-3-030-33509-0_8.

[38] R. Doku, D. B. Rawat, M. Garuba, and L. Njilla, "LightChain: On the lightweight blockchain for the Internet-of-Things," in *Proc. IEEE Int. Conf. Smart Comput. (SMARTCOMP)*, Jun. 2019, pp. 444–448, doi: 10.1109/SMARTCOMP.2019.00085.

[39] R. A. Michelin, N. Ahmed, S. S. Kanhere, A. Seneviratne, and S. Jha, "Leveraging lightweight blockchain to establish data integrity for surveillance cameras," 2019, *arXiv:1912.11044*. [Online]. Available: http://arxiv.org/abs/1912.11044

[40] R. D. Caytiles and B. Park, "ECC based authentication scheme for securing data contents over open wireless network systems," *J. Adv. Inf. Technol. Converg.*, vol. 8, no. 2, pp. 1–11, Dec. 2018. [Online]. Available: http://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE07588707&language=ko_KR

[41] V. Soi, B. S. Dhaliwal, and M. Kumar, "ECC algorithm for WSN," *Int. J. Eng. Sci.*, vol. 6, no. 6, pp. 125–130, Oct. 2017. [Online]. Available: https://www.ijesm.co.in/abstract.php?article_id=3443&title=ECC%20Algorithm%20for%20WSN

[42] Y. Liu and S. Zhang, "Information security and storage of Internet of Things based on block chains," *Future Gener. Comput. Syst.*, vol. 106, pp. 296–303, May 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X19316784

[43] F. Mallouli, A. Hellal, N. Sharief Saeed, and F. Abdulraheem Alzahrani, "A survey on cryptography: Comparative study between RSA vs ECC algorithms, and RSA vs el-gamal algorithms," in *Proc. 6th IEEE Int. Conf. Cyber Secur. Cloud Comput. (CSCloud)/ 5th IEEE Int. Conf. Edge Comput. Scalable Cloud (EdgeCom)*, Jun. 2019, pp. 173–176, doi: 10.1109/CSCloud/EdgeCom.2019.00022.

[44] R. K. Ansah, S. Effah-Poku, D. A. Addo, B. A. Adjei, B. K. Bawuah, and P. Antwi, "Relevance of elliptic curve cryptography in modern-day technology," *J. Math. Acumen Res.*, vol. 3, no. 2, pp. 1–10, Aug. 2018. [Online]. Available: https://jomaar.org/index.php/jmr/article/view/76

[45] R. Martino and A. Cilardo, "SHA-2 acceleration meeting the needs of emerging applications: A comparative survey," *IEEE Access*, vol. 8, pp. 28415–28436, 2020, doi: 10.1109/ACCESS.2020.2972265.

[46] Y. Luo, X. Ouyang, J. Liu, and L. Cao, "An image encryption method based on elliptic curve Elgamal encryption and chaotic systems," *IEEE Access*, vol. 7, pp. 38507–38522, 2019, doi: 10.1109/ACCESS.2019.2906052.

[47] M. Ragavan and K. Prabu, "Dynamic key generation for cryptographic process using genetic algorithm," *Int. J. Comput. Sci. Inf. Secur.*, vol. 17, no. 4, pp. 246–250, 2019.

[48] S. M. Myint, M. M. Myint, and A. A. Cho, "A study of SHA algorithm in cryptography," *Int. J. Trend Sci. Res. Develop.*, vol. 3, no. 5, pp. 1453–1454, 2019.

[49] *Temperature Reading From IoT*. Accessed: Apr. 1, 2020. [Online]. Available: https://www.kaggle.com/atulanandjha/temperature-readings-iot-devices

J. GURUPRAKASH is currently a Research Scholar with the School of Information Technology and Engineering, Vellore Institute of Technology, Vellore, India, with more than 15 years of experience in IT industry and teaching. His research interests include blockchain technologies, the IoT, and AI/ML.

SRINIVAS KOPPU received the M.Tech. degree under the major of information technology with specialization in human–computer interaction from IIIT Allahabad, India, and the Ph.D. degree from the Vellore Institute of Technology (VIT), Vellore, India. He has been working as an Associate Professor with the School of Information Technology and Engineering, VIT. He has more than 12 years of experience in teaching. He has published more than 30 international/national journals and conferences. He is a Life Member of the Computer Society of India. His research interests include blockchain technologies, the IoT, data analytic, cryptography, medical image processing, image security analysis, video processing and computer vision, and high-performance computing.