A Blockchain Framework in Post-Quantum Decentralization

Rahul Saha, Gulshan Kumar, Tannishtha Devgun, William J. Buchanan, Reji Thomas, Mamoun Alazab, Tai Hoon-Kim, and Joel J. P. C. Rodrigues, *Fellow, IEEE*

Abstract—The decentralization and transparency have provided wide acceptance of blockchain technology in various sectors through numerous applications. The claimed security services by blockchain have been proved using various cryptographic techniques, mainly public key infrastructure and digital signatures. However, the use of generic cryptographic primitives using large prime numbers or elliptic curves with logarithms is going to be an issue with quantum computers as those techniques are vulnerable in post-quantum era. Therefore, the paradigm shift from pre-quantum to the post-quantum era has necessitated new cryptographic developments which are robust against quantum attacks and applicable in blockchain for post-quantum decentralization. Therefore, we have presented a solution for post-quantum decentralization in the blockchain. It uses lattices with polynomials for identity-based encryption (IBE) and aggregate signatures for the consensus to ensure efficiency and suitability in post-quantum blockchain applications. We experiment the proposed approach based on delay, throughput, energy consumption and complexity. The comparative results prove that the presented work is efficient.

Index Terms—Cryptography, blockchain, post-quantum, security, signature, identity

1 Introduction

Blockchain technology, a process of storing digital information (block) in a public ledger (chain) is one of the most promising technologies developed at the beginning of the 20th century. In 2001, a group of researchers conceptualize the framework of the blockchain to time-stamp the digital documents that remains idle for almost 7 years [1]. Satoshi Nakamoto further exploits the time-stamp features of this technology in Bitcoin during the 2008-2009 period. Subsequently, bitcoin and its underlying blockchain becomes popular. Besides, this technology gains more attention in information technology and research interest exponentially increases to exploit it further in various relevant applications. Even though, blockchain product (bitcoin) has become viral before the technology behind it initially, blockchain technology's real potential unearthed by those researches. Various technologies are trying their best to adapt with blockchain technology in the present scenario. The advantageous features of blockchains include, decentralized, distributed, secure services that are faster, transparent and immune and hence, 'blockchain technology' is an enabler of present and future technologies [2].

To realize the simplicity in the blockchain architecture, it considers a shift from centralized to the decentralized one initially [3], [4]. Blockchain is nothing but a 'shared registry' or 'distributed ledger' that accounts for the information about all assets in either form of tangible (application-specific) or intangible (as any digital data) and their transactions/movements in a Peer-to-Peer (P2P) network. In this technology, cryptographic primitives make the transactions secured, histories are grouped, and information is stored in a block. Such blocks are then linked further with cryptographic hashes to prevent any modification of the stored data. In short, this overall architectural and functional specifications avoid forgery and ensure immunity to the transactions across the network [5]. Additionally, one of the best advantages of blockchain is the trust mechanism in which the mutual distrust of the blockchain participants makes the blockchain secured. With more and more original research and extensions of the existing methods, various categories of blockchain exist; we can classify them into permissioned and permissionless categories. These architectural variations of blockchain have produced various applications in different domains, viz., cybersecurity, agriculture, online data storage, networking and IoT, multimedia, supply chain

- *Rahul Saha and Gulshan Kumar are with the School of Computer Science and Engineering, Lovely Professional University, Punjab 144001, India. E-mail: {rsahaaot, gulshan3971}@gmail.com.*
- *Tannishtha Devgun is with the Nokia Solutions and Networks, Karnal, Haryana 132001, India. E-mail: jmd.tannishtha@gmail.com.*
- *William J. Buchanan is with the Blockpass ID Lab, Edinburgh Napier University, EH11 4BN Edinburgh, U.K. E-mail: B.Buchanan@napier.ac.uk.*
- *Reji Thomas is with the Divison of Research and Development, Lovely Professional University, Punjab 144001, India. E-mail: rthomas.eyyalil@gmail.com.*
- *Mamoun Alazab is with the Charles Darwin University, Casuarina 0810, Australia. E-mail: alazab.m@ieee.org.*
- *Tai Hoon-Kim is with the Glocal Campus of Konkuk University, 268, Chungwon-daero, Chungju-si, Chungcheongbuk-do 27478, South Korea. E-mail: taihoonn@daum.net.*
- *Joel J. P. C. Rodrigues is with the Federal University of Piauí, 64049-550 Teresina, PI, Brazil, and also with the Intituto de Telecomunicações, 6201-01 Covilhã, Portugal. E-mail: joeljr@ieee.org.*

Manuscript received 9 June 2021; revised 9 September 2021; accepted 26 September 2021. Date of publication 30 September 2021; date of current version 6 February 2023.
This work was supported in part by FCT/MCTES through National Funds and when applicable co-funded EU funds under Grant UIDB/EEA/50008/2020 and in part by the Brazilian National Council for Scientific and Technological Development (CNPq) under Grant 313036/2020-9.
(Corresponding authors: Gulshan Kumar and Tai Hoon-Kim.)
Digital Object Identifier no. 10.1109/TSC.2021.3116896

management, crowdfunding, real estate and many other sectors of the industry to name a few [6]. It has revolutionized the current business models of many organizations [7].

However, being a part of networking technology with heavy traffic, blockchains also have inherent risk concerns, and these challenges need to be addressed in the fast-growing information technology field [8]. Cryptography, as the core part of blockchains, is a prime factor to tackle insecurity. The two basic cryptographic features used in blockchains are: public-key cryptography/or asymmetric cryptography and hash functions [9]. All these generic cryptographic algorithms use large odd prime numbers, such as Rivest–Shamir–Adleman (RSA) algorithm or discrete logarithms (eg. Elliptic Curve Cryptography). The advent of quantum computers puts all such algorithms at risk as the prime factorization or discrete logarithm problems can be easily cracked to obtain the private keys using Shor's algorithm and its future qubit variations [10], [11], [12]. This can lead to the collapse of profound security of blockchain and its sustainability in the ultra-sensitive technology. Researchers realize this imminent threat to this flamboyant technology and they attempts to explore its feasibilities in the post-quantum period [13]. Furthermore, the use of certificateless cryptography approaches is also proving benefits in terms of avoiding centralized key generation failure and security [14], [15]. Therefore, in our solution we have used this concept with quantum layout.

The objective of the present study is to identify the cryptographic building blocks that can be embedded in blockchain technologies to be applicable in a post-quantum regime. Therefore, the work considers a solution for the blockchain framework that can provide security and robustness in post-quantum computing. The contribution in our present work are as follows:

- The presented post-quantum resistant blockchain framework addresses the insecurity problem of the existing frameworks that use generic cryptographic methods.
- Our solution generates the keys from the wallet identities transforming the email addresses of the wallets to rings. Lattice-based cryptographic primitives are used for aggregate signatures. We use Certificateless Identity-Based Encryption (Certificateless IBE) with lattices to solve the key-escrow problem.
- This solution provides significant improvements in terms of time and storage consumption and complexity. Additionally, the security of blockchain has been improved with the use of lattices.

The rest of the paper is organized as follows. Section 2 reviews some recent developments in the related field. Section 3 shows the proposed framework and its functionalities. We perform a security analysis of the approach and explain in Section 4. Experimental setup and performance analysis is shown in Section 2. Conclusion and overall findings are drawn in Section 6.

2 Background and Related Works

In this section, we review some of the important research works in recent times. As the present work uses the concept of aggregate signature and identity-based encryption in related blockchain developments, we discuss the related literature divided into the following three subsections.

2.1 Aggregate Signatures

An aggregate signature concept uses a compact signature for n distinct messages from n distinct signers [16]. As the verification of the single compact signature ensures the validity of all the signers, storage and bandwidth reduction become the straight forward benefits. Therefore, aggregate signatures are great help in the blockchain. There are two primary mechanisms for the signature aggregation; general and sequential schemes. In general signature aggregation scheme, each user from the group of users creates a signature s_i on his/her own message $\mathcal{M}_i$. Anyone then runs public aggregation algorithm, takes all n signatures $s_1, s_2, \ldots, s_n$ and compress them into a single compact signature $\ddot{s}$. On the other hand, sequential signature aggregation scheme deals with signatures from the users in a forward feed mode where user 1 signs $\mathcal{M}_1$ to obtain s_1, user 2 then uses s_1 and $\mathcal{M}_2$ to obtain s_2 and, so on until the final signature s_n is generated by n^{th} user. Various extensions and advancements have been observed in aggregate signatures in the recent past. Some aggregate signature variations and its applications can be seen elsewhere [17], [18], [19]. A CertificateLess Aggregate Signature (CLAS) scheme must have a short signature size and the aggregation must be easy to follow [17]. The scheme is based on bilinear maps and helps in reducing storage cost. However, such traditional cryptography-based CLAS schemes are not so much secure against fully chosen-key attacks [18]. The solution is also shown for the same to withstand such attack. An Elliptic Curve Cryptography (ECC) based CLAS scheme is also noteworthy [19]. It provides low overhead and privacy assurance. Similar approach is also shown in [20]. Identity-based aggregate signatures are discussed in a recent literature [21]. With the urge of post-quantum needs, some lattice-based aggregate signature methods are also noteworthy to be mentioned [22], [23], [24], [25]. The security of the method shown in [22] is based on worst-case lattice problems. It follows a sequential structure to aggregate the signatures and output one signature. Another such method is applied in [23]. This scheme avoids using a signer to retrieve the keys of other signers. It verifies the aggregate-so-far before adding its own signature. A solution for the universal forgery attack against NTRU-based Structure-free Compact Rapid Authentication (SCRA) method is observed [24]. In [25], authors show an anti-quantum lattice-based blind signature scheme. The scheme is applicable to blockchains. It improves the security in post-quantum era. The blindness of the scheme ensures anonymity. Another such method is shown in [26].

2.2 Identity-Based Encryption

Identity-Based Encryption (IBE) is a type of public-key cryptography introduced in 1984 [27]. Such encryption systems use a public key from a known identity of the receiver. Three variants are available of IBE: certificate-based encryption, secure key issuing cryptography and certificateless cryptography. Some of the recent developments of IBE contribute significantly to the literature. Various mathematical

models including bilinear pairing, polynomial constructions and lattices for IBEs are discussed in [28], [29], [30], [31]. Signcryption with Equality Test (SCET) is shown in [28]. It uses equality test to IBE construction. The approach in [29] uses accountability with distributed Public Key Generator (PKG) to detect the pirated key and its creator. A concrete construction of the method combines the advantages of both distributed PKGs and accounted IBE. Using the similar approach with broadcast is shown in [30]. A generic construction of Leakage Resistant IBE (LR-IBE) scheme is noteworthy in this direction [31]. It provides security against Chosen Ciphertext Attack (CCA). It follows the concept of Identity-based Hash Proof System (IB-HPS).

Some lattice-based constructions for IBEs also provide significant contribution in this research. A recent approach uses a proxy-oriented identity-based encryption with keyword search (PO-IBEKS) scheme [32]. It is based on lattices and useful for cloud storage. The use of lattice makes it post-quantum secure. Key revocation is another important feature for Revocable IBE (RIBE). Such a development is shown in [33]. It is able to provide Decryption Key Exposure Resistance (DKER) without using re-randomization of keys.

Certificateless IBEs also draw major attraction and some significant works are described in [34], [35], [36], [37]. In [34], a Searchable Public-key Encryption (SPE) for smart healthcare is shown. This scheme avoids using the secure channels and is able to provide resistance against key guessing attack. Certificateless Encryption (CLE) combines the advantages of PKI-based public-key encryption and IBEs. Therefore, research for certificateless fully homomorphic encryption (CLFHE) is also under exploration as homomorphic systems are able to execute different functions on encrypted data [35]. CLFHE is semantically secure based on Learning With Errors (LWE) problem in the random oracle model. However, their scheme supports only homomorphic addition, but not homomorphic multiplication. A lightweight searchable encryption method based on certificateless cryptosystem is researched in [36]. Such kind of approach along with privacy preservation is explained in [37].

2.3 Blockchain Developments

The main functioning modules of blockchains include the creation of blocks with hash and previous hash and a consensus protocol. In these functionalities, cryptography plays an important role. The public-private key pairs, hash functions, signatures are those entities that make the blockchain secure and reliable for providing decentralized transparency [38]. Various applications and developments in blockchain in recent years are discussed elsewhere [5], [7]; however, the core framework research for blockchain is still in infancy. Blockchains are still in the process of using generic cryptographic primitives [9] and ready to move forward to the post-quantum time by upgrading the security and robustness of underlying cryptographic usage. Such possibilities are shown in the recent literature [39], [40], [41]. Authors show a new lattice-based signature scheme in [39]. Bonsai Trees technology creates public-private keys. RandBasis algorithm is used along with this to generate the keys from the root keys ensuring the randomness. It also affirms the lightweight non-deterministic wallets. Another conceptual model for a quantum blockchain is shown in [40]. It uses a temporal GHZ (Greenberger–Horne–Zeilinger) state of photons for encoding utilizing the quantum advantage. Another significant work is observed in [41]. In this, the authors show a blockchain platform combining the original BFT state-machine replication without use of digital signatures and Quantum Key Distribution (QKD) for providing authentication. The platform is experimented in an urban QKD network.

The above discussion of the literature shows that there is lack of research in the development of the post-quantum resistant blockchain frameworks. This has motivated us for the present solution. Though separately lattice signatures, certificateless cryptography are used in blockchains, the combination of these two is not explored yet. Therefore, in our present work we have developed the framework using lattice based aggregate signature with IBE. The framework novelties are as follows.

- Certificateless IBE : IBE is exiting in literature; however, the certificateless IBE for blockchain using the quantum attributes for blockchains is new.
- Aggregate signature: The use of aggregate lattice-based signature for blockchains are also new in this direction.
- Results: The comparative results show the efficiency of our solution based on throughput, delay, energy consumption and complexity.

3 Proposed Framework

In this section, we discuss about lattices, polynomial rings related to lattices with computationally hard problems. This is followed by a network model for the presented work and the proposed scheme.

3.1 Preliminaries

We consider a polynomial over a ring R as a formal sum $a_n^{x_n} + \ldots + a_1 x + a_0$, where the coefficients come from the ring R. $R[x]$ denotes the set of all polynomials (in one variable) over a ring R. The degree of the polynomial is the highest power of x with a non-zero coefficient. A lattice $\mathcal{L}$ for a ring of n real numbers R^n is a discrete subgroup of R^n [42], [43]. We consider only integer lattices and therefore, $\mathcal{L} \subseteq \mathbb{Z}^n$. We define it as a set of points in n-dimensional space with a periodic structure. Given a n linearly independent vectors $\{b_1, b_2, \ldots, b_n\} \in R^n$, the lattice generated by them is a set of vectors mathematically shown as

$$\mathcal{L}(b_1, b_2, \ldots, b_n) = \sum_{i=1}^{n} x_i b_i, x_i \in \mathbb{Z}^n. \tag{1}$$

The vectors $\{b_1, b_2, \ldots, b_n\}$ generating a $\mathcal{L}$ called as bases of lattice $\mathcal{L}$.

Let K be a field denoted as: $R = K[t], F = K(t)$, where R is the polynomial ring and F is the rational function field in the indeterminate t over K. For any rational function, $x = \frac{a}{b} \in F$ and $a, b \in R$ and $b \neq 0$, we can value of F as

$$v(\infty)(x) = \begin{cases} deg\ b - deg\ a,\ if\ x \neq 0, \\ \infty,\ if\ x = 0 \end{cases}. \tag{2}$$

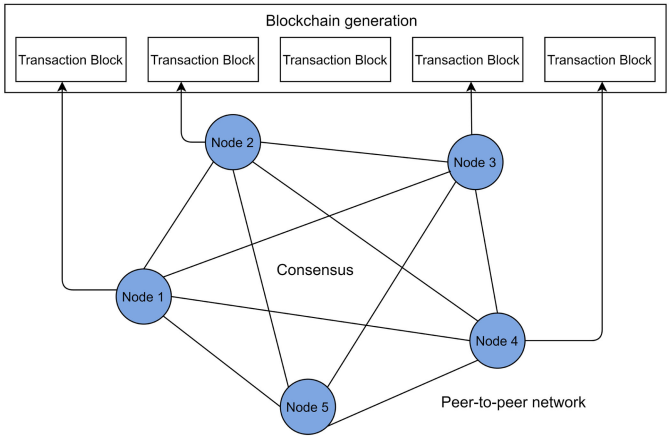

Fig. 1. Logical relations among blockchain entities.

This is a discrete valuation on F with a valuation ring $R_\infty = K[t^{-1}]_{t^{-1}} \subset F$ and the maximum ideal $P_\infty = t^{-1}R_\infty$. Considering a principle ideal domain $\mathcal{D}$ with field of fractions $F_\mathcal{D} \subset F_\infty$. The typical instances can be produced as: $\mathcal{D} = R, R_\infty$, or , $\widehat{R_\infty}$. These ideals are important for lattice-based cryptographic developments. It uses the ideals in rings $\mathbb{Z}[x]/\langle f \rangle$, where f is an irreducible polynomial of degree n. For experimentation, the most popular ring used is: $R_q = \frac{\mathbb{Z}_q[x]}{x^n+1}$, where n is the power of two and the prime q is calculated as $q \equiv 1 \ mod \ 2n$.

3.2 Network Model

The proposed solution for blockchain consists of the following types of entities. The structure of the proposed scheme is shown in Fig. 1.

Peer-to-Peer Network. The blockchain network is peer-to-peer to provide decentralized functionalities. Any node can be a participant of the network and can be connected with others.

Nodes. The nodes are users. Depending upon the application, the nodes may vary in different profiles like organizations, clients, and various stakeholders.

Consensus. To validate a new transaction in the blockchain, consensus to be applied. It makes the decision transparent and preventive from compromised activities on blockchain such as decision manipulation problems and byzantine problems .

The connections among these entities are shown in Fig. 1. It shows an example of peer-to-peer blockchain network of five nodes. The nodes generate public-private keys of their own and public keys are broadcast in the blockchain network. Each node uses the keys for transaction block generation (black solid arrows) and to take participation in consensus. The consensus validates the blocks for adding in the blockchain.

We generalize the network model with n number of nodes. it uses IBE concept for generating public keys and private keys tailored with nodes' email-ids and a unique id transformed into polynomials.

3.3 Framework Functionalities

The overall functionalities of the proposed framework are segregated in a series of stages: setup and key generation, IBE-based transaction block generation and consensus with aggregate signature. We provide the description of various stages involved in the process as in the following.

3.3.1 Set Up and Key Generation

Each node chooses two polynomials $f, f' \in R_q$ with degree at most $n-1$. The polynomial $f \in \mathcal{L}(f)$ must satisfy the additional condition that the inverse modulo q and modulo p exist, i.e., $f.f_p = 1(mod\ p)$ and $f.f_q = 1(mod\ q)$ must hold for f. A temporary variable is created as: $temp \leftarrow pf_q.f'\ (mod\ q)$. Each node then computes its own private key Pr_{u_i} and public key Pu_{u_i} as summarized in Algorithm 1. The nodes uses IBE-based key generation concept here. It uses the email address for this purpose. Email addresses are transformed into polynomials, say $\mathcal{P}_1$ with generator g_1. The node chooses a random lattice r′ from $\mathcal{L}(\mathcal{P}_1)$. It then generates its private key as: $Pr_{u_i} = r^{'}.\mathcal{P}_1(mod\ q)$ and public key as: $Pu_{u_i} = temp.\mathcal{P}_1(mod\ q)$. Public keys are published in the blockchain network.

Algorithm 1. Set Up and Nodes' Key Generation

1: **Input:** $f, f', Email\ address$
2: **Output:** Pu_{u_i}, Pr_{u_i}
3: Choose $f, f' \in R_q$
4: **if** $(f.f_p = 1\ (mod\ p)\ and\ f.f_q = 1\ (mod\ q))$
5: **Then**
Go to step 6
Else
Go to step 1
6: $temp \leftarrow pf_q.f'\ (mod\ q)$
7: Convert $\{Email\ address\} \rightarrow \mathcal{P}_1$
8: Choose a random lattice r'
9: $Pr_{u_i} = r^{'}.\mathcal{P}_1\ (mod\ q)$
10: $Pu_{u_i} = temp.\mathcal{P}_1\ (mod\ q)$
11: Publish Pu_{u_i}

3.3.2 Transaction Block Generation

As per the general blockchain functions, whenever a any node u_i initiates a transaction, it uses hash and its private key to generate a signature. The transaction pools are having some transactions signed by various entities and are assumed to be the candidates for a particular block. Generally, in the blockchain system, miners create a new block and in the proposed scheme, nodes are having the capability of mining. Therefore, nodes can add a new block in the proposed blockchain framework. The overall process is shown in Algorithm 2.

To create a transaction block, the node first collects the transactions and creates a random oracle. This model works as a hash that outputs uniformly the bits in the digest creating a set of binary vectors $\mathbb{B}_n^k$ of n vectors each of k weight. Thus, in a block of transactions, the digest (δ) is generated as:

$\delta_i = H(B_i)$; B_i is the block of transactions of i^{th} node, $H()$ is the has function $\in$ Random oracle i^{th} node signs this digest δ_i. The node first samples a lattice vector y from m-dimensional discrete Gaussian distribution D_σ^m. It first computes the hash: $h = H(Pu_{u_i}\ y\ mod\ 2q, \delta_i)$, then samples b bits, and computes the output: $\beta = y + (-1)^b Pr_{u_i} h$. After the rejection sampling, the signature output becomes $s =$

(β, h) with a probability of $\frac{1}{\left(Mexp\left(-\frac{||Pr_{u_i}h||^2}{(2\sigma^2)}\right)cos\ h\left(\frac{<\beta, Pr_{u_i}h>}{\sigma^2}\right)\right)}$. For the probability calculation, we have used the knowledge base as explained in [44].

Algorithm 2. Block Generation

Input: $\mathcal{T}_i$ transactions, Pu_{u_i}, q
Output: Signature s
Collect the transaction $\mathcal{T}_i \rightarrow B_j$
Select $H : \{0,1\}^n \in \mathbb{B}_n^k$
Calculate $\delta_i = H(B_i)$
Sample $y \leftarrow$ discrete Gaussian distribution $\mathcal{D}_\delta^m$
Compute $h_i = H(Pu_{u_i} y \ mod\ 2q, \delta_i)$
Sample b bits
Compute $\beta_i = y + (-1)^b Pr_{u_i} h$
If Prob$(\beta, h) = \frac{1}{\left(Mexp\left(-\frac{||Pr_{u_i}h||^2}{(2\sigma^2)}\right)cos\ h\left(\frac{<\beta, Pr_{u_i}h>}{\sigma^2}\right)\right)}$
Then return $(s_i = (\beta_i, h_i))$
Else recalculate
Broadcast (B_j, s_i) in the blockchain network

3.3.3 Consensus Participation With Aggregate Signature

All the nodes considered here are the participants of the consensus algorithm. In a blockchain network of N nodes, a node can receive $N-1$ transaction blocks from others at an instant t. Overall, n blocks can be there in the network at a time instant without any block issue rate condition. To execute the consensus, each block needs to be verified by all the other peers (nodes) of the network. Therefore, a single node can verify $n-1$ blocks separately (except its own block). To utilize the resource more effectively, the proposed framework uses the concept of aggregate signature concept. Though aggregate signature for blockchain applications is shown in [45], the lattice framework for aggregation is the novelty of the present work. In this concept, all the nodes have the ability to aggregate the signature and broadcast it to the other peers (nodes) in the network. Upon receiving the aggregate signature from the nodes, the receiving peers verify it. For this purpose, each node is able to run the aggregate algorithm to aggregate all the signatures arrived at time t into a single one. The receivers use the function $aggrverify()$ to verify the aggregate signature so that all the signatures can be verified simultaneously. As a result, the verification of $(n-1)$ signatures can be done with one aggregation leading to the reduced overhead in the network. Thus, it helps to provide scalability to the system as the aggregation is a single operation for multiple blocks and nodes. The public keys are available in the blockchain network which is used for this process. For aggregation, each node considers a tuple information piece as: Pu_{u_i}, B_j, s_i followed by an unordered aggregation [46]. With this, the aggregate function constructs the lattice polynomials as

$$\begin{cases} s = s_1 \ mod\ \mathcal{L}_1 \\ s = s_2 \ mod\ \mathcal{L}_2 \\ \quad\quad \dots \\ s = s_i \ mod\ \mathcal{L}_i \end{cases}, \tag{3}$$

where, $\mathcal{L}_1 + \mathcal{L}_2 + \ldots + \mathcal{L}_i = \mathbb{Z}^n$ and, $\mathcal{L}_1 \cap \mathcal{L}_2 \cap \ldots \cap \mathcal{L}_i = \mathcal{L}_q^\perp(A); A \in \mathbb{Z}^n$. With these equations s is calculated. Then sampling is executed following a gaussian distribution with $(\mathfrak{B}_a, v_a, -s)$ where $\mathfrak{B}_a$ is short basis on $\mathcal{L}_q^\perp(A)$ with $||B'|| \leq O(n\ log\ q)$. With this distribution s_0 is calculated and finally the aggregated signature as: $s_a = s_0 + s$. The process is shown in Algorithm 3.

Algorithm 3. Aggregate Signature in Consensus

Input: Pu_{u_i}, B_j, s_i
Output: s_a
Construct i lattice polynomials
Calculate s
Sample a gaussian distribution with $(\mathfrak{B}_a, v_a, -s)$
Calculate s_0
Calculate $s_a = s_0 + s$
Return s_a

Algorithm 4. Verification of Aggregate Signature

Input: $s_a, Pu_{u_i}, B_j, s_i; i = 1, 2, .., n-1$
Output: Aggregate signature validation
Select $H' : \{0,1\}^n \in \mathbb{B}_n^k$
Calculate $s_a \leq v_a\sqrt{n}$
Calculate $w = H'[H(B_1), H(B_2), .., H(B_j)]$
Calculate $w' = H'[Pu_{u_1}(s_a, mod\mathcal{L}_1)modq,$
$Pu_{u_2}(s_a mod\ \mathcal{L}_1)modq, \ldots, Pu_{u_{n-1}}(s_a, mod\mathcal{L}_{n-1})]$
If $(w = w')$
{s_a is valid, Consensus agreed }
Else
{ s_a is invalid, Abort the block }
return NULL

Once the receiving peers receive the aggregated signature, they run the $aggrverify()$ as shown in Algorithm 4. Once the aggregate signature is validated, the consensus reaches a final decision for adding the newly initiated block in the blockchain. All the peers update its database as the blocks get updated in the blockchain. This provides consistency. The overall process is shown in Fig. 2. It shows a proposed framework example with three nodes for the simplicity in the representation. Node 3 initiates block and the other two nodes aggregates and verifies accordingly. Keys are generated by the nodes itself and public keys are broadcast in the network.

4 Security Analysis

The security of lattice-based solutions depends on solving some problems known as 'lattice problems'[47]. These problems are assumed to be hard to solve with a polynomial-time solution. The three important problems considered in the recent work are mentioned below. *Shortest Vector Problem (SVP):* Given a lattice basis $\mathfrak{B}$, it is hard to find the shortest non-zero vector in the lattice $\mathcal{L}(\mathfrak{B})$ with a polynomial-time algorithm. *Closest Vector Problem (CVP):* For a given lattice $\mathcal{L} \subset \mathbb{Z}_q$ and a target $\mathcal{T} \in \mathbb{Z}_q$, it is computationally hard to find lattice vector $b_i \in \mathcal{L}$ with $||\mathcal{T} - b_i|| = dist(\mathcal{T}, \mathcal{L})$. *Learning with Errors (LWE):* n, q two numbers, $\mathfrak{D}$ is the probability distribution on $\mathbb{Z}$ and s' is a secret vector in $\mathbb{Z}_q^n$. With these considerations the probability of

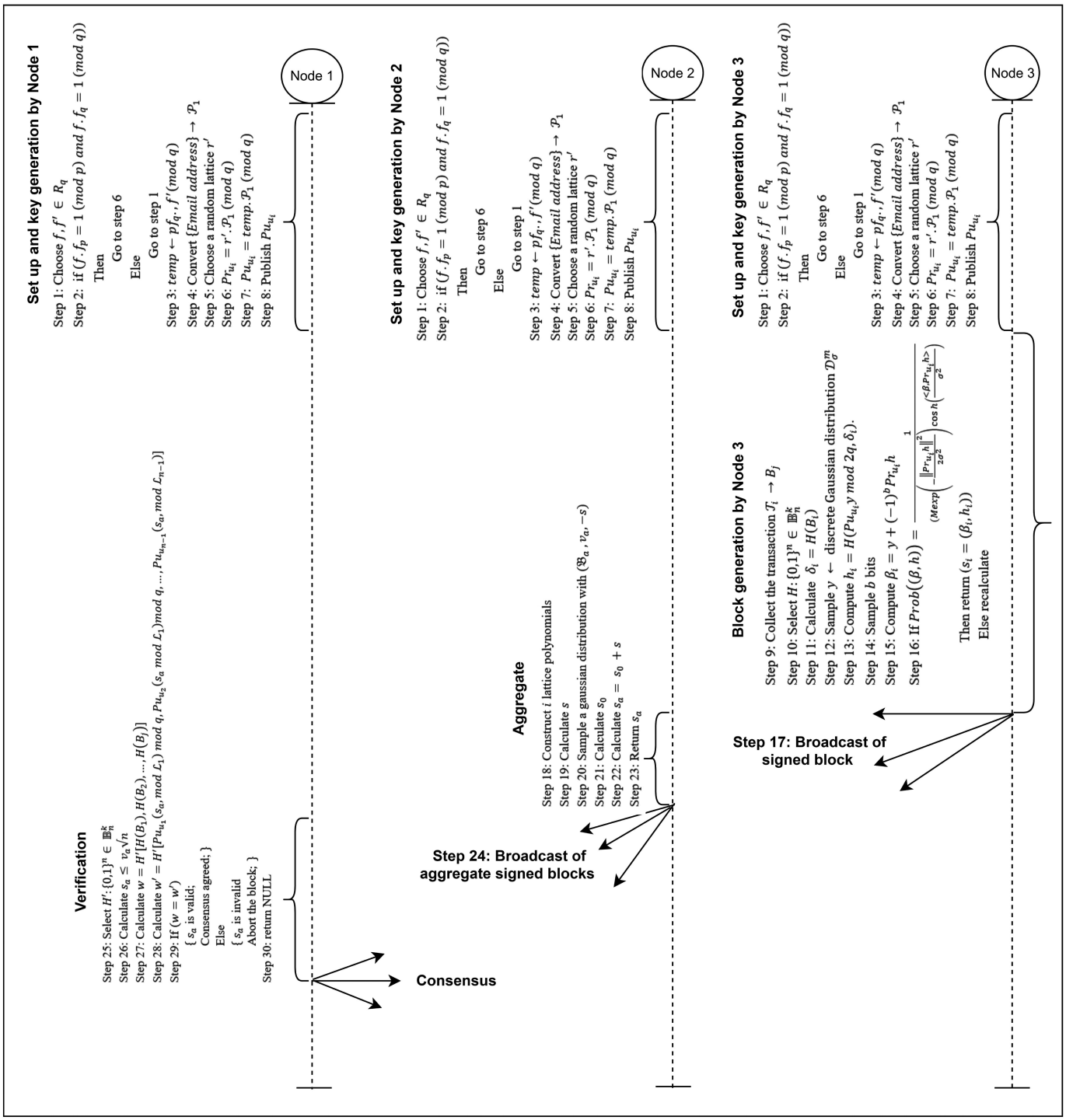

Fig. 2. Sequence of computation processes in the proposed framework.

distribution on $\mathbb{Z}_q^n \times \mathbb{Z}_q$ is $P_{s',\mathfrak{D}}$ obtained from a random $a \in \mathbb{Z}_q^n$ and $e \in \mathbb{Z}_q$ with the distribution of $\mathfrak{D}$ and calculating for $(a, b) = (a,\ \langle a, s^{'} \rangle\ + e) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$. The variation of LWE, Ring-LWE (RLWE) is more useful in the present work to analyse. Here, the polynomials a_i, e_i and $s^{'}$ are drawn from a ring $Z[x]/\langle f \rangle$, where f is an irreducible polynomial of degree n.

The security proof of the lattice-based system with respect to the above three problems are already mentioned in the literature [47]. Following the same, the proposed work is also not having any polynomial-time computation. In the process considered to solve the above problems is thus secure. Further, we analyze the correctness and unforgeability characteristics of the framework.

Correctness. The individual node generates signature $s_i = (\beta_i, h_i)$. It is statistically drawn from Gaussian distribution D_σ^m with $\sigma = \frac{2^{-\omega(logm)M}}{M}$. With this deviation in D_σ^m, the probability of s_i is calculated as: $P(s_i) \leq 2\sigma\sqrt{m_i}$ with an overwhelming output. Following the proposed approach, for any random and uniformly distributed matrix $M \in \mathbb{Z}_q^{n \times m}$, $(M_i, s_i) = H_1(B_i) mod\ q$ and $||s_i|| \leq v_a\sqrt{n}, M_i$ is the block B_i.

Thus, the individual signature verification is expected to be reasonable and correct.

The solution to the aggregate signature polynomials is a vector s which is not short. To shorten it, Gaussian sampling is executed and s_0 is calculated which also satisfies the condition of $||s_0|| \leq s_a\sqrt{n}$, where s_a has the same coset as s . Thus, s_a also satisfies the aggregate signature polynomial equation. Therefore, $s_a \ mod \ \mathcal{L}_i = s_i \ mod \ \mathcal{L}_i$ leads to the equality of the following:

$$H^{'}[H(B_1), H(B_2), \ldots, H(B_j)] = H^{'}[Pu_{u_1}(s_a \ mod \ \mathcal{L}_1) mod \ q, Pu_{u_2}(s_a \ mod \ \mathcal{L}_1) mod \ q, \ldots, Pu_{u_{n-1}}(s_a \ mod \ \mathcal{L}_{n-1}) \ mod \ q]$$

Unforgeability. The proposed work ensures unforgeability with a chosen message attack (UF-CMA). Unforgeability is to be proved by calculating the advantage of an adversary $\mathcal{X}$ which is having a non-negligible advantage against the UF-CMA security of the proposed framework. If the advantage is negligible, it means the Short Integer Solution (SIS) problem cannot be solved by the attacker and the proposed framework is unforgeable.

On a hypothetical contradiction, it is assumed that the advantages for solving UF-CMA and SIS are $\mathcal{A}$ and $\mathcal{A}'$ and both are non-negligible. An algorithm υ exists that uses the trapdoor function $Trapgen(q, n)$ to generate $A \in \mathbb{Z}_q^{n \times m}$. First, υ sets the master public key as A and master secret key as $T_A.A$ then provides the public parameters to $\mathcal{X}$. After the queries to the random oracle, $\mathcal{X}$ can output a forged signature which does not match with the tuples stored by the υ. As a result, υ aborts the further processing of signature verification. The probability of getting an advantage by the attacker $\mathcal{X}$ becomes $2^{-\omega log n}$ which is negligible. It means $\mathcal{X}$ is unable to solve SIS problem in the proposed method, leading to significant UF-CMA security. The process of [40] has been executed for the purpose and detailed mechanisms can be followed from the same. From the above discussion, it is clear that the proposed framework ensures security through lattice-based constructions.

5 Results and Discussion

The presented decentralized framework of blockchain using lattice-based aggregation and IBE is the first attempt for the post-quantum computing blockchain scenario. Therefore, the comparison of the experimented results has been done between the proposed framework and a hypothetical framework without aggregation and IBE process. Furthermore, some recent attempts of developing post-quantum resistant blockchains are also compared as per the existing works mentioned in [25], [39], [41]. All these approaches show the feasibility of post-quantum sustainable blockchain frameworks. These methods are simulated on the same platform and the experimental setup as of the proposed approach. The implementation process, related results of our solution are explained in the following subsections.

5.1 Implementation Process

Table 1 describes the implementation of the framework [49]. An Ethereum network with solidity contract and Remix IDE has been used for the compilation of PoS. The implementation steps are given below.

TABLE 1
Implementation Framework

Consensus protocol	Proof of Stake (PoS)
Geographic distribution of nodes	Ethereum network
Hardware environment of all peers	3.6 GHz, 16 GB RAM, Octa-core, 2 TB HDD
Number of nodes involved in the test transaction	Simulated nodes 50
Test tools and framework	Hyperledger Caliper
Type of data store used	LevelDB

Step 1: Pre-installation of Homebrew [50] and Node/npm [51].
Step 2: Installation of Ethereum, Solidity, Remix IDE [52] and Microsoft LatticeCrypto Library [53].
Step 3: Genesis blocks are initialized.
Step 4: The blockchain is initialized with 10 blocks.
Step 5: A folder is created for the blockchain to reside.
Step 6: Ethereum is initiated and run with lattice cryptographic signatures and aggregates
Step 7: Geth Javascript console is used to connect to the Ethereum blockchain.
Step 8: Account has been created and dummy Ethers are mined.
Step 9: PoS is initialized
Step 10: Remix IDE is initialized to deploy the generated PoS.
Step 11: Remix IDE is updated with wallet account of the gradual nodes' information in the Ethereum network.
Step 12: PoS is executed on Ethereum blockchain.

5.2 Performance Metrics

The systems' performance to generate overall 100 blocks of transactions with 50 nodes is measured. Resource consumption of the system is calculated. We evaluate various timing measurements and throughput as mentioned in Table 2. Additionally, we evaluate the space complexity.

Transaction latency starts from the time of submission to the point and is widely available in the network. It includes the propagation and intermediating settling times due to the adopted consensus mechanism in the model. Transaction throughput is not considered for a single node rather, it is observed for the overall blockchain network for all the nodes to perform the transactions properly.

5.3 Performance Analysis

The first metric that has been considered to evaluate is the 'time' or 'latency' requirements. We measure the timing parameter for 100 blocks all are generated by the new nodes joining in the blockchain. We show the results in Table 3.

As can be seen in Table 3, time consumption increases with the increasing number of block generation. The rate of increasing factor is linear and therefore, it may be efficient in blockchain applications. Note that, the main reason of this linear increasing factor of latency is the inclusion of key generation time and signature time together. The key

TABLE 2
Measurement Parameters

Performance parameter	Definition	Formula for calculation used
Read latency	Time calculated between submission of a read request and receipt of a reply. In our experimentation for read latency, we consider the time from key generation to the aggregate signature verification.	Read Latency = Time when the response received – submit time
Read Throughput	The number of read operations completed in a defined time period, expressed as reads per second (RPS).	Read Throughput = Total read operations / total time in seconds
Transaction Latency	Time taken for a transaction's effect to be usable across the network.	Transaction Latency = (Confirmation time @ network threshold) – submit time
Transaction Throughput	The rate at which valid transactions are committed by the blockchain in a defined time period. This rate is expressed as transactions per second (TPS) at a network size.	Throughput = Total committed transactions / total time in seconds @ #committed nodes

TABLE 3
Proposed Approach Performance

	Number of blocks										
	10	20	30	40	50	60	70	80	90	100	complexity
Read latency (seconds)	12.33	14.8	15.56	18.01	24.50	25.33	27.80	28.67	31.88	34.23	$O(log\ n^2)$
Transaction latency (seconds)	46.43	78.70	111.33	173.33	208.23	260.47	317.63	352.03	371.33	410.02	$O((N-1)log\ n))$
Space complexity						$O(nN)$					

n is the number of blocks; N is the number of nodes involved in the transaction of blocks. The assumption of the internet speed is 1.2 Mbps.

generation time increases with the number of blocks increase which is $O(N)$ as N is the number of nodes. Assuming that all the N nodes are new to the network and able to issue the blocks at the same time; the nodes require keys, and the key generation time increases. Once the nodes get the keys in the first epoch; the other epochs the latency decreases or becomes almost stable. We show this effect in Fig. 3. We run four epochs of transactions; we use the term epoch for iterations. In the first epoch, average latency is 31.7 seconds, In the second epoch where some nodes are new and some nodes already have the keys, the average latency is 21.2 seconds. Similarly, in epoch three and epoch four the average latencies reduce to 14.2 seconds and 9.1 seconds respectively. Moreover, the aggregate signature time is static as its a single operation of signature generation and verification and thus, the complexity is $O(1)$. Even though, number of nodes or blocks increase, it does not affect the aggregate signature process. Therefore, our proposed approach is scalable with the condition that we have to optimize the key generation time further for better effect. Table 3 also shows about storage complexity and is less as compared to the storage complexity $O(N^n)$ of the non-aggregated approach.

We also show the comparative results for read latency in Fig. 4. It depicts that the read latency of our framework is much lower than existing approaches. The proposed framework uses aggregate signature and lattice-based cryptography which is faster than the other generic algorithms and

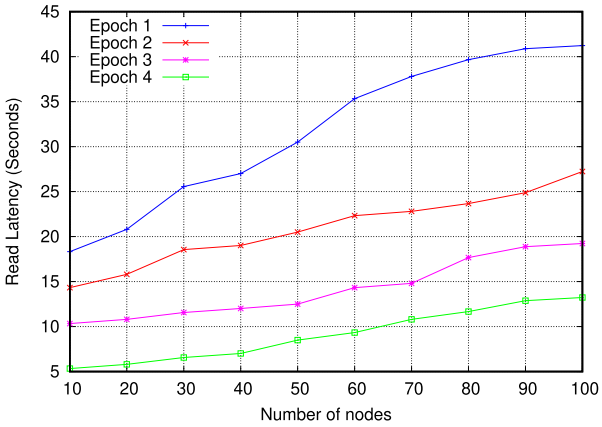

Fig. 3. Epochs and read latency.

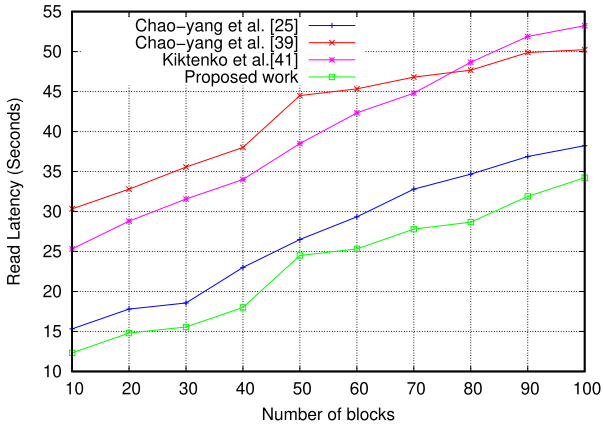

Fig. 4. Comparison of read latency.

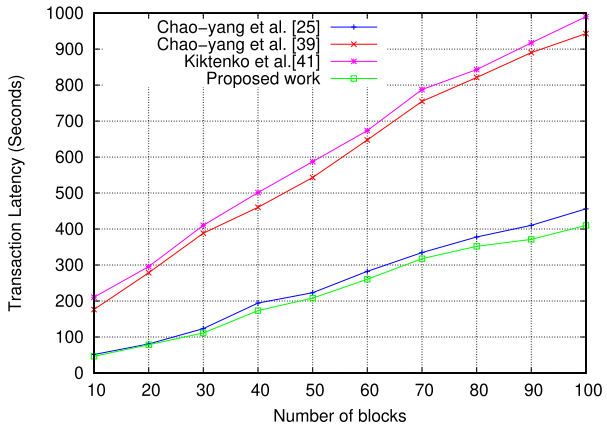

Fig. 5. Comparison of transaction latency.

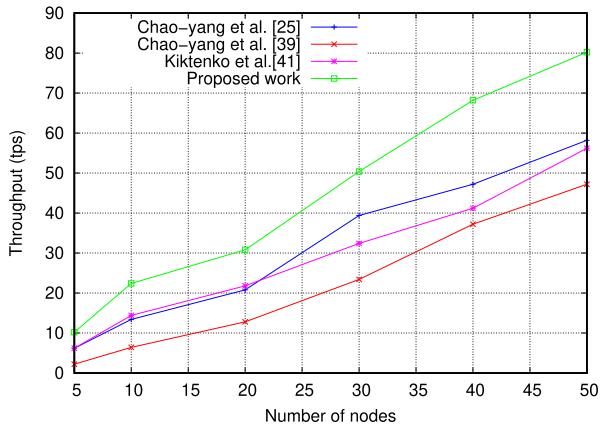

Fig. 7. Throughput comparison.

therefore it can produce approximately 60.5% faster in reading the blocks. The works in [25] also produces similar latency as our approach is almost in the same direction. However, the blindness of signature is not applied in our method. The other approaches have higher latency and therefore, our approach is better in performance for read latency parameter. This read latency also affects transaction latency as it includes this read latency too and we show the corresponding results in Fig. 5. Transaction latency also follows the same behaviour of the read latency where our proposed solution performs better as compared to existing approaches.

In the next experimentation, we measure the throughput (read throughput and transaction throughput) as per the definitions given in Table 2. The corresponding results are shown in Fig. 6. It shows that the read throughput for the non-aggregated blockchain framework starts with 7 tps and ends up to 57 tps with an average increase of 21.5% reads per second. The proposed framework with aggregated features considerably shows better features by having on average of 40% more reads per second. It signifies that the proposed framework is faster than the existing generic framework of blockchain. Moreover, it continues the exploration of throughput feature considering the overall transactions' throughput. It can be seen that the proposed method is 60% better in overall transactional throughput.

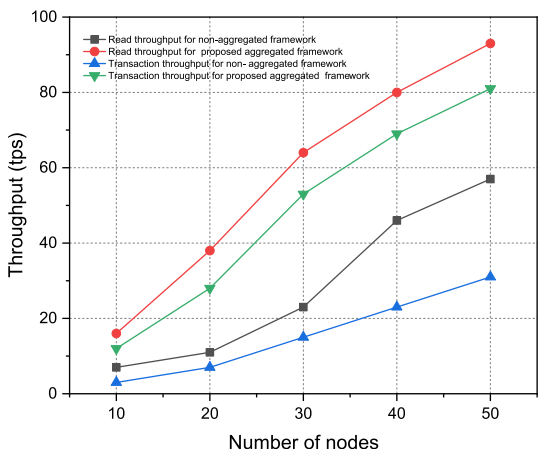

Fig. 6. Throughput measurements for read and overall transaction.

The use of the aggregated scheme in consensus makes it efficient in this aspect. Furthermore, we also measure the transaction throughput (as it includes the read of the blocks) with increasing number of nodes for state-of-the-art approaches in comparison. We show the comparative analysis in Fig. 7. It shows that, throughput for our proposed approach is better by 34% on average compared to other methods. This average is calculated by adding up the all the methods together. It is due to the less complex signature and aggregation of the signature. It helps in reducing the latency for reads and writes and more number of blocks can be successfully committed.

The traditional blockchain framework consumes more energy due to its key generation, block generation with hashes and consensus. The use of Proof of State (PoS) is more likely to reduce energy consumption to some extent. Proof of Stake (POS) is reported as less risky in terms of the potential for miners to attack the network, as it features compensation in a way that makes an attack less advantageous for the miner. Therefore, from miners perspectives consensus is not vulnerable. Moreover, the lattice construction of the frame uses less energy in the process. The energy consumption is calculated on each node with the number of transactions and averaged in the plot. The node wise results are shown in Fig. 8. Fig. 8a depicts the energy consumption plots for a blockchain framework using a generic public key structure with IBE. The key generation for all the nodes produces a static energy consumption in the process showing a static line with 1.8% of the total energy. Energy for blockchain in the same is higher with a minimum near about 2.2% and a maximum of 6.23% with an average increasing step of 0.8%. The consensus is higher energy consumable as it starts with 4.1% increasing up to 11.2%. Similarly, Fig. 8b shows the energy consumption for the proposed framework. It depicts that the lattice key generation consumes less energy maintaining a static line with 1.1%. Finally, the consensus shows the benefits of using aggregate signatures and verification by the average energy consumption from 1.8% to 6.4% which is approximately 40% less from the former approach. Overall, the average energy consumption is 47.4% less as compared to the framework without aggregate signature and lattice construction. Fig. 8c shows the average total energy consumption of both methods. We also compare the energy consumption of all the approaches. We

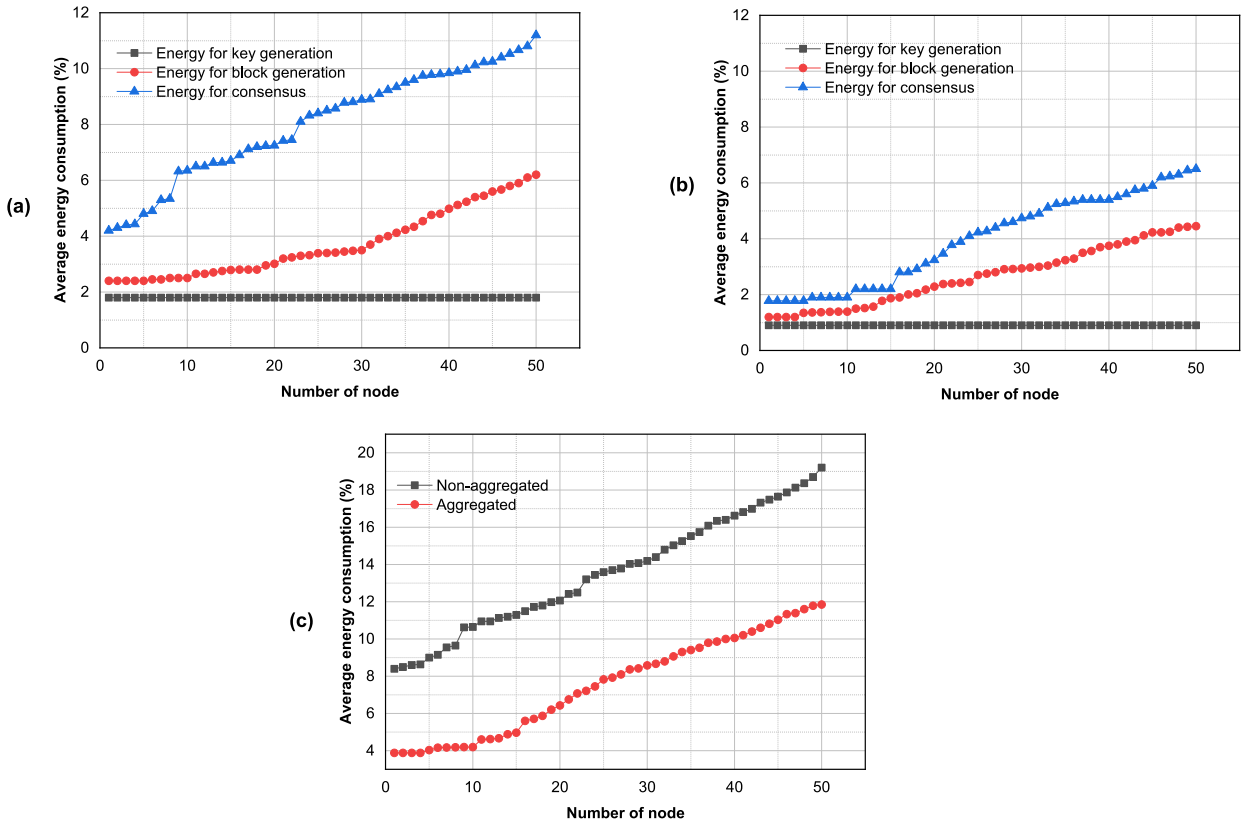

Fig. 8. Node wise average energy consumption: a) Non-aggregated generic blockchain framework with IBE, b) Proposed aggregated blockchain framework and c) Comparison of average total energy consumption.

calculate the node's energy consumption by 100 transactions of each node and then we average it and converted into percentage as per their initialized power base. The comparative result is shown in Fig. 9. It shows that our proposed approach is 30% better in terms of energy consumption on average.

The next measurement is based on cryptographic complexities. To measure the complexities, some notations are used as shown in Table 4. The approximate time for calculation of the functions is also shown for the ease of complexity computation. For IBE, pairing is transformed into polynomials for synchronized comparison. The comparison of the complexities is shown in Table 5. In this table, n is the number of transactions with signatures.

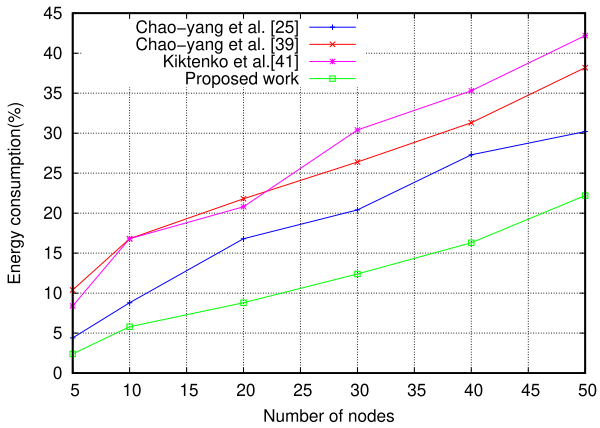

Fig. 9. Comparison of energy consumption percentage.

The comparison of the complexity shows that the complexity of the proposed work is less than the others. The blinding and unblinding iterations in [25] causing extra complexity to the system; other approaches use a longer process of key generation and signing process, and therefore, the complexity increases. Even the aggregation is also not supported by the approaches in comparison. Our proposed framework with aggregate lattice-based signature completes the cryptographic functions by 25% reduced time which is significant for blockchain applications, specifically for post-quantum regime. The complexity analysis is computed based on the algorithms shown in the respective literature. This analysis signifies that the proposed blockchain framework is efficient in terms of latency, throughput, cryptographic computation, and energy consumption.

5.4 Solution Scalability

From the comparative analysis and the experimental results we can observe that our solution is not imposing any serious overhead on the transactions. The complexity analysis

TABLE 4
Notations for Cryptography Complexity Measurement

Function	Notation	Approximate time
Polynomial selection	T_P	0.014s
Lattice creation	T_L	0.128s
Multiplication in ring	T_*	0.118s
Ring addition	T_+	0.008s
Hash time	T_H	0.012s

TABLE 5
Comparison of Computational Complexity

Framework	Cryptographic Computational cost		
	Key generation	Signature	Verification
Chao-yang *et al.* [25]	$4T_P + 4T_* + nT_L$	$2T_P + 2nT_* + 4T_H + 2nT_+$	$nT_P + 4nT_* + nT_H + 2nT_+$
Chao-yang *et al.* [39]	$4T_P + 4T_* + 2nT_L$	$4T_P + 2nT_* + 2T_H + 4T_+$	$nT_P + 4nT_* + 2nT_H + nT_+$
Kiktenko *et al.* [41]	$2T_P + 2T_* + 2nT_L$	$2T_P + 2nT_* + 4T_H + 2nT_+$	$nT_P + 4nT_* + 2nT_H + nT_+$
The Proposed framework	$2T_P + T_L + 3T_*$	$2T_P + 4T_* + 2T_H + nT_L + T_+$	$T_P + 2T_H + (n-1)T_L$

for read latency and transaction latency are also improved in our method. We show that the use of aggregate signature does not create any overhead or any increased time consumption; N number of nodes have N signatures, which we can aggregated as a single function and also verify as a single function. Therefore, its complexity is negligible. All these aspects also show the direction of the inferiority of our solution towards scalability of the framework. Generally, the blockchain architectures lack behind with scalability issues. In our present solution too, this issue may occur at a later stage but, scalability comparison of the similar approaches tell us that our approach is better in handling scalability.

5.5 Future Directions

In the present solution we have used the generic blockchain structures which is considered as the generation 1 of Distributed Ledger Technology. The scalability is an issue which draws the researchers towards the more scalable options of Directed Acyclic Graph (DAG) options such as IOTA. Its true that the drawbacks of the generation 1 blockchain architecture can be removed in generation 2 and generation 3 DLTs, in such cases the applicability of the proposed solution remains same. Moreover, the use of lattice cryptography is able to provide more scalability in DLTs like DAGs.

6 Conclusion

We develop a decentralized blockchain framework for post-quantum computing. The proposed framework uses identity-based encryption and aggregate signatures based on lattices. We analyze latency, throughput and complexity of the framework. The obtained results are superior to the conventional non-aggregated approach. We also study some existing approaches that provide the similar objectives and try to simulate their concept on the same platform as of ours. Further, the security analysis of the framework confirms the resistance of the framework against insecurity of post-quantum. The use of lattice has helped significantly in reducing the time and storage; hence, our framework is suitable for devices with low energy resources. This attempt of using blockchain along with aggregated lattice potentials is the first attempt in the direction of post-quantum decentralization. We consider the optimization of key generation process and trust management as some significant future directions to include in the present work.

References

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2009. [Online]. Available: https://metzdowd.com

[2] S. Allidina, "The future of blockchain in 8 charts," Accessed: Jun. 27, 2016. [Online]. Available: https://www.raconteur.net/the-future-of-blockchain-in-8-charts/

[3] M. Belotti, N. Božić, G. Pujolle, and S. Secci, "A vademecum on blockchain technologies: When, which, and how," *IEEE Commun. Surv. Tut.*, vol. 21, no. 4, pp. 3796–3838, Oct.–Dec. 2019.

[4] T. A. Syed, A. Alzahrani, S. Jan, M. S. Siddiqui, A. Nadeem, and T. Alghamdi, "A comparative analysis of blockchain architecture and its applications: Problems and recommendations," *IEEE Access*, vol. 7, pp. 176 838–176 869, 2019.

[5] A. M. Saghiri, "Blockchain architecture," in *Advanced Applications of Blockchain Technology*. Singapore: Springer

[6] J. A. Jaoude and R. G. Saade, "Blockchain applications – Usage in different domains," *IEEE Access*, vol. 7, pp. 45360–45381, 2019.

[7] D. DiF. Maesa and P. Mori, "Blockchain 3.0 applications survey," *J. Parallel Distrib. Comput.*, vol. 138, pp. 99–114, 2020.

[8] N. Drljevic, D. A. Aranda, and V. Stantchev, "Perspectives on risks and standards that affect the requirements engineering of blockchain technology," *Comput. Standards Interfaces*, vol. 69, 2020, Art. no. 103409.

[9] L. Wang, X. Shen, J. Li, J. Shao, and Y. Yang, "Cryptographic primitives in blockchains," *J. Netw. Comput. Appl.*, vol. 127, pp. 43–58, 2019.

[10] P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *Proc. Annu. Symp. Found. Comput. Sci.*, 1994, pp. 124–134.

[11] Z. Kirsch and M. Chow, "Quantum computing: The risk to existing encryption methods," Tufts Univ., Medford, MA, USA, Tech. Rep., 2015.

[12] W. Buchanan and A. Woodward, "Will quantum computers be the end of public key encryption?," *J. Cyber Secur. Technol.*, vol. 1, no. 1, pp. 1–22, 2017.

[13] T. M. Fernández-Caramès and P. Fraga-Lamas, "Towards post-quantum blockchain: A review on blockchain cryptography resistant to quantum computing attacks," *IEEE Access*, vol. 8, pp. 21091–21116, 2020.

[14] A. W. Dent, "A survey of certificateless encryption schemes and security models," *Int. J. Inf. Secur.*, vol. 7, pp. 349–377, 2008.

[15] G. Zhang and X. Wang, "Certificateless encryption scheme secure in standard model," *Tsinghua Sci. Technol.*, vol. 14, no. 4, pp. 452–459, Aug. 2009.

[16] D. Boneh, "Aggregate signatures," in *Encyclopedia of Cryptography and Security*. H. C. A. van Tilborg and S. Jajodia, Eds. Boston, MA, USA: Springer, pp. 27–29, 2011.

[17] K. Hashimoto and W. Ogata, "Unrestricted and compact certificateless aggregate signature scheme," *Inf. Sci.*, vol. 487, pp. 97–114, 2019.

[18] G. Wu, F. Zhang, L. Shen, F. Guo, and W. Susilo, "Certificateless aggregate signature scheme secure against fully chosen-key attacks," *Inf. Sci.*, vol. 514, pp. 288–301, 2020.

[19] J. Cui, J. Zhang, H. Zhong, R. Shi, and Y. Xu, "An efficient certificateless aggregate signature without pairings for vehicular ad hoc networks," *Inf. Sci.*, vol. 451-452, pp. 1–15, 2018.

[20] I. A. Kamil and S. O. Ogundoyin, "An improved certificateless aggregate signature scheme without bilinear pairings for vehicular ad hoc networks," *J. Inf. Secur. Appl.*, vol. 44, pp. 184–200, 2019.

[21] Y. Yao, Z. Li, and H. Guo, "A unified framework of identity-based sequential aggregate signatures from 2-level HIBE schemes," *Inf. Sci.*, vol. 516, pp. 505–514, 2020.

[22] R. El Bansarkhani and J. Buchmann, "Towards lattice based aggregate signatures," in *Progress in Cryptology – AFRICACRYPT*. D. Pointcheval, D. Vergnaud, Eds., Cham, Cham, Switzerland: Springer, pp. 336–355, 2014.
[23] Z. Yanhua, H. Yupu, J. Mingming, and X. Lili, "Lattice-based sequential aggregate signatures with lazy verification," *J. China Univ. Posts Telecommun.*, vol. 22, no. 6, pp. 36–44, 2015.
[24] M. O. Ozmen, R. Behnia, and A. A. Yavuz, "Fast authentication from aggregate signatures with improved security," in *Financial Cryptography and Data Security*. I. Goldberg, T. Moore, Eds. Berlin, Germany: Springer, pp. 686–705, Feb. 2019.
[25] C. Li, Y. Tian, X. Chen, and J. Li, "An efficient anti-quantum lattice-based blind signature for blockchain-enabled systems," *Inf. Sci.*, vol. 546, pp. 253–264, 2021.
[26] Y. Zhao, "Practical aggregate signature from general elliptic curves, and applications to blockchain," in *Proc. ACM Asia Conf. Comput. Commun. Secur. Assoc. Comput. Machinery*, 2019, pp. 529–538.
[27] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. Adv. Cryptol.*, 1984, pp. 47–53.
[28] X.-J. Lin, L. Sun, and H. Qu, "Generic construction of public key encryption, identity-based encryption and signcryption with equality test," *Inf. Sci.*, vol. 453, pp. 111–126, 2018.
[29] Z. Zhao, G. Wu, W. Susilo, F. Guo, B. Wang, and Y. Hu, "Accountable identity-based encryption with distributed private key generators," *Inf. Sci.*, vol. 505, pp. 352–366, 2019.
[30] Z. Zhao, F. Guo, J. Lai, W. Susilo, B. Wang, and Y. Hu, "Accountable authority identity-based broadcast encryption with constant-size private keys and ciphertexts," *Theor. Comput. Sci.*, vol. 809, pp. 73–87, 2020.
[31] Y. Zhou, B. Yang, Z. Xia, M. Zhang, and Y. Mu, "Identity-based encryption with leakage-amplified chosen-ciphertext attacks security," *Theor. Comput. Sci.*, vol. 809, pp. 277–295, 2020.
[32] X. Zhang, Y. Tang, H. Wang, C. Xu, Y. Miao, and H. Cheng, "Lattice-based proxy-oriented identity-based encryption with keyword search for cloud storage," *Inf. Sci.*, vol. 494, pp. 193–207, 2019.
[33] S. Katsumata, T. Matsuda, and A. Takayasu, "Lattice-based revocable (hierarchical) IBE with decryption key exposure resistance," *Theor. Comput. Sci.*, vol. 809, pp. 103–136, 2020.
[34] M. Ma, D. He, S. Fan, and D. Feng, "Certificateless searchable public key encryption scheme secure against keyword guessing attacks for smart healthcare," *J. Inf. Secur. Appl.*, vol. 50, 2020, Art. no. 102429.
[35] M. Li, "Leveled certificateless fully homomorphic encryption schemes from learning with errors," *IEEE Access*, vol. 8, pp. 26749–26763, 2020.
[36] X. Yang, G. Chen, M. Wang, and X. Pei, "Lightweight searchable encryption scheme based on certificateless cryptosystem," in *Proc. Int. Conf. Mech., Control Comput. Eng*, 2019, pp. 669–6693.
[37] Y. Lu, J. Li, and Y. Zhang, "Privacy-preserving and pairing-free multi-recipient certificateless encryption with keyword search for cloud-assisted IIoTs," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 2553–2562, Sep. 2019.
[38] M. Gupta, *Blockchain for Dummies-IBM Limited Edition*. Hoboken, NJ, USA: Wiley, USA, 2017.
[39] C. Li, X. Chen, Y. Chen, Y. Hou, and J. Li, "A new lattice-based signature scheme in post-quantum blockchain network," *IEEE Access*, vol. 7, pp. 2026–2033, 2019.
[40] D. Rajan and M. Visser, "Quantum blockchain using entanglement in time," *Quantum Rep.*, vol. 1, no. 1, pp. 3–11, 2019.
[41] E. O. Kiktenko *et al.*, "Quantum-secured blockchain," *Quantum Sci. Technol.*, vol. 3, no. 3, pp. 1–7, 2018.
[42] D. Micciancio and O. Regev, "Lattice-based Cryptography," Accessed: Jan. 01, 2021. [Online]. Available: https://cims.nyu.edu/ regev/papers/pqc.pdf
[43] S. Bhasin, J.-P. D'Anvers, D. Heinz, T. Pöppelmann, and M. Van Beirendonck, "Attacking and defending masked polynomial comparison for Lattice-based cryptography," Accessed: Jan. 01, 2021. [Online]. Available: https://eprint.iacr.org/2021/104.pdf
[44] P. Bert and A. Roux-Langlois, "From identification using rejection sampling to signatures via the Fiat-Shamir transform: Application to the BLISS signature," in *Proc. Adv. Inf. Comput. Secur.*, 2018, pp. 297–312.
[45] Y. Zhao, "Aggregation of gamma-signatures and applications to bitcoin," IACR, Leon, France, Tech. Rep. 2018/414, 2018.
[46] X. Lu, W. Yin, Q. Wen, Z. Jin, and W. Li, "A lattice-based unordered aggregate signature scheme based on the intersection method," *IEEE Access*, vol. 6, pp. 33986–33994, 2018.
[47] D. J. Bernstein, "Comparing proofs of security for lattice-based encryption," IACR, Leon, France, Tech. Rep. 2019/691, 2019.
[48] X. Zhang, C. Xu, C. Jin, and R. Xie, "Efficient forward secure identity-based shorter signature from lattice," *Comput. Elect. Eng.*, vol. 40, no. 6, pp. 1963–1971, 2014.
[49] Ethereum Developers Resources, Accessed: Apr. 03, 2020. [Online]. Available: https://ethereum.org/en/developers/
[50] Homebrew Instrictions, Accessed: Apr. 03, 2020. [Online]. Available: https://docs.brew.sh/Homebrew-on-Linux
[51] Nodejs/NPM Installation, Accessed: Apr. 08, 2020. [Online]. Available: https://linuxize.com/post/how-to-install-node-js-on-ubuntu-18.04/
[52] Remix-IDE Help, Accessed: Sep. 09, 2020. [Online]. Available: https://github.com/ethereum/remix-ide
[53] Lattice Cryptography Library, Accessed: Apr. 10, 2020. [Online]. Available: https://www.microsoft.com/en-us/download/details.aspx?id=52371