IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 8 September 2023, accepted 9 November 2023, date of publication 14 November 2023, date of current version 20 November 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3333032

RESEARCH ARTICLE

Multi-Party Audit and Regulatory Mechanism for P2P Electricity Transaction Based on Distributed Traceable Linkable Group Signature

ZHIHU LI[1], BING ZHAO[1], HONGXIA GUO[2], FENG ZHAI[1], AND LIN LI[2]
[1]Department of Metrology, China Electric Power Research Institute, Beijing 100192, China
[2]Marketing Service Center (Metrology Center), State Grid Shandong Electric Power Company, Jinan 250001, China

Corresponding author: Zhihu Li (lizhihu@epri.sgcc.com.cn)

This work was supported in part by the Research Program of State Grid Corporation of China under Grant 5700-202255294A-2-0-QZ.

ABSTRACT Peer-to-Peer (P2P) electricity transaction is an emerging power trading model, which can effectively solve the drawbacks of traditional power trading that requires the exist of intermediaries. As the application spreads, P2P electricity transaction faces some regulatory issues. At present, some existing regulatory models are based on centralized trusteeship centers. There is no distributed regulatory mechanism, and no dynamic addition and deletion mechanism to prevent malicious collusion of regulators. In this article, we propose a multi-party audit and regulatory mechanism for P2P electricity transaction, which is based on a dynamic distributed traceable linkable group signature (DDT-LGS) scheme. This solution realizes the distributed trace and audit for P2P electricity transaction, and further realizes the dynamic permission of regulators to join and exit. We give the system architecture of multi-party audit and regulatory mechanism, formally define the algorithm process, and give a specific DDT-LGS construction. Security and performance analysis shows that the proposed scheme holds strong security, more comprehensive functions and better performance, that is suitable for P2P power trading scenarios.

INDEX TERMS P2P electricity transaction, linkable group signature, multi-party regulatory, distributed key generation.

I. INTRODUCTION

The traditional power trading system is a centralized market managed and controlled by central institutions or power companies, where suppliers (generators or power companies) provide electricity to consumers through the power grid [1]. Transactions typically occur through centralized trading platforms and involve long-term contracts and periodic pricing [2]. However, the traditional system has several disadvantages. It suffers from information asymmetry, high transaction costs, and limited flexibility, which hampers market competition and inhibits the integration of renewable energy sources [3]. Additionally, the centralized structure limits transparency and consumer choice, leading to inefficiencies and reduced innovation [4]. To address these issues and drive energy transition and sustainability, there is a growing exploration of more flexible, transparent, and decentralized power trading models, such as Peer-to-Peer (P2P) power trading [5], [6].

P2P electricity trading is a decentralized model of electricity exchange that enables direct transactions between individual producers and consumers. In P2P electricity trading, participants can buy and sell electricity from and to each other, bypassing the traditional intermediaries such as utilities or electricity retailers. The concept of P2P electricity trading offers several advantages. Firstly, it provides increased market flexibility and autonomy, allowing consumers to choose their electricity sources based on their preferences and needs. Secondly, P2P electricity trading promotes distributed energy access by encouraging small-scale and decentralized energy producers, such as rooftop solar panel owners, to participate in the market, thereby driving the utilization and development of renewable energy resources. Additionally, P2P electricity trading creates

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

a more transparent and fair trading environment, where consumers have real-time access to electricity pricing and supply-demand information, enabling them to make informed purchasing decisions. Lastly, P2P electricity trading enhances market efficiency by reducing intermediaries and transaction costs, resulting in more competitive prices for participants. In summary, P2P electricity trading, with its decentralized nature, flexibility, transparency, and efficiency, brings innovation and transformation to the energy market, facilitating the development of a more sustainable and intelligent energy system [7], [8]. Blockchain technology is widely used in P2P electricity transactions, providing a secure, transparent and decentralized transaction basis [9]. A smart contract is a self-executing computer program that specifies and enforces transaction conditions and behaviors for P2P electricity transactions [10].

P2P electricity trading faces some regulatory issues [11], [12]. First, regulators need to ensure the compliance and fairness of P2P electricity trading platforms. This involves putting in place appropriate rules and standards to ensure the safety, transparency and reliability of transactions and to prevent fraud and misconduct from occurring. Second, regulators need to address issues of data security and privacy protection. P2P electricity trading involves a large amount of energy data and personal information, so it is necessary to formulate corresponding policies and measures to ensure the safe storage, transmission and use of data, while protecting the privacy of participants. In addition, regulators also need to consider the issues of market supervision and market clearing price prediction [13]. P2P electricity transactions involve direct transactions between multiple participants, and regulators need to ensure the compliance of transactions, the accuracy of transaction data, and the ability to effectively handle disputes and disputes. Finally, regulators also need to focus on issues of market competition, fairness and trust evaluation [14]. The development of the P2P electricity trading market may lead to imbalances among participants. Regulators need to formulate measures to promote market competition, prevent monopolistic behavior and unfair competition, and ensure the fairness and sustainable development of the market. Therefore, regulators need to actively follow up and supervise the development of the P2P electricity trading market, and formulate corresponding policies and regulations to protect the rights and interests of participants, maintaining the stability and fairness of the market [15]. This can provide a more reliable and healthy development environment for P2P electricity trading, promoting the transformation and innovation of the energy market.

A. RELATED WORK

P2P electricity trading is a rapidly developing and innovative field, where the market design and trading mechanism are crucial for efficient, fair, and sustainable between electricity consumers and producers. Researchers have started exploring the direct transaction connection between electricity consumers and generators, studying the potential advantages and challenges. Zhang et al. [16] and Wang et al. [17] provided comprehensive reviews of energy applications and P2P energy trading in microgrids, while Abdella et al. [18] analyzed P2P energy trading in smart grids, and Wu et al. [19] proposed multi-agent-based P2P energy trading in microgrids.

With the rise of blockchain technology, P2P electricity trading has gained more attention. Blockchain offers a secure, transparent, and decentralized infrastructure suitable for implementing P2P electricity trading. Guo et al. [20] presented frameworks, applications, and future directions for blockchain-based energy trading, while Khan et al. [21] and Pareek et al. [22] discussed the challenges and applications of blockchain-based P2P energy trading. Li et al. [23] focused on market design and future opportunities for blockchain-based P2P energy trading. At this stage, some successful P2P electricity trading projects have achieved market scale and commercialization [24], [25]. These projects have attracted more participants and investments, involving a wider range of electricity transactions.

During the development of P2P electricity trading, privacy protection and regulation problems have become crucial. Governments and regulatory authorities have started to pay attention to developing frameworks and policies for privacy protection and regulation, aiming to promote healthy market development and address challenges and risks. P2P electricity trading involves sensitive energy data, making data security and privacy protection become highly important. Aitzhan et al. [26] provided a review of data security and privacy in P2P energy trading, Son et al. [27] proposed blockchain-based privacy protection techniques using functional encryption for P2P energy trading, and Ping et al. [28] discussed a privacy-preserving blockchain-based method to optimize energy trading. Baig et al. [29] proposed a blockchain-based P2P energy trading system using open-source angular framework and hypertext transfer protocol, while Schneiders et al. [30] and Soto et al. [31] summarized regulatory challenges and opportunities in P2P energy trading. Gbadega et al. [32] proposed consumer-centric regulatory designs for P2P energy trading. Li et al. [33] conducted research on privacy protection techniques for regulated blockchain transactions in the power industry using cryptographic algorithms such as Pedersen homomorphic commitments and Diffie-Hellman protocol. Gangjun et al. [34] proposed a unified regulatory and shared trading model for power data based on blockchain using a Merkle tree structure.

B. OUR CONTRIBUTION

At present, it has already appeared some regulatory models for P2P electricity trading systems, but most of them are based on centralized trust centers. Even the subsequent distributed architectures using blockchain also only utilize it for data storage, without truly distributed defining regulatory

managers to supervise the identities of electricity trading users, not even further addressing the issues of pre-audit and dynamic addition/deletion of regulatory roles. Designing effective regulatory solutions is an urgent issue for P2P electricity trading.

To address these concerns, we propose a P2P electricity trading multi-party audit and regulatory mechanism based on a dynamic distributed traceable linkable group signature (DDT-LGS) scheme. To the best of our knowledge, it is the first time to employ a dynamic distributed linkable group signature scheme to ensure the audit and regulatory function of electricity trading as well as the dynamic addition/deletion of regulatory managers. The key features of our proposed solution are as follows:

- It achieves distributed regulation for P2P electricity trading, allowing multiple regulatory managers to track the identities of illegal trading users.
- It enables linkable audit for P2P electricity trading, enabling audit managers to judge whether transactions originate from the same illegal user.
- It incorporates dynamic characteristics, allowing for the dynamic addition and deletion of regulatory managers. Moreover, when a regulatory manager joins or exits, the overall public-private key pair of the regulatory group remains unchanged.

The rest of paper is organized as follows: Section II establishes the basic cryptographic protocols required for the construction of the scheme. Section III introduces the system architecture of multi-party audit and regulatory mechanism for P2P electricity transaction and the formal definition of dynamic distributed traceable linkable group signature scheme. Section IV detailly introduces the specific construction of the scheme, and further analyzes its correctness, security and performance. Conclusions are drawn in Section V.

II. PRELIMINARIES

This section gives the basic building blocks to construct multi-party audit and regulatory mechanism, including digital certificate, group signature and distributed key generation.

A. DIGITAL CERTIFICATE

Digital Certificate is a secure tool used for verifying and proving identities in network communications. They are generated based on the Public Key Infrastructure (PKI) to ensure the confidentiality, integrity, and authenticity of communications. The concept of digital certificate was first proposed by L. Kohnfelder in 1978 [35].

The main components of PKI include Certificate Authoritie (CA), Registration Authoritie (RA), Certificate Repository, and Certificate Revocation List (CRL). Digital Certificates are signed by a CA. They typically contain the public key of the certificate holder, certificate holder information, CA information, and the certificate's validity period.

The algorithmic process of a digital certificate is as follows:

1) Certificate Application: The certificate holder submits an application to the CA, providing necessary identity verification and public key information.
2) Identity Verification: The CA verifies the identity of the certificate holder to ensure their legitimacy and authenticity.
3) Certificate Issuance: The CA uses its private key to digitally sign the certificate holder's public key and related information, generating the digital certificate.
4) Certificate Verification: In network communications, other parties can use the CA's public key to verify the signature of the digital certificate, ensuring its authenticity and integrity.

Through digital certificates, communication participants can verify each other's identities, ensuring the security and trustworthiness of communications. Digital certificates are widely used in encryption communications, e-commerce, and identity authentication, providing a crucial security foundation for network communications.

B. GROUP SIGNATURE

Group signature scheme is a cryptographic mechanism aimed at achieving the anonymity of group members, verifiability of signatures, and traceability of group members. They allow members of a group to use an anonymous signature to represent the entire group in signature operations without revealing the identities of individual members. Additionally, each group is managed by a trusted group manager who has the ability to open the group signature and reveal the identity of the signer in case of disputes. The concept of group signatures was initially proposed by D. Chaum and E. Van. Heyst in 1991 [36].

The algorithmic process of a group signature scheme is as follows:

1) Key Generation: In a group signature scheme, the first step is to generate the necessary keys. The group manager is responsible for generating and distributing the key pairs required for the group signature.
2) Group Member Registration: Before joining the group, members go through a registration process. During registration, the group manager verifies the identity of the member and assigns the corresponding individual private key.
3) Group Signature Generation: When group members need to perform signature operations, they can use the group signature key and their individual private key to generate an anonymous signature.
4) Signature Verification: Anyone can verify the validity of the signature using public parameters and the group signature without needing to know the specific identities of the individual members.
5) Signature Opening: The group manager uses its private key to decrypt the group signature and reveal the identity of the signer.

Group signature schemes typically possess security features such as anonymity, traceability, unforgeability, and

resistance to collusion attacks. They provide a solution that balances anonymity and traceability, preserving individual privacy, ensuring the trustworthiness of signatures, and managing the identities of group signers.

C. DISTRIBUTED KEY GENERATION

Distributed Key Generation (DKG) is a cryptographic technique used to generate key pairs in a distributed environment. In traditional key generation methods, a single entity generates keys and distributes them to the parties involved. However, in a distributed environment, where parties may not trust each other or potential attackers may exist, DKG allows multiple participants to collaborate in generating key pairs, ensuring the security and trustworthiness of the keys. Feldman-DKG [37] and Pedersen-DKG [38] are two common DKG schemes.

The algorithm steps for a distributed key generation scheme are as follows:

1) Setup: Determine the number of participants, security parameters, and key lengths.
2) Key Generation Initialization: Each participant generates their own partial private and public keys.
3) Key Reconstruction: Each participant uses their partial private key to reconstruct the complete private key. Multiple-party computation protocols (such as Lagrange interpolation) are used to combine the partial private keys from all participants.
4) Key Verification: The generated key pair needs to be verified to ensure its validity and security.

The goal of distributed key generation is to ensure the generation of secure and trustworthy key pairs in a distributed environment to support secure communication and data protection. It has important applications in various fields such as blockchain, secure multi-party computation, and cryptographic protocols. By decentralizing the key generation process and utilizing cryptographic protocols for validation and collaboration, distributed key generation ensures the security and reliability of the keys.

D. LAGRANGIAN INTERPOLATION ALGORITHM

For a polynomial function $q(x)$ of degree $d-1$, given d point values $(1, q_1), (2, q_2), \ldots, (d, q_d)$, where q_i corresponds to the value of the function, $i \in \{1, \ldots, d\}$.

By using the lagrangian interpolation algorithm, the polynomial $q(x)$ can be obtained

$$q(x) = \sum_{i=1}^{d} q_i \Delta_{i,D}(x)$$

where the lagrange coefficient of $q(i)$ is $\Delta_{i,D}(x) = \prod_{j \in D, j \neq i} \frac{x-j}{i-j}$, and D is a set containing d elements.

III. MULTI-PARTY AUDIT AND REGULATORY MECHANISM FOR P2P ELECTRICITY TRANSACTION

The multi-party audit and regulatory framework of P2P electricity trading is mainly composed of four entities: P2P electricity trading platform, power purchaser, power seller and management authorities. The system architecture is depicted in FIGURE 1.

- P2P power trading platform: Located at the P2P electricity transaction layer, connecting power buyers and power sellers. The platform provides transaction matching, data management, transaction clearing and other functions to ensure market compliance and fair competition.
- Power purchaser: A user located at the P2P electricity transaction layer, who is the power purchaser of the P2P power transaction. They can directly choose to buy electricity from specific generators and participate in trading activities.
- Power seller: A user located at the P2P electricity transaction layer, who is the power seller of the P2P power transaction. They can publish their electricity supply information on the platform and conduct transactions with buyers.
- Management Authorities: Located at the dynamic multi-party audit and regulatory layer, responsible for auditing and supervising the entire P2P electricity trading market. They include registration management, audit management, and regulatory management, which are used to formulate trading system access control strategies, handle transaction dispute complaints, track illegal users, and resolve disputed transactions.

Compared with the P2P electricity transaction with traditional centralized supervision, the system architecture has increased the role of distributed regulators, and has multiple functions of registration, audit and regulatory. It can be noticed that registration administrators, audit administrators, and regulatory administrators can all be distributed. Here, we only discuss the case of distributed regulatory administrators, because the implement method is the same. At the same time, the system architecture can enable administrators to join and exit dynamically to resist collusion attacks. The algorithm process is depicted in FIGURE 2.

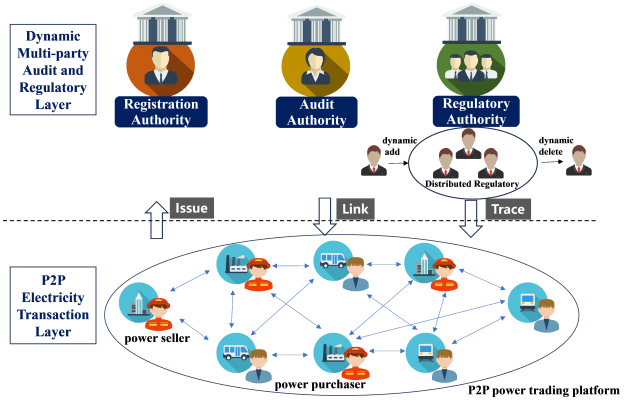

FIGURE 1. System architecture of multi-party audit and regulatory for P2P electricity transaction.

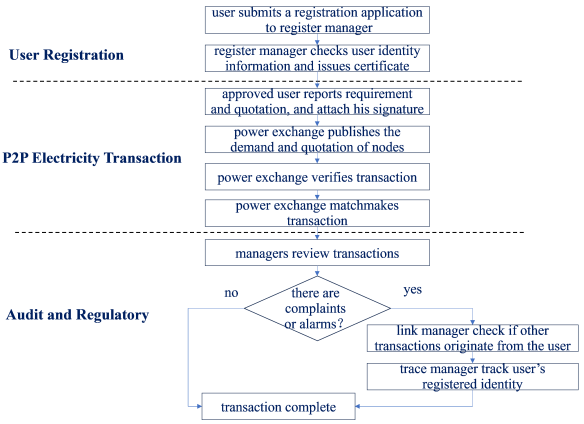

FIGURE 2. Algorithm process of multi-party audit and regulatory for P2P electricity transaction.

The multi-party audit and supervision function of P2P electricity transaction system is realized based on the cryptographic scheme of dynamic distributed traceable linkable group signature, which is a new scheme proposed for the first time. This scheme is based on group signature. But compared with the traditional group signature, it adds link, distributed trace, dynamic add and delete algorithms. These steps are also the key to multi-party audit and regulatory.

A dynamic distributed traceable linkable group signature (DDT-LGS) scheme includes the following algorithms:

- *Setup* : System generates the public parameters.
- *RKGen* : Registration authority manager generates the register key.
- *LKGen*: Audit authority manager generates the link key.
- *TKGen*: Regulatory authority manager generates the trace key.
- *Issue*: Power trading user interacts with the registration manager to generate transaction certificate.
- *GSign*: Power trading user conducts transaction and generates signature.
- *GVer*: The P2P power trading platform verifies user's signature.
- *Link*: Audit authority manager judges any two transactions to determine whether the transactions come from the same user.
- *Trace*: Regulatory authority manager track the identity of user for specific transaction.
- *TMAdd*: Dynamic join of regulatory manager.
- *TMDele*: Dynamic exit of regulatory manager.

IV. DYNAMIC DISTRIBUTED TRACEABLE LINKABLE GROUP SIGNATURE

In this section, we construct a concrete dynamic distributed traceable linkable group signature scheme, and give the correctness, security and performance analysis of this construction.

A. CONCRETE SCHEME

According to the given system model, we present the DDT-LGS scheme construction in this part. Specifically, TABLE 1 and TABLE 2 first present some frequently used symbols and notations in the scheme. Then, it is the specific construction.

TABLE 1. Symbol description.

Symbol	Descriptions
(rpk, rsk)	register manager RM's key pair
(lpk, lsk)	link manager LM's key pair
(tpk_i, tsk_i)	distributed trace manager TM_i's key pair
tpk	public key of trace manager group
U_{uid}	user U's identity
usk	user U's private key
$Cert_u$	user U's certificate
ST_u	identity and certificate status of the registered user
TM_r	added trace manager
TM_v	deleted trace manager

TABLE 2. Notation description.

Notation	Descriptions
G_1, G_2, G_T	cyclic groups
e	a type 3 bilinear map
H	a hash function
$QR(N)$	quadratic remainder of N
$PKDL$	a zero-knowledge proof of discrete logarithm
m	message
σ	signature
(n, t)	threshold for t from n

***Setup*(1^λ)**: Define G_1, G_2 and G_T be cyclic groups with the same large prime p. Let $e : G_1 \times G_2 \to G_T$ is the type 3 pairing, where $G_1 \neq G_2$, and there is no valid homomorphic mapping between G_1 and G_2. g_1 and g_2 are the generators of group G_1 and group G_2 respectively. Let H is a hash function $H : \{0, 1\}^* \to Z_p^*$. Randomly chooses $g, h \leftarrow G_1$.

Finally, outputs the system public parameter $params = (p, G_1, G_2, G_T, e, g_1, g_2, g, h)$.

***RKGen*$(params)$**: Input the system public parameters $params$, the register manager RM generates his public-private key pair. RM randomly chooses $\gamma \leftarrow Z_p^*$ and calculates $\omega = g_2^\gamma$. Then the public key of the register manager is $rpk = \omega$, and the private key is $rsk = \gamma$. At the same time, RM initializes the identity and certificate status of the registered user by setting $ST_u = \emptyset$.

***LKGen*$(params)$**: Input the system public parameters $params$, the link manager LM generates his public-private key pair. LM generates a RIPE composite number N, $N = PQ, P = 2p' + 1, Q = 2q' + 1$. Then chooses a subgroup $G_3 =< g_3 >$ from $\mathbb{Z}_N^*$, $(g_3|N) = 1$ (i.e. $G_3 \subset QR(N)$), and the order of group G_3 is $p'q'$. Chooses a subgroup G_4 of G_3, makes the order of group G_4 is p'. Chooses a random element $h_3 \in G_4$, then the order of h_3 is p'(i.e. ${h_3}^{p'} = 1$), the order of g_3 is $p'q'$. Then the public key of the link manager is $lpk = (N, g_3, h_3, G_3, G_4)$, and the private key is $lsk = p'$.

***TKGen*$(params)$**: Input the system public parameters $params$, the distributed trace managers

$\{TM_1, TM_2, \cdots, TM_n\}$ generates their own public-private key pair. The specific operation is as follows:

1) TM_i randomly chooses $a_{i,1}, a_{i,2}, \cdots, a_{i,n} \leftarrow Z_p^*$, generates the polynomial $f_i(x) = a_{i,0} + a_{i,1}x + \cdots + a_{i,t-1}x^{t-1} \mod p$, computes commitment $C_{i,k} = g^{a_{i,k}}$, where $k = 0, 1, 2, \cdots, t-1$.
2) TM_i computes sub-secret share $s_{i,j} = f_i(j)$, where $j \in \{1, \cdots, n\}$, then sends $s_{i,j}$ to other trace manager TM_j.
3) TM_j receives the set $\{s_{1,j}, \cdots, s_{n,j}\}$, then he verifies the equation

$$g^{s_{i,j}} = \prod_{k=0}^{t-1} C_{i,k}^{j^k}, \quad i \in \{1, \cdots, n\}$$

If there exists $s_{i,j}$ that does not satisfy the above equation, TM_j declares the result invalid and terminates the protocol.
4) TM_j computes his share $d_j = \sum_{i=1}^n s_{i,j} = \sum_{i=1}^n f_i(j) \mod p$ by defining $f(x) = f_1(x) + f_2(x) + \ldots + f_n(x)$, thus d_j is a share of $f(0)$. TM_j computes $S_j = g^{d_j}$. Then the public key of the trace manager TM_j makes is $tpk_j = S_j$, and the private key is $tsk_j = d_j$.
5) Here, among the distributed trace managers $\{TM_1, TM_2, \cdots, TM_n\}$, More than t participants can collaboratively reconstruct the shared public key through lagrange interpolation

$$\prod_{i=1}^{n} C_{i,0} = g^{\sum_{i=1}^{t} d_i \prod_{j \in D, j \neq i} \frac{j}{j-i}} = \prod_{i \in D} S_i^{\Delta_{i,D}(0)}$$

where $\Delta_{i,D}(x) = \prod_{j \in D, j \neq i} \frac{x-j}{i-j}$.

Finally, the distributed trace managers $\{TM_1, TM_2, \cdots, TM_n\}$ computes $S = \prod_{i=1}^n C_{i,0}$, and set the communal trace public key $tpk = S$.

***Issue*[*RM*, *U*](*params*, *rpk*, U_{uid})**: Input the system public parameters *params*, user U interacts with the register manager RM, produces a certificate $Cert_u$ to complete the registration. The specific operation is as follows:

1) U randomly chooses $y \leftarrow Z_p^*$, computes $Y = h_1^y$, and a zero-knowledge proof of y,

$$\pi_Y = PKDL\{y | Y = h_1^y\}$$

Then, U sends his identity U_{uid} along with the generated data Y, π_Y to the register manager RM.
2) RM verifies the identity of U_{uid}. If $U_{uid} \in ST_u$, or $Y \in ST_u$, or π_Y is invalid, then aborts. Otherwise, RM chooses $x \leftarrow Z_p^*$, computes $A = (g_1 Y)^{\frac{1}{\gamma+x}}$, then RM send $Cert_u = (A, x)$ to user U. Simultaneously, RM stores $(U_{uid}, Cert_u, Y, \pi_Y)$ in ST_u, and updates ST_u.
3) U verifies $e(A, g_2)^x e(A, \omega) = e(g_1 Y, g_2)$. If the equality fails, then aborts. Otherwise, U generates his certificate and private key as $Cert_u = (A, x)$ and $usk = y$.

***GSign*(*params*, *m*, *Cert_u*, *rpk*, *lpk*, *tpk*, *usk*)**: Input the system public parameters *params*, user U performs linkable group signature on message $m \in \{0, 1\}^*$ with private key usk and certificate $Cert_u$. The specific operation is as follows:

1) U randomly chooses $\alpha \leftarrow Z_p^*$, computes $T_1 = g^\alpha, T_2 = S^\alpha h^y, T_3 = A \cdot S^\alpha, T_4 = g_3^y h_3^\alpha$, then computes $\delta_1 = x\alpha, \delta_2 = xy$.
2) U randomly chooses $r_\alpha, r_x, r_y, r_{\delta_1}, r_{\delta_2} \leftarrow Z_p^*$, and completes the zero-knowledge proof of parameter $(\alpha, x, y, \delta_1, \delta_2)$ through the following operations.
$R_1 = g^{r_\alpha}, R_2 = S^{r_\alpha} h^{r_y}, R_3 = e(T_3, g_2)^{r_x} \cdot e(S, \omega)^{-r_\alpha} \cdot e(S, g_2)^{-r_{\delta_1}} \cdot e(h, g_2)^{-r_y}, R_4 = g_3^{r_y} h_3^{r_\alpha}, R_5 = T_1^{r_x} g^{-r_{\delta_1}}, R_6 = T_2^{r_x} \cdot S^{-r_{\delta_1}} \cdot h^{-r_{\delta_2}}, R_7 = T_4^{r_x} g_3^{-r_{\delta_2}} h_3^{-r_{\delta_1}}$
3) U computes $c = H(m \parallel T_1 \parallel T_2 \parallel T_3 \parallel T_4 \parallel R_1 \parallel R_2 \parallel R_3 \parallel R_4 \parallel R_5 \parallel R_6 \parallel R_7)$.
4) U computes $s_\alpha = r_\alpha + c\alpha, s_x = r_x + cx, s_y = r_y + cy, s_{\delta_1} = r_{\delta_1} + c\delta_1, s_{\delta_2} = r_{\delta_2} + c\delta_2$.

Finally, the signature is $\sigma = (T_1, T_2, T_3, T_4, c, s_\alpha, s_x, s_y, s_{\delta_1}, s_{\delta_2})$.

***GVer*(*params*, *m*, *σ*, *rpk*, *lpk*, *tpk*)**: Input the system public parameters *params*, message m and signature σ, the group signature verification operation is as follows:

1) Verify user V computes $\tilde{R_1}, \tilde{R_2}, \tilde{R_3}, \tilde{R_4}, \tilde{R_5}, \tilde{R_6}, \tilde{R_7}$, where $\tilde{R_1} = g^{s_\alpha} \cdot T_1^{-c}, \tilde{R_2} = S^{s_\alpha} h^{s_y} \cdot T_2^{-c}, \tilde{R_3} = e(T_3, g_2)^{s_x} \cdot e(S, \omega)^{-s_\alpha} \cdot e(S, g_2)^{-s_{\delta_1}} \cdot e(h, g_2)^{-s_y} \cdot (\frac{e(\tilde{T_3}, \omega)}{e(g_1, g_2)})^c, \tilde{R_4} = g_3^{s_y} h_3^{s_\alpha} \cdot T_4^{-c}, \tilde{R_5} = T_1^{s_x} g^{-s_{\delta_1}}, \tilde{R_6} = T_2^{s_x} \cdot S^{-s_{\delta_1}} \cdot h^{-s_{\delta_2}}, \tilde{R_7} = T_4^{s_x} g_3^{-s_{\delta_2}} h_3^{-s_{\delta_1}}$
2) Verify user V verifies $c \stackrel{?}{=} H(m \parallel T_1 \parallel T_2 \parallel T_3 \parallel T_4 \parallel \tilde{R_1} \parallel \tilde{R_2} \parallel \tilde{R_3} \parallel \tilde{R_4} \parallel \tilde{R_5} \parallel \tilde{R_6} \parallel \tilde{R_7})$.

If the cheak succeeds, accepts the group signatire and output 1, otherwise output 0.

***Link*(*params*, *lsk*, (*m*, *σ*), (*m′*, *σ′*))**: Input the system public parameters *params*, message-signature pairs (m, σ) and (m', σ'), the link manager LM uses his private key lsk to perform the link operation.

1) LM first verifies the validity of given message-signature pairs $(m, \sigma), (m', \sigma')$ by above $GVer$ algorithm. If any signature is invalid, it aborts.
2) Otherwise, for the component T_4 in signature σ and T_4' in signature σ', LM judges $(\frac{T_4}{T_4'})^{p'} \stackrel{?}{=} 1$. If the cheak succeeds, it implies the two message-signature pairs are from the same signing user, outputs 1 when this case occurs. otherwise outputs 0.

***Trace*(*params*, *tsk_i*, (*n*, *t*), (*m*, *σ*))**: Input the system public parameters *params*, threshold (n, t), signature pair (m, σ), the distributed trace managers $\{TM_1, TM_2, \cdots, TM_n\}$ uses their private key $tsk_i, i \in \{1, 2, \cdots, n\}$ to perform the trace operation.

1) trace manager $TM_i, i \in \{1, 2, \cdots, n\}$ first verifie the validity of given message-signature pair (m, σ) by above $GVer$ algorithm. If any signature is invalid, it aborts.
2) Otherwise, TM_i computes $\hat{T}_i = T_1{}^{d_i}$ using his trace private key, and also makes a non-interactive proof $PKDL$ to prove that he does own the trace private key d_i, where $\pi_{\hat{T}_i} = PKDL\{d_i | \hat{T}_i = T_1{}^{d_i} \wedge S_i = g^{d_i}\}$, and

generate his tracing share $share_i = (i, \hat{T}_i, \pi_{\hat{T}_i})$. Then, trace manager $\{TM_1, TM_2, \cdots, TM_n\}$ generate their respective tracking shares $share_1, share_2, \cdots, share_n$.

3) According to the threshold value t, select the set $D \subset \{1, 2, \cdots, n\}$, $| D | = t$. For any $i \in D$, if $share_i \neq \emptyset$ and $\pi_{\hat{T}_i}$ is effective, then computes

$$\tilde{Y} = T_2 / \prod_{i \in D} (\hat{T}_i)^{\Delta_{i,D}(0)}, \tilde{A} = T_3 / \prod_{i \in D} (\hat{T}_i)^{\Delta_{i,D}(0)}$$

4) According to the registration state list ST_u given by register manager RM, if exists Y and $Cert_u$ in the state list ST_u such that $Y = \tilde{Y} \wedge A = \tilde{A}$, returns the corresponding user information U_{uid}.

TMAdd $(params, TM\_r, \{tsk\_1, tsk\_2, \cdots, tsk\_n\})$: Input the system public parameters $params$, the original distributed trace managers $TM_1, TM_2, \cdots, TM_n$ uses their private key $tsk_i, i \in \{1, 2, \cdots, n\}$ to perform the manager addition operation for new trace manager TM_r. They interactively generate temporary private key d_r for TM_r while keeping the public key S unchanged. The specific operation is as follows:

1) Primite trace manager $TM_i, i = 1, \cdots, n$ randomly chooses $b_{i,0}, b_{i,1}, b_{i,2}, \cdots, b_{i,t-1} \leftarrow \mathbb{Z}_q^*$, generates a polynomial $g_i(x)$ of degree $t-1$ which satisfies $g_i(r) = 0$.

$$g_i(x) = b_{i,0} + b_{i,1}x + b_{i,2}x^2 + \cdots + b_{i,t-1}x^{t-1} \bmod p$$
$$g_i(r) = 0$$

Trace manager TM_i computes the corresponding commit $C'_{i,k}$ and broadcasts it.

$$C'_{i,k} = g^{b_{i,k}}, \quad k = 0, 1, \cdots, t-1$$

At the same time, TM_i computes sub-secret share

$$s'_{i,j} = g_i(j)$$

for other $n-1$ members $TM_j, j = 1, \cdots, n, j \neq i$, and sends it to TM_j.

2) After received the sub-secret share $s'_{i,j}$ from TM_i, trace manager TM_j verifies whether the sub-secret share is valid by the following equation.

$$g^{s'_{i,j}} = \prod_{k=0}^{t-1} C'^{j^k}_{i,k}, \quad i \in \{1, \cdots, n\}$$

3) After verifying the sub-secret shares generated by all other trace managers, trace manager TM_j uses the following equation to generate its temporary private key.

$$d'_j = d_j + \sum_{i=1}^{n} g_i(j)$$

And sends this temporary private key d'_j to new trace manager TM_r.

4) After received the temporary private key d'_j from at least t members $TM_i, i = 1, \cdots, t$, new trace manager TM_r can calculate its own temporary private key d_r by using Lagrange interpolation algorithm.

$$d_r = \sum_{i=1}^{t} d'_i \prod_{j=1, j \neq i}^{t} \frac{j-r}{j-i}$$

Then, the new trace manager TM_r obtains its own temporary private key d_r without knowing the real permanent private key of other managers, and the public signature verify key S not been changed.

TMDele $(params, TM\_v, \{tsk\_1, tsk\_2, \cdots, tsk\_n\})$: Input the system public parameters $params$, the original distributed trace managers $TM_1, TM_2, \cdots, TM_n$ uses their private key $tsk_i, i \in \{1, 2, \cdots, n\}$ to perform the manager delete operation for deleting manager TM_v. They interactively reconstruct the private key $\tilde{d}_i, i = 1, \cdots, n-1$ while keeping the public key S unchanged. The specific operation is as follows:

1) Primite trace managers $TM_i, i = 1, \cdots, n, i \neq v$ randomly chooses $c_{i,1}, c_{i,2}, \cdots, c_{i,t-1} \leftarrow \mathbb{Z}_q^*$, generates a polynomial $h_i(x)$ of degree $t-1$.

$$h_i(x) = c_{i,1}x + c_{i,2}x^2 + \cdots + c_{i,t-1}x^{t-1} \bmod p$$

Trace manager TM_i computes the corresponding commit $\tilde{C}_{i,k}$ and broadcasts it.

$$\tilde{C}_{i,k} = g^{c_{i,k}}, \quad k = 1, 2, \cdots, t-1$$

At the same time, TM_i computes sub-secret share

$$\tilde{s}_{i,j} = h_i(j)$$

for other $n-2$ members $TM_j, j = 1, \cdots, n, j \neq i, v$, and sends it to TM_j.

2) After received the sub-secret share $\tilde{s}_{i,j}$ from TM_i, trace manager TM_j verifies whether the sub-secret share is valid by the following equation.

$$g^{\tilde{s}_{i,j}} = \prod_{k=0}^{t-1} \tilde{C}^{j^k}_{i,k}, \quad i \in \{1, \cdots, n\}, i \neq v$$

3) After verifying the sub-secret shares generated by all other trace managers, trace manage TM_j uses the following equation to generate its new private key.

$$\tilde{d}_j = d_j + \sum_{i=1, i \neq v}^{n} h_i(j)$$

Then, the private keys of all other members will be reconstructed and updated, so the deleted node TM_v is naturally invalid, and the public signature verify key S not been changed.

B. CORRECTNESS ANALYSIS

The correctness of this DDT-LGS scheme is justified by the following equation.

– Correctness of group signature verification.

$$\begin{aligned}
\tilde{R_1} &= g^{s_\alpha} \cdot T_1^{-c} = g^{r_\alpha} \cdot g^{c\alpha} \cdot g^{-c\alpha} = R_1 \\
\tilde{R_2} &= S^{s_\alpha} h^{s_y} \cdot T_2^{-c} = S^{r_\alpha} S^{c\alpha} \cdot h^{r_y} h^{cy} \cdot (S^\alpha h^y)^{-c} = R_2 \\
\tilde{R_3} &= e(T_3, g_2)^{s_x} \cdot e(S, \omega)^{-s_\alpha} \cdot e(S, g_2)^{-s_{\delta_1}} \\
&\quad \cdot e(h, g_2)^{-s_y} \cdot (\frac{e(T_3, \omega)}{e(g_1, g_2)})^c \\
&= e(T_3, g_2)^{r_x} e(T_3, g_2)^{cx} \cdot e(S, \omega)^{-r_\alpha} e(S, \omega)^{-c\alpha} \\
&\quad \cdot e(S, g_2)^{-r_{\delta_1}} e(S, g_2)^{-c\delta_1} \cdot e(h, g_2)^{-r_y} e(h, g_2)^{-cy} \\
&\quad \cdot (\frac{e(T_3, \omega)}{e(g_1, g_2)})^c \\
&= R_3 \cdot (e(T_3, g_2)^x \cdot e(S, \omega)^{-\alpha} \cdot e(S, g_2)^{x\alpha} \\
&\quad \cdot e(h, g_2)^{-y})^c \cdot (\frac{e(T_3, \omega)}{e(g_1, g_2)})^c \\
&= R_3 \cdot e(A^x \cdot S^{-\alpha\gamma} \cdot h^{-y}, g_2)^c \cdot (\frac{e(T_3, \omega)}{e(g_1, g_2)})^c \\
&= R_3 \cdot (\frac{e(g_1, g_2)}{e(T_3, \omega)})^c \cdot (\frac{e(T_3, \omega)}{e(g_1, g_2)})^c \\
&= R_3
\end{aligned}$$

The verification principle of remaining $\tilde{R_4} - \tilde{R_7}$ is the same as $\tilde{R_1} - \tilde{R_2}$.

– Linkable correctness.
If for any two message-signature pairs (m, σ) and (m', σ'), they are from the same user's signature, then they are signed by the same private key. Therefore, there is

$$(\frac{T_4}{T_4'})^{p'} = (\frac{h_3^\alpha}{h_3^{\alpha'}})^{p'} = (h_3{}^{p'})^{\alpha-\alpha'} = 1$$

– Traceable correctness.

$$\begin{aligned}
\tilde{Y} &= T_2 / \prod_{i \in D} (\hat{T}_i)^{\Delta_{i,D}(0)} = T_2 / \prod_{i \in D} ((g^{d_i})^{\Delta_{i,D}(0)})^\alpha \\
&= T_2 / S^\alpha = Y \\
\tilde{A} &= T_3 / \prod_{i \in D} (\hat{T}_i)^{\Delta_{i,D}(0)} == T_3 / \prod_{i \in D} ((g^{d_i})^{\Delta_{i,D}(0)})^\alpha \\
&= T_3 / S^\alpha = A
\end{aligned}$$

C. SECURITY ANALYSIS

As a group signature scheme, DDT-LGS satisfies the security of anonymity and traceability. In this section, we analyze its security.

Theorem 1: The proposed DDT-LGS construction satisfies anonymity under the Decisional Diffie-Hellman Problem (DDH) assumption.

proof: Assuming $\mathcal{A}$ is an adversary to attack the new scheme anonymity with an advantage ε, algorithm $\mathcal{B}$ is to solve the DDH problem. Based on the existence of DDH assumption, it's reduced to get the anonymity security of DDT-LGS construction. We first briefly introduce the DDH assumption.

1) DDH ASSUMPTION

Suppose that (p, G_1, g) is defined the same as *Setup* algorithm in DDT-LGS Scheme. The Decision Diffie-Hellman assumption states that, for any $a, b, c \in \mathbb{Z}_p$, given $(g, g^a, g^b, g^c) \in G_1$, it is hard to decide $c = ab$. That is, the probability

$$|\Pr[\mathcal{A}(g, g^a, g^b, g^{ab}) = 1] - \Pr[\mathcal{A}(g, g^a, g^b, g^c) = 1]|$$

is negligible.

The details of attack-simulation experiment is as follows.

– Setup phase. After obtaining $(O_1, O_2, O_3, O_4) = (g, g^a, g^b, g^c)$ as the input of DDH problem, algorithm $\mathcal{B}$ generates public parameters $(p, G_1, G_2, G_T, e, g_1, g_2, g, h)$ to $\mathcal{A}$.
– Query Phase. Adversary $\mathcal{A}$ makes queries of *Issue* oracle. When querying for *Issue* phase, $\mathcal{B}$ computes $Cert_{\mathcal{A}} = (A, x)$, sends $Cert_{\mathcal{A}}$ to $\mathcal{A}$ as certificate.
– Challenge Phase. Aadversary $\mathcal{A}$ selects $(usk_0^*, Cert_0^*)$, $(usk_1^*, Cert_1^*)$, message m^*, and challenges to $\mathcal{B}$. Algorithm $\mathcal{B}$ chooses bit $b \leftarrow \{0, 1\}$, and generates challenge group signature σ^* by using $(usk_b^*, Cert_b^*)$ and m^*. This signature simulates the real *GSign* algorithm, only the difference is that it sets $g = O_1, S = O_2, T_1^* = O_3, T_2^* = O_4 \cdot h^{usk_b^*}, T_3^* = O_4 \cdot A_b^*$, then returns σ^* to $\mathcal{A}$.
– Guess Phase. Finally, $\mathcal{A}$ outputs a bit b' as a guess result to b. If $b' = b$, then $\mathcal{B}$ outputs 1.

From the above query-answer interaction, for adversary $\mathcal{A}$, σ^* is a valid signature. If adversary $\mathcal{A}$ guesses the value of b with non-negligible probability greater than 1/2, then $\mathcal{B}$ could solve the DDH problem with the same probability. We know it's contradict by DDH assumption. Thus, the proposed DDT-LGS construction satisfies anonymity. □

Theorem 2: The proposed DDT-LGS construction satisfies traceability under the q-Strong Diffie-Hellman (q-SDH) assumption.

Proof: Assuming $\mathcal{A}$ is an adversary to attack the new scheme traceability with an advantage ε, algorithm $\mathcal{B}$ is to solve the q-SDH problem. Based on the existence of q-SDH assumption, it's reduced to get the traceability security of DDT-LGS construction. We first briefly introduce the q-SDH assumption.

2) Q-SDH ASSUMPTION

Suppose that (p, G_1, G_2, g_1, g_2) is defined the same as *Setup* algorithm in DDT-LGS Scheme. The q-Strong Diffie-Hellman assumption states that, for any $\gamma \in \mathbb{Z}_p^*$, given a (q+2)-tuple $(g_1, g_1^\gamma, g_1^{\gamma^2}, \ldots, g_1^{\gamma^q}, g_2, g_2^\gamma)$, it is hard to compute $(g_1^{\frac{1}{\gamma+x}}, x)$. That is, the probability

$$|\Pr[\mathcal{A}(g_1, g_1^\gamma, g_1^{\gamma^2}, \ldots, g_1^{\gamma^q}, g_2, g_2^\gamma) = (g_1^{\frac{1}{\gamma+x}}, x)]|$$

is negligible.

The details of attack-simulation experiment is as follows.

– Setup phase. After obtaining (p, G_1, G_2, g_1, g_2) and $(g_1, g_1^\gamma, g_1^{\gamma^2}, \ldots, g_1^{\gamma^q}, g_2, g_2^\gamma)$ as the input of q-SDH

problem, algorithm $\mathcal{B}$ computes $g_1' = g_1 Y$, generates public parameters $(p, G_1, G_2, G_T, e, g_1', g_2, g, h)$ to $\mathcal{A}$.
- Query Phase. Adversary $\mathcal{A}$ makes queries of *Issue* oracle and *GSign* oracle. When querying for *Issue* phase, $\mathcal{B}$ computes $g_1' = g_1 Y$, chooses $x \leftarrow Z_p^*$, computes $A = (g_1')^{\frac{1}{\gamma+x}}$, and sends $Cert_{\mathcal{A}} = (A, x)$ to $\mathcal{A}$ as certificate. When querying for *GSign* phase, $\mathcal{B}$ simulates the real *GSign* algorithm to get signature σ, and returns σ to $\mathcal{A}$.
- Forge Phase. Aadversary $\mathcal{A}$ outputs a forged group signature σ^* on message m^*.

From the above query-answer interaction, for adversary $\mathcal{A}$, if he succeeds in forging the signature with non-negligible probability, it means the certificate is valid, which also means that $\mathcal{B}$ could solve the q-SDH problem with the same probability. We know it's contradict by q-SDH assumption. Thus, the proposed DDT-LGS construction satisfies traceability. □

D. PERFORMANCE ANALYSIS

In this section, we analyze the performance of DDT-LGS scheme, from the perspectives of functional features, the key and signature lengths, and the phase running time, and then compare it with the GS signature proposed by Boneh et al. [39], the LDGS signature proposed by Manulis et al. [40], the DT-GS signature proposed by Lu et al. [41], and the LGS signature proposed by Zheng et al. [42]. The results are shown in TABLE 3, TABLE 4, TABLE 5 and FIGURE 3 respectively.

TABLE 3. Features comparison with related works.

Scheme	Anony	Trace	D-Trace	Link	D-Ad/De
GS	✓	✓	×	×	×
LDGS	✓	✓	×	✓	×
DT-GS	✓	✓	✓	×	×
LGS	✓	✓	×	✓	×
Ours	✓	✓	✓	✓	✓

As shown in TABLE 3, we give the functional features comparison of proposed DDT-LGS with other related works, where D-Trace denotes distributed trace, D-Ad/De denotes dynamic add and delete. It can be concluded from the table that our proposed DDT-LGS scheme possesses the properties of anonymity, traceability, distributed-traceability, linkability, dynamic addition and deletion, which is better than other schemes.

TABLE 4. Performance comparison with related works.

Scheme	$\lvert M-pk\rvert$	$\lvert M-sk\rvert$	$\lvert U-sk\rvert$	$\lvert\sigma\rvert$
GS	$\lvert\mathbb{G}\rvert$	$2\lvert\mathbb{Z}_p\rvert$	$\lvert\mathbb{Z}_p\rvert+\lvert\mathbb{G}\rvert$	$3\lvert\mathbb{G}\rvert+6\lvert\mathbb{Z}_p\rvert$
LDGS	ϕ	ϕ	$\lvert\mathbb{Z}_q\rvert+3\lvert\mathbb{G}\rvert$	$4\lvert\mathbb{Z}_p\rvert+6\lvert\mathbb{G}\rvert$
DT-GS	$(n+1)\lvert\mathbb{G}\rvert$	$(n+1)\lvert\mathbb{Z}_p\rvert$	$\lvert\mathbb{Z}_p\rvert$	$9\lvert\mathbb{G}\rvert$
LGS	$4\lvert\mathbb{G}\rvert+2\lvert\mathbb{Z}_n\rvert$ $+4l_g+k$	$4\lvert\mathbb{Z}_n\rvert$	$2\lvert\mathbb{Z}_n\rvert$	$3\lvert\mathbb{G}\rvert+k$ $+\varepsilon(4l_g+3k)$
Ours	$(n+2)\lvert\mathbb{G}\rvert$	$(n+2)\lvert\mathbb{Z}_p\rvert$	$\lvert\mathbb{Z}_p\rvert$	$4\lvert\mathbb{G}\rvert+6\lvert\mathbb{Z}_p$

As shown in TABLE 4, we give the performance comparison of proposed DDT-LGS with other related works from the perspective of key and signature length. Here, $|M-pk|$,$|M-sk|$,$|U-sk|$,$|\sigma|$ denote the size of managers' public keys, managers' secret keys, user's secret key and group signature. $|\mathbb{G}|$, $|\mathbb{Z}_p|$,$|\mathbb{Z}_q|$,$|\mathbb{Z}_n|$ denote the size of group $\mathbb{G}$, $\mathbb{Z}_p$, $\mathbb{Z}_q$, and $\mathbb{Z}_n$. n denotes the number of distributed trace members. l_g,k,ε denote the size of security parameters. ϕ denotes there is no such parameter variable in the scheme. It can be concluded from the table that our proposed DDT-LGS scheme possesses a slightly shorter user secret key and signature size, but with a slightly longer managers'public key and managers' secret keys size due to the existence of multiple regulatory authorities and distributed managers.

TABLE 5. Runtime with threshold (t,n).

(t,n)	*TKGen* phase runtime	*Trace* phase runtime
(3,5)	0.579	0.030
(5,10)	5.277	0.051
(10,20)	30.828	0.103
(20,40)	179.472	0.211
(30,60)	621.972	0.326

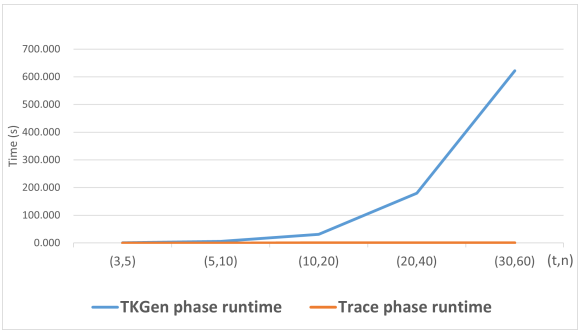

FIGURE 3. *TKGen* phase and *Trace* phase runtime.

As shown in TABLE 5 and FIGURE 3, we give the simulation runtime of proposed DDT-LGS from the perspective of *TKGen* and *Trace* phases, because these two phases are related to the threshold t and number n. The simulation was implemented in Java by using a computer of 64 bits Windows 10 with 1 core Intel i9 3.60 GHz and 32 GB RAM, which environment supports the time to perform a multiplication is about 0.03ms and an exponentiation is about 10ms. From TABLE 5, when $(t, n) = (3, 5)$, the *TKGen* phase runtime is 0.579s. When the participating regulatory manager n and tracing threshold t increse, for example, $(t, n) = (10, 20)$, the *TKGen* phase runtime is 30.828s. The reason for big gap in time is because each trace manager needs to communicate and verify each other. When $(t, n) = (3, 5)$, the *Trace* phase runtime is 0.03s. When the participating regulatory manager n and tracing threshold t increse, for example, $(t, n) = (20, 40)$, the *Trace* phase runtime is 0.211s. The reason for small difference in time is because each manager only needs to perform verify operation. From FIGURE 3, we can see,

Trace phase runtime remains highly efficient as the threshold (t, n) increases. But the *TKGen* phase runtime increases with a quadratic curve function, because the interaction process increases with the number of participants during the *TKGen* algorithm.

E. APPLICATION ANALYSIS

In the scenario application of the proposed DDT-LGS scheme, we add this mechanism to P2P electricity transaction model to realize multi-party audit and regulatory functions, and then deploy the participating nodes of the scheme to P2P network. At this time, we need to consider the time complexity and space complexity of the P2P network. In this paper, a typical distributed hash tables (DHT) protocol such as Chord can be used to construct a structured P2P network during deployment process. Because the time complexity and space complexity of Chord protocol both are $\mathcal{O}(logN)$, which can accommodate a large number of participating nodes. Chord protocol has good scalability to support the dynamic join and exit of nodes, making the scheme implementation can be done on online monitoring system.

V. CONCLUSION

In this paper, we proposed an multi-party audit and regulatory mechanism based on dynamic distributed traceable linkable group signature, and further extended it to P2P electricity transaction. The system architecture is constructed based on basic cryptography blocks. Then, we formally defined the dynamic distributed traceable linkable group signature scheme and gave specific instantiation, which satisfying the security properties of anonymity and traceability. Finally, performance analysis shows that the proposed scheme is efficient and practical. In future work, we will further optimize the threshold performance and solve the bottleneck of efficiency.

REFERENCES

[1] F. A. Wolak, "Diagnosing the California electricity crisis," *Electr. J.*, vol. 16, no. 7, pp. 11–37, Aug. 2003.

[2] J. B. Bushnell and E. T. Mansur, "Consumption under noisy price signals: A study of electricity retail rate deregulation in San Diego," *J. Ind. Econ.*, vol. 53, no. 4, pp. 493–513, Dec. 2005.

[3] F. Teng, Q. Zhang, G. Wang, J. Liu, and H. Li, "A comprehensive review of energy blockchain: Application scenarios and development trends," *Int. J. Energy Res.*, vol. 45, no. 12, pp. 17515–17531, Oct. 2021.

[4] M. L. Di Silvestre, P. Gallo, J. M. Guerrero, R. Musca, E. R. Sanseverino, G. Sciumè, J. C. Vásquez, and G. Zizzo, "Blockchain for power systems: Current trends and future applications," *Renew. Sustain. Energy Rev.*, vol. 119, Mar. 2020, Art. no. 109585.

[5] J. A. P. Lopes, N. Hatziargyriou, J. Mutale, P. Djapic, and N. Jenkins, "Integrating distributed generation into electric power systems: A review of drivers, challenges and opportunities," *Electric Power Syst. Res.*, vol. 77, no. 9, pp. 1189–1203, Jul. 2007.

[6] G. Mendes, C. Ioakimidis, and P. Ferrão, "On the planning and analysis of integrated community energy systems: A review and survey of available tools," *Renew. Sustain. Energy Rev.*, vol. 15, no. 9, pp. 4836–4854, Dec. 2011.

[7] Y. Parag and B. K. Sovacool, "Electricity market design for the prosumer era," *Nature Energy*, vol. 1, no. 4, pp. 1–6, Mar. 2016.

[8] T. Sousa, T. Soares, P. Pinson, F. Moret, T. Baroche, and E. Sorin, "Peer-to-peer and community-based markets: A comprehensive review," *Renew. Sustain. Energy Rev.*, vol. 104, pp. 367–378, Apr. 2019.

[9] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Decentralized Bus. Rev.*, 2008.

[10] V. Buterin et al., "A next-generation smart contract and decentralized application platform," *White Paper*, vol. 3, no. 37, pp. 1–2, 2014.

[11] H. van Soest, "Peer-to-peer electricity trading: A review of the legal context," *Competition Regulation Netw. Industries*, vol. 19, nos. 3–4, pp. 180–199, Sep. 2018.

[12] J. Anderson, "Regulatory and market design upgrades could accelerate peer-to-peer energy trading," *Platts Megawatt Daily*, vol. 23, no. 87, pp. 5–6, 2018.

[13] A. Saxena, "Optimized fractional overhead power term polynomial grey model (OFOPGM) for market clearing price prediction," *Electric Power Syst. Res.*, vol. 214, Jan. 2023, Art. no. 108800.

[14] X. Li, F. Zhou, and X. Yang, "A multi-dimensional trust evaluation model for large-scale P2P computing," *J. Parallel Distrib. Comput.*, vol. 71, no. 6, pp. 837–847, Jun. 2011.

[15] A. Schneiders, M. J. Fell, and C. Nolden, "Peer-to-peer electricity trading and the sharing economy: Social, markets and regulatory perspectives," *Energy Sources, B, Econ., Planning, Policy*, vol. 17, no. 1, Dec. 2022, Art. no. 2050849.

[16] C. Zhang, J. Wu, Y. Zhou, M. Cheng, and C. Long, "Peer-to-peer energy trading in a microgrid," *Appl. Energy*, vol. 220, pp. 1–12, Jun. 2018.

[17] N. Wang, W. Xu, Z. Xu, and W. Shao, "Peer-to-peer energy trading among microgrids with multidimensional willingness," *Energies*, vol. 11, no. 12, p. 3312, Nov. 2018.

[18] J. Abdella and K. Shuaib, "Peer to peer distributed energy trading in smart grids: A survey," *Energies*, vol. 11, no. 6, p. 1560, Jun. 2018.

[19] Y. Wu, T. Zhao, H. Yan, M. Liu, and N. Liu, "Hierarchical hybrid multi-agent deep reinforcement learning for peer-to-peer energy trading among multiple heterogeneous microgrids," *IEEE Trans. Smart Grid*, vol. 14, no. 6, pp. 4649–4665, Nov. 2023.

[20] X. Guo, G. Zhang, and Y. Zhang, "A comprehensive review of blockchain technology-enabled smart manufacturing: A framework, challenges and future research directions," *Sensors*, vol. 23, no. 1, p. 155, Dec. 2022.

[21] S. N. Khan, F. Loukil, C. Ghedira-Guegan, E. Benkhelifa, and A. Bani-Hani, "Blockchain smart contracts: Applications, challenges, and future trends," *Peer Peer Netw. Appl.*, vol. 14, pp. 2901–2925, Apr. 2021.

[22] A. Pareek, P. Singh, and J. Lather, "Blockchain technology in smart grids and microgrids: A critical review of challenges and opportunities," in *Power Electronics and High Voltage in Smart Grid*, 2022, pp. 353–363.

[23] H. Li, F. Xiao, L. Yin, and F. Wu, "Application of blockchain technology in energy trading: A review," *Frontiers Energy Res.*, vol. 9, Apr. 2021, Art. no. 671133.

[24] E. Mengelkamp, J. Gärttner, K. Rock, S. Kessler, L. Orsini, and C. Weinhardt, "Designing microgrid energy markets: A case study: The Brooklyn Microgrid," *Appl. Energy*, vol. 210, pp. 870–880, Jan. 2018.

[25] P. Ledger, "Power ledger white paper," Power Ledger Pty Ltd, Saint Georges Terrace Perth, WA, Australia, White Paper, vol. 8, 2017.

[26] N. Z. Aitzhan and D. Svetinovic, "Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams," *IEEE Trans. Dependable Secure Comput.*, vol. 15, no. 5, pp. 840–852, Sep. 2018.

[27] Y.-B. Son, J.-H. Im, H.-Y. Kwon, S.-Y. Jeon, and M.-K. Lee, "Privacy-preserving peer-to-peer energy trading in blockchain-enabled smart grids using functional encryption," *Energies*, vol. 13, no. 6, p. 1321, Mar. 2020.

[28] J. Ping, Z. Yan, and S. Chen, "A privacy-preserving blockchain-based method to optimize energy trading," *IEEE Trans. Smart Grid*, vol. 14, no. 2, pp. 1148–1157, Mar. 2023.

[29] M. J. A. Baig, M. T. Iqbal, M. Jamil, and J. Khan, "Blockchain-based peer-to-peer energy trading system using open-source angular framework and hypertext transfer protocol," *Electronics*, vol. 12, no. 2, p. 287, Jan. 2023.

[30] A. Schneiders and D. Shipworth, "Energy cooperatives: A missing piece of the peer-to-peer energy regulation puzzle?" SSRN 3252486, Tech. Rep., 2018.

[31] E. A. Soto, L. B. Bosman, E. Wollega, and W. D. Leon-Salas, "Peer-to-peer energy trading: A review of the literature," *Appl. Energy*, vol. 283, Feb. 2021, Art. no. 116268.

[32] P. A. Gbadega and Y. Sun, "Centralized peer-to-peer transactive energy market approach in a prosumer-centric residential smart grid environment," *Energy Rep.*, vol. 8, pp. 105–116, Nov. 2022.

[33] L. Zhihu, Z. Lin, X. Haiqing, C. Fangyuan, W. Nina, Z. Chunying, and Z. Bing, "Research on privacy protection technology of regulatory power blockchain transaction," *Cryptograph. J.*, vol. 9, no. 6, pp. 1014–1027, 2022.
[34] G. Gangjun, W. Peifang, S. Yue, Z. Tong, Y. Haixia, L. Xiangjun, and W. Yafeng, "Unified supervision and shared transaction model of power data under blockchain," *Inf. Technol. Netw. Secur.*, no. 3, p. 6, 2019.
[35] L. M. Kohnfelder, "Towards a practical public-key cryptosystem," Ph.D. thesis, Massachusetts Inst. Technol., Cambridge, MA, USA, 1978.
[36] D. Chaum and E. Van Heyst, "Group signatures," in *Advances in Cryptology–EUROCRYPT'91: Workshop on the Theory and Application of Cryptographic Techniques*. Brighton, U.K.: Springer, Apr. 1991, pp. 257–265.
[37] P. Feldman, "A practical scheme for non-interactive verifiable secret sharing," in *Proc. 28th Annu. Symp. Found. Comput. Sci. (SFCS)*, Oct. 1987, pp. 427–438.
[38] T. P. Pedersen, "Non-interactive and information-theoretic secure verifiable secret sharing," in *Proc. Annu. Int. Cryptol. Conf.* Cham, Switzerland: Springer, 1991, pp. 129–140.
[39] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in *Proc. Annu. Int. Cryptol. Conf.* Cham, Switzerland: Springer, 2004, pp. 41–55.
[40] M. Manulis, A.-R. Sadeghi, and J. Schwenk, "Linkable democratic group signatures," in *Information Security Practice and Experience*, 2nd ed. Hangzhou, China, Springer, Apr. 2006, pp. 187–201.
[41] T. Lu, J. Li, L. Zhang, and K.-Y. Lam, "Group signatures with decentralized tracing," in *Information Security and Cryptology*, Nanjing, China: Springer, Dec. 2020, pp. 435–442.
[42] H. Zheng, Q. Wu, B. Qin, L. Zhong, S. He, and J. Liu, "Linkable group signature for auditing anonymous communication," in *Proc. Australas. Conf. Inf. Secur. Privacy*, Wollongong, NSW, Australia, Cham, Switzerland: Springer, Jul. 2018, pp. 304–321.

ZHIHU LI was born in 1975. He received the M.S. degree in applied mathematics from the Beijing University of Posts and Telecommunications, in 2004. His main research interests include cryptography application technology, data security, and artificial intelligence technology.

BING ZHAO was born in 1971. He received the M.S. degree in pattern recognition and intelligent system from the Beijing University of Technology, Beijing, China, in 2002, and the Ph.D. degree in engineering from North China Electric Power University, Beijing, in 2022. He is currently the Director of the Metrology Research Institute, China Electric Power Research Institute. His main research interests include cryptography application and intelligent power technology.

HONGXIA GUO is currently with the Marketing Service Center (Metrology Center), State Grid Shandong Electric Power Company. Her main research interests include smart grids, energy metering, and electrical information collection.

FENG ZHAI was born in 1979. He received the M.S. degree in circuits and systems from North China Electric Power University, Beijing, China, in 2010. He is currently the Director of the Research Section, Metrology Research Institute, China Electric Power Research Institute. His main research interests include cryptography application technology and intelligent power technology.

LIN LI received the master's degree in automation from Shandong University, in 2018. She is currently with the Marketing Service Center (Metrology Center), State Grid Shandong Electric Power Company. Her main research interests include smart grids and energy metering.