



Article

Secure Ring Signature Scheme for Privacy-Preserving Blockchain

Lin Wang ¹, Changgen Peng ^{2,*}  and Weijie Tan ^{1,2} 

¹ State Key Laboratory of Public Big Data, College of Computer Science and Technology, Guizhou University, Guiyang 550025, China; gs.linwang20@gzu.edu.cn (L.W.); wjtan@gzu.edu.cn (W.T.)

² Key Laboratory of Advanced Manufacturing Technology, Ministry of Education, Guizhou University, Guiyang 550025, China

* Correspondence: cgpeng@gzu.edu.cn

Abstract: Blockchain integrates peer-to-peer networks, distributed consensus, smart contracts, cryptography, etc. It has the unique advantages of weak centralization, anti-tampering, traceability, openness, transparency, etc., and is widely used in various fields, e.g., finance and healthcare. However, due to its open and transparent nature, attackers can analyze the ledger information through clustering techniques to correlate the identities between anonymous and real users in the blockchain system, posing a serious risk of privacy leakage. The ring signature is one of the digital signatures that achieves the unconditional anonymity of the signer. Therefore, by leveraging Distributed Key Generation (DKG) and Elliptic Curve Cryptography (ECC), a blockchain-enabled secure ring signature scheme is proposed. Under the same security parameters, the signature constructed on ECC has higher security in comparison to the schemes using bilinear pairing. In addition, the system master key is generated by using the distributed key agreement, which avoids the traditional method of relying on a trusted third authorizer (TA) to distribute the key and prevents the key leakage when the TA is not authentic or suffers from malicious attacks. Moreover, the performance analysis showed the feasibility of the proposed scheme while the security was ensured.

Keywords: blockchain; ring signature; privacy protection; distributed key generation; elliptic curve cryptography



Citation: Wang, L.; Peng, C.; Tan, W. Secure Ring Signature Scheme for Privacy-Preserving Blockchain. *Entropy* **2023**, *25*, 1334. <https://doi.org/10.3390/e25091334>

Academic Editors: Bill William Buchanan, Jawad Ahmad and Arslan Munir

Received: 7 August 2023

Revised: 9 September 2023

Accepted: 13 September 2023

Published: 14 September 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Blockchain technology is the underlying technology of Bitcoin, a concept mentioned by Nakamoto [1] in his Bitcoin white paper published in 2008, which has triggered new industrial and technological revolutions and is a popular research area at present. In essence, blockchain is a kind of distributed database [2], and its underlying chain structure is the data blocks arranged in chronological order, which inherits the technologies of smart contracts [3], peer-to-peer networks (P2P) [4], the consensus mechanism [5], cryptography, etc., has the unique advantages of weak centralization, being tamper-proof, traceability, openness, transparency, etc., and is able to realize the direct circulation of value between untrusted nodes without the need for third-party institutions, which not only reduces the trust cost of transactions, but also greatly shortens the interaction time. It is considered as a key technology to realize the transformation of the “information Internet” to the “value Internet” [6].

Blockchain’s technical advantages make it applicable to various domains. One of them is digital currencies, where Bitcoin and its derivatives are expanding fast. According to a 2017 report by ARK Investments, there are more than 10 million Bitcoin users worldwide, with over USD 150-million in daily transactions [7]. In the financial sector, blockchain technology is highly valued by central banks, who design their own digital currencies by applying or studying it. They use blockchain technology to enhance the traditional financial system, which suffers from long delays and low efficiency in reconciliation, clearing, and cross-border settlement, as well as the high costs of maintaining central ledger data.

In healthcare, the companies and stakeholders are benefiting from blockchain technology, which helps them streamline business processes, improve patient outcomes, manage patient data, comply with regulations, cut costs, and leverage healthcare-related data more effectively [8]. In the energy sector, blockchain technology, which is based on decentralization, is seen as a game-changing technology for creating distributed energy systems. It offers a decentralized trust mechanism that can be used for distributed energy operations and can help overcome management weaknesses and challenges in distributed energy systems [9]. In the field of cultural industry, the data inerrancy and high trustworthiness of the blockchain industry can be utilized to carry out many businesses such as certificate storage, digital property rights protection, and cultural relic identification. Furthermore, there are other fields such as justice, military, and supply chains that are gradually using blockchain to improve the problems existing in each field.

As blockchain technology continues to develop and become widely used, the privacy leakage concerns confronting it are becoming increasingly noticeable and must be given sufficient attention. While the blockchain mechanism avoids potential failures of individual servers and the exposure of data in terms of data storage, all transactions have to be made public to all nodes in order to reach consensus among the decentralized blockchain nodes. This exposes the privacy of the transactions greatly, and a main challenge is the safeguarding of users' identity privacy. Identity privacy refers to the connection between a user's real identity and the blockchain nodes. The blockchain stores data in an unchangeable way as a distributed ledger that any node can access. Transactions on the blockchain are somewhat anonymous, but not completely secure against privacy breaches. With advanced computing techniques, an attacker can track and analyze the correlation of public data in a global ledger to reveal sensitive information. For example, if there are consistent and correlated transactions, an attacker can extract some user characteristics using a graph of transactions between different addresses [10]. Moreover, an attacker can search all possible transactions to obtain the transaction addresses and estimate the balances, which can help to infer the user's identity and location [11]. Cryptography is the prevalent method for privacy protection among researchers. Therefore, it should be combined with blockchain technology to offer a suitable solution for the blockchain's privacy issue and guarantee the safety of users' data.

A common security technique in blockchain systems is to use digital signatures to check the integrity of a document or a message. This ensures non-repudiation. The ring signature scheme [12] is a special type of group signature [13], which uses a group of L in a user's private key and L in all the users' public keys to complete the signature; the signature of the verifier can only verify that the signature comes from the group, but who signed the name is unknown. The ring signature does not need a trusted center and can hide the identity of the signer, which protects the user's privacy. Considering its features, the scheme can be applied to anonymous payment applications or untraceable transactions and also to other scenarios that require privacy protection, such as elections, voting, and identity verification. Therefore, we propose a new privacy protection scheme by combining the ring signature algorithm and blockchain technology. However, the key generation link of traditional ring signature algorithms requires a secret key management center, which will face the risk of key leakage and increase the probability of attackers forging ring signatures. Moreover, most of the ring signatures are constructed based on bilinear pairing, and the computational complexity is usually high, which may lead to longer signing and verification time. It also lacks security. In order to improve the efficiency and security of ring signatures, we also made a new design of the scheme.

To solve the above-mentioned issues in the blockchain system, this paper studied and analyzed the related ring signature schemes and designed a new ring signature scheme suitable for the blockchain environment by using the anonymity feature of the ring signature. The scheme is based on Shamir verification secret sharing theory [14] and the Feldman protocol [15]; the key generation link in the scheme was improved, and the main body of the scheme adopted the elliptic curve cryptography principle.

The main contributions of the paper are stated as follows:

1. The algorithm exploits the concept of distributed key generation to create a system master key, which enhances the process of distributing the key by a trusted authorizer (TA) in traditional signature algorithms and eliminates the risk of key leakage when the TA is untrustworthy or subjected to malicious attacks.
2. This scheme is a ring signature constructed based on ECC, which provides better security with the same length of key compared to the scheme based on bilinear pairing. This algorithm strengthens the signature's unforgeability, which reduces the attackers' probability of succeeding in cracking the key.
3. This scheme improves the efficiency of ring signature generation and verification and is more compatible with the environment of blockchain systems.

The rest of this paper is organized as follows. In Section 2, we review different blockchain privacy-preserving schemes. In Section 3, we provide some preliminary concepts, including general ring signature algorithms and security model definitions. Section 4 describes the algorithms and system models of secure ring signature schemes. Section 5 proves the security of the proposed scheme, and Section 6 gives the efficiency and performance comparison of the signature schemes. Finally, the paper is summarized in Section 7.

2. Related Works

Blockchain technology faces challenges in its development in existing industries due to privacy breaches and other security issues. To protect users' identity privacy in blockchain, different schemes have been suggested to increase the anonymity level. In order to resist the book analysis technique, researchers propose a defense mechanism for exchanging assets and obfuscating addresses, i.e., the address obfuscation mechanism, in response to the assumptions on which the technique is based. In 2014, Bonneau et al. [16] proposed the Mixcoin protocol, which enhances asset security through an electronic-signature-based commitment mechanism. In 2015, Valenta et al. [17] proposed the Blindcoin protocol, which guarantees the internal privacy of centralized hybrid coin schemes by using blind signature techniques. In 2018, Ziegeldorf et al. [18] proposed CoinParty; it occupies a unique position in the design space of hybrid services by decrypting the novel combination of hybrid nets and threshold signatures, combining the advantages of previously proposed centralized and decentralized hybrid services into a single system. The address obfuscation mechanism can protect the privacy of the ledger to a certain extent; however, the result of address obfuscation will still be stored in the public ledger, and an attacker can threaten user privacy to a certain extent by analyzing the obfuscated transactions with features.

Ledger information hiding mainly preserves the confidentiality of the ledger by encrypting the private data in the ledger and provides "credentials" through cryptographic techniques to keep the correctness of the blockchain ledger verifiable. Most of the existing techniques for the implementation of the ledger-information-hiding mechanism belong to zero-knowledge proof techniques [19], which means that zero-knowledge proofs do not convey any proof of knowledge other than the correctness of the proposition under discussion. Li et al. [20], based on the ring-zero-proof-of-knowledge and blockchain technology, proposed a secure and efficient fair transaction mechanism for a sharing environment. The mechanism utilizes ring-zero-knowledge proofs to hide transaction contents and relationships without affecting authentication by adding a new trusted player. However, zero-knowledge proofs have a high computational cost and storage space, require the use of complex cryptographic algorithms and a large number of data structures, and may affect the performance and scalability of blockchain systems.

Ring signature schemes allow participants to sign messages with the group name and preserve the confidentiality of the signer's identity from disclosure. The ring signature technique has two main features: first, any member of the group can issue the correct signature alone; second, any member of the group only knows whether he or she has initiated the signature, and members outside the group only know whether the signer belongs to the group. A higher level of privacy is achieved by VOTOR, a practical remote

voting scheme that uses product anonymization channels and linkable ring signatures. It was proposed by Thomas et al. [21]. An information sharing system that ensures the confidentiality of applicants was created by Patil et al. [22] using an ID-based ring signature technique. They removed the certificate validation process to make the system secure and reliable. A new ring signature scheme based on elliptic curves was proposed by Li et al. [23], which improves the signature unforgeability and anonymity compared with the traditional ring signature scheme. Wang et al. [24] proposed a flexible threshold ring signature scheme in chronological order, which in practice has the advantage of solving both the update problem and the chronological order problem. In addressing the difficulty of sharing medical records between healthcare organizations, Lai et al. [25] introduced a secure medical-data-sharing scheme based on traceable ring signatures and blockchain. Samra et al. [26] proposed a new framework, a certificate-less aggregation scheme based on traceable ring signatures (CLA-TRS), which ensures conditional privacy-preserving authentication in vehicular ad-hoc network (VANET) communications. Table 1 summarizes the application scenarios, techniques, advantages, etc., of the above schemes. It can be seen that most of the above schemes are based on bilinear pairing construction, and their security needs to be improved. Moreover, most of them rely on a trusted key generation center (KGC), which cannot avoid attacks and reduces the difficulty of forging signatures.

Table 1. Comparison of ring signature schemes.

Scheme	Scenario	Techniques	Advantages	Drawbacks
[21]	Vote	Bilinear Hash Anonymous-channel	Linkable Practical	Relies on trusted center Lack of efficiency analysis
[22]	Cloud computing	Bilinear Hash ID-based	Simplified management High efficiency	Relies on trusted center Does not support key revocation and update
[23]	Blockchain	ECC Hash	Improves unforgeability Improves anonymity	Lack of efficiency analysis Relies on trusted center
[24]	Edge computing	Bilinear Hash Threshold	Flexible Renewable	Relies on trusted center Lack of efficiency comparison of related schemes
[25]	Medical sharing	Bilinear Hash DKG	Traceable Controllable	High computational cost
[26]	VANET	Bilinear Hash ECC	Traceable High efficiency	Relies on trusted center

3. Preliminaries

In this context, we will present some preliminary knowledge including elliptic curves, difficult assumptions, and the general ring signature algorithm and its security models below.

3.1. Elliptic Curve

An elliptic curve is not an ellipse; it is called an elliptic curve because the equation for the curve is similar to the equation for calculating the perimeter of an ellipse. In general, the curve equation of an elliptic curve is a cubic equation of the following form:

$$y^2 + axy + by = x^3 + cx^2 + dx + e, \quad (1)$$

where a, b, c, d , and e are real numbers satisfying some simple conditions.

Elliptic curves over finite fields are commonly used in cryptography, which refers to the curve defined by Equation (1) in which all coefficients are elements in a finite field $GF(q)$, where q is a large prime number. The most-commonly used of these is the curve defined by Equation (2):

$$y^2 \bmod q = (x^3 + ax + b) \bmod q, \quad (2)$$

where $a, b \in GF(q)$ and $\Delta = (4a^3 + 27b^2) \bmod q \neq 0$.

An elliptic curve is symmetric with respect to the X-axis, and the addition operation on it is defined as follows: if three points lie on the same line, the sum of them is O . Addition on an elliptic curve is defined as follows:

1. O is the additive identity element, that is, for any point P of the elliptic curve, $P + O = P$.
2. Let $P_1 = (x, y)$ be a point on an elliptic curve whose additive inverse element is defined as $P_2 = -P_1 = (x, -y)$. This is because, when the connection of P_1 and P_2 is extended to infinity, another point O on the elliptic curve is obtained, that is the three points P_1, P_2 , and O on the elliptic curve are collinear, so $P_1 + P_2 + O = O, P_1 + P_2 = -O$, that is $P_2 = -P_1$.
3. Let $P = (x_p, y_p), Q = (x_q, y_q)$ and $P \neq -Q$, then $R = P + Q = (x_r, y_r)$ is determined by the following rule:

$$x_r \equiv (\lambda^2 - x_p - x_q) \bmod q, \quad (1)$$

$$y_r \equiv (\lambda^2(x_p - x_r) - y_p) \bmod q, \quad (2)$$

where

$$\lambda = \begin{cases} \frac{y_q - y_p}{x_q - x_p} \bmod q, & \text{if } P \neq Q \\ \frac{3x_p^2 + a}{2y_p} \bmod q, & \text{if } P = Q \end{cases} \quad (3)$$

4. The multiple of a point P is defined as $2P = P + P$.

3.2. Problem Assumptions

Definition 1. Elliptic curve discrete logarithm problem (ECDLP): Given any two points P, Q on an elliptic curve $E(F_p)$, solving for the value x satisfying the equation $Q = x \cdot P$ is unsolvable in polynomial time.

3.3. Ring-Signature-Generation Algorithm

The ring signature is a unique type of group signature that uses a set of public keys instead of one. It hides the identity of the actual signer from the verifier. Unlike other group signatures, ring signatures do not require a manager or any coordination among the members. The basic ring signature has three components: KeyGen(), Sign(), and Verify():

- KeyGen(): This algorithm needs to input a security parameter l and, then, generate a key pair (pk, sk) for each user, where pk is the public key and sk is the private key.
- Sign(): This algorithm takes the message m , which needs to be encrypted, the private key sk of a ring member, and the public key set $L = \{pk_1, pk_2, \dots, pk_n\}$ of the selected ring members and generates a signature σ for the message m . One of the parameters in the signature σ follows a ring according to certain rules.
- Verify(): This algorithm is a deterministic algorithm, which takes the public key set $L = \{pk_1, pk_2, \dots, pk_n\}$, the message m , and the signature σ as the input and outputs “accept” if the verification passes and “reject” otherwise.

3.4. Security Models

The ring signature scheme is supposed to satisfy the requirements of correctness, unconditional anonymity, and unforgeability.

3.4.1. Game I Correctness

The output of the ring-signature-generation algorithm serves as the input for the ring signature verification algorithm, which always outputs acceptance. The unforgeability and unconditional anonymity of a ring signature scheme are defined by a game between a simulator \mathcal{R} and an adversary \mathcal{A} . To begin with, we introduce \mathcal{A} 's inquirable oracle machines JO, CO, and SO:

- Join oracle machine ($\text{JO}(\perp) \rightarrow PK$): With this query, a new user is added to the system and the public key PK of the new user is output.
- Corruption oracle machine ($\text{CO}(PK_i) \rightarrow sk_i$): The user's public key PK_i is input, and the corresponding private key sk_i is output.
- Signed oracle machine ($\text{SO}(m, n, L, PK) \rightarrow \sigma$): Input signed message m , and set $L = \{PK_1, PK_2, \dots, PK_n\}$ of public keys of size n ; the signer's public key PK_π ($1 \leq \pi \leq n$) returns a valid ring signature σ .

The definition of the general ring signature and the ring signature defined in this paper contains four basic algorithms: system initialization algorithm, key-generation algorithm, ring signature generation, and ring signature verification. The key point is that the general ring signature puts forward more-specific requirements for the signature-value-generation process: Given a message M , the public key $(PK_1, PK_2, \dots, PK_n)$ of n members, the signer's private key sk_π ($1 \leq \pi \leq n$), and a secure hash function, produce a set $r_1, r_2, \dots, r_n, h_1, h_2, \dots, h_n$, and finally, output the signature value σ , where $r_i \neq r_j, i \neq j, h_i$ ($1 \leq i \leq n$) are the hash values determined by m, r_i ($1 \leq i \leq n$) and the public keys of the ring members, the signature value σ is completely determined by $r_1, r_2, \dots, r_n, h_1, h_2, \dots, h_n$, and message m is decided.

3.4.2. Game II Unforgeability

The unforgeability of a ring signature is defined by the following game between a simulator \mathcal{R} and an adversary \mathcal{A} :

- \mathcal{R} generates the system parameters params and sends them to \mathcal{A} .
- \mathcal{A} adaptively queries oracles JO, CO, and SO and random oracles \mathcal{H} .
- \mathcal{A} outputs a signature message M^* , a set S^* consisting of n user public keys, and two forged signature values σ_0^*, σ_1^* .

\mathcal{A} is said to have won the above game if the following four conditions are met:

Step 1: σ_0^*, σ_1^* are valid ring signatures on the message M^* , that is $R\text{Verify}(M^*, S^*, \sigma_i^*)$, ($i \in \{0, 1\}$) \rightarrow accept.

Step 2: All public keys in S^* are obtained by querying the oracle JO.

Step 3: All the public keys in S^* are not corrupted, that is the adversary cannot obtain the private keys of any ring member in S^* .

Step 4: σ^* is not obtained by querying the signed oracle machine SO.

A ring signature scheme is said to be unforgeable if, for any PPT adversary \mathcal{A} , the probability of winning the above game is negligible.

3.4.3. Game III Unconditional Anonymity

The unconditional anonymity of a ring signature scheme is defined by a game between a simulator \mathcal{R} and an adversary \mathcal{A} with infinite computational power as follows:

- \mathcal{R} generates the system parameters params and sends them to \mathcal{A} .
- \mathcal{A} can adaptively query join oracle machine JO.
- \mathcal{A} sends a signature message M^* and a set $S^* = PK_1, PK_2, \dots, PK_n$ consisting of public keys of n users to \mathcal{R} , where all public keys are obtained by the JO query. \mathcal{R} ran-

domly selects $\pi \in \{1, 2, \dots, n\}$ and computes a signature $\sigma_\pi = \text{Sign}(M^*, n, S^*, sk_\pi)$, where sk_π is the private key corresponding to PK_π . Finally, \mathcal{R} sends σ_π to \mathcal{A} .

- \mathcal{A} outputs a guess $\pi' \in \{1, 2, \dots, n\}$.

A ring signature is said to satisfy unconditional anonymity if, for an adversary \mathcal{A} with infinite computing power, the probability of guessing the correct signer π is at most $\frac{1}{n}$, where n is the cardinality of the public key set S .

4. Secure Ring Signature Scheme

In this section, a secure blockchain ring signature scheme is proposed by incorporating the idea of distributed key generation. The following is a detailed description of the system model and the signature algorithm.

4.1. System Description

The system model is shown in Figure 1 and contains entities such as the group users, the distributed generation center KDC, and the blockchain network. The purpose of the KDC here is to generate the system parameters, validate and manage the cluster personnel, and verify the signature. The specific scheme is described as follows:

Step 1: The KDC picks the security parameter l and generates the system parameters for the signature.

Step 2: The user A_π applies to the KDC to become a member of the group; A_π sends ID_π to the KDC; the KDC passes the verification, returns Q_π , and adds the user A_π to the group; the user completes the registration.

Step 3: The member A_π interacts with other group members A_i ($i = 1, 2, \dots, n, i \neq \pi$) through a secure channel to generate the system's master private key and master public key.

Step 4: The member A_π signs the data message m to generate signature σ_π .

Step 5: All KDCs in the blockchain system verify the signature σ_π and upload the data information and signatures to the blockchain database after verification.

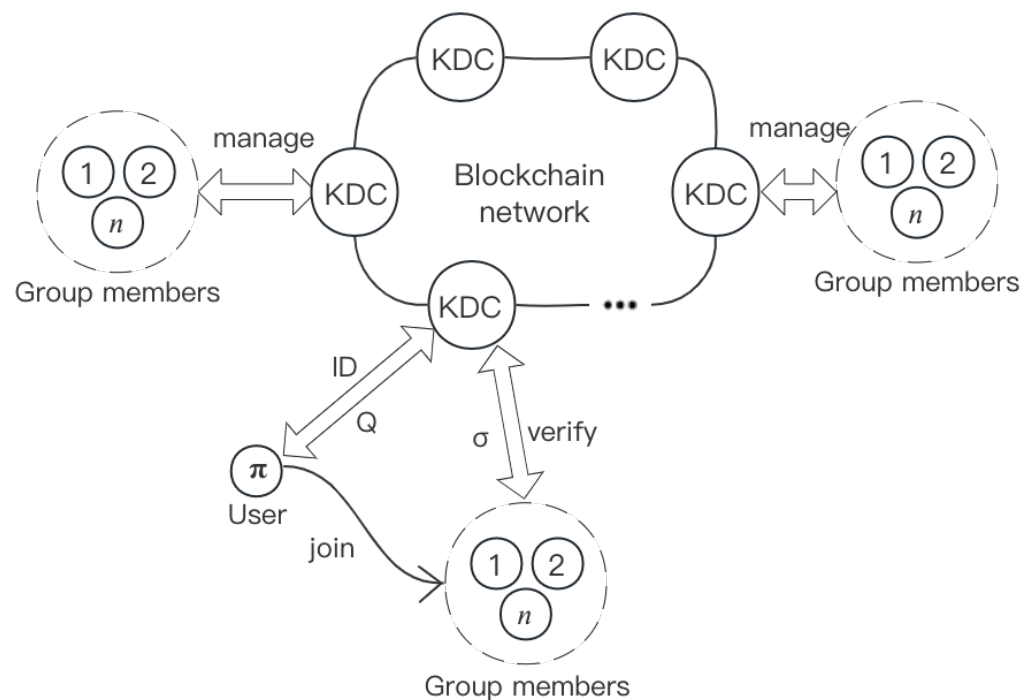


Figure 1. System model.

Miner nodes in the blockchain network pack the set of transactions for a period of time and then continuously calculate the random numbers that meet the conditions to construct blocks that meet the predefined conditions for confirming the transactions. The KDCs

mentioned in this paper can be considered as miner nodes, also known as full nodes, and users as regular nodes. Full nodes act as servers in the distributed network, and they maintain consensus rules among other nodes, as well as transaction validation. Ordinary nodes retain some of the information on the block. After the group users sign the data information, all KDCs verify the signatures and upload them into the blockchain network.

4.2. Algorithm Description

4.2.1. Setup Algorithm

The system server selects security parameters l and randomly picks a large prime number $q > l$. G is a base point on the elliptic curve. Let G_1 be an additive group of order q generated by the generating element P . The hash functions are: $H_0 : \{0, 1\}^* \rightarrow E(F_q)$, $H_1 : \{0, 1\}^* \rightarrow Z_q^*$, $H_2 : \{0, 1\}^* \times G_1 \rightarrow Z_q^*$, $H_3 : \{0, 1\}^* \rightarrow G_1$. N represents the number of authorized users; $\{q, G, G_1, H_0, H_1, H_2, H_3, N\}$ is the public parameter. When the system parameters are determined, every authorized user A_i , ($i = 1, 2, \dots, N$) picks a random polynomial of degree $N - 1$, $f_i(x) = a_{i0} + a_{i1}x + a_{i2}x^2 + \dots + a_{i(N-1)}x^{(N-1)}$ over Z_q^* , where $f_i(0) = a_{i0}$. Each A_i computes and broadcasts $T_{ij} = P^{a_{ij}} \pmod{q}$ ($j = 0, 1, \dots, N - 1$). Meanwhile, every authorized user A_i transmits the calculated secret value $s_{ik} = f_i(k)$ through a secure channel to the other A_k ($k = 0, 1, 2, \dots, N - 1, k \neq i$) in the group. After that, every A_k receives the secret value s_{ik} and determines if it is correct by using the formula $P^{s_{ik}} \stackrel{?}{=} \prod_{j=0}^{N-1} (T_{ij})^{s_{ij}} \pmod{q}$. If the equation is not satisfied, the secret value is wrong, and A_k sends an error message to A_i , who has to resend the right secret value until the equation holds. Then, the master key $S = \sum_{i=1}^{N-1} b_{i0} \pmod{q}$ and the master public key $P_0 = S \cdot P$ are established by the N authority members.

4.2.2. Key Generation

For each authorized signer of the system, A_i transmits its identity information ID_i to the KDC. Then, the KDC randomly selects $k_i \in Z_q^*$, computes $Q_i = H_3(ID_i || k_i)$, and secretly transmits it to A_i over a secure channel. Next, signer A_i computes $D_i = S \cdot Q_i$, randomly picks $x_i \in Z_q^*$, and computes its private key $sk_i = H_2(x_i \cdot D_i)$ and public key $pk_i = sk_i * G$.

4.2.3. Signature Generation

Assuming that the signature user in the system is π , the public key is $pk_\pi = sk_\pi * G$, and the private key is $sk_\pi = H_2(x_\pi \cdot D_\pi)$. Choose a set $L = \{ID_1, ID_2, \dots, ID_n\}$ consisting of n identities of other authorized users of the system, and if the public key pk_π of the system is not in L , assign the attribute values Y_i, A_i for each public key pk_i as follows:

Step 1: Randomly chose $v_i, t_i, r_i \in Z_q^*$, and compute:

$$Y_i = \begin{cases} (v_i + t_i) * G, & \text{if } i = \pi \\ (t_i + r_i) * pk_i + v_i * G & \text{if } i \neq \pi \end{cases} \quad (4)$$

$$A_i = \begin{cases} (v_i + t_i) * H_0(pk_i), & \text{if } i = \pi \\ v_i * H_0(pk_i) + (t_i + r_i) * I_\pi & \text{if } i \neq \pi \end{cases} \quad (5)$$

where: $I_\pi = sk_\pi * H_0(pk_\pi)$ is a message signature image that prevents double-spending attacks in the system. It is obtained by mapping pk_i to a curve point in the finite field using $H_0(pk_i)$.

Step 2: Randomly select $s \in Z_q^*$, and then, calculate:

$$h = H_2(m || s). \quad (6)$$

$$c_i = \begin{cases} H_1(h, Y_1, \dots, Y_n, A_1, \dots, A_n) - \sum_{i=1, i \neq \pi}^n c_i & \text{if } i = \pi \\ t_i + r_i & \text{if } i \neq \pi, \end{cases} \quad (7)$$

$$d_i = \begin{cases} (v_i + t_i) - c_i * sk_i & \text{if } i = \pi \\ v_i & \text{if } i \neq \pi \end{cases} \quad (8)$$

where: m stands for the content of the signature, and the final output of the transaction initiator π 's ring signature for the message m is $\sigma = (I_\pi, c_1, c_2, \dots, c_\pi, \dots, c_n, d_1, d_2, \dots, d_\pi, \dots, d_n)$.

4.2.4. Verify

The following steps can be used to verify the transaction signature σ by anyone who possesses the public keys of all the members in the ring signature.

$$\begin{cases} \zeta_i = c_i * pk_i + d_i * G \\ \eta_i = c_i * I_\pi + d_i * H_0(pk_i) \end{cases} \quad (9)$$

$$\sum_{i=1}^n c_i = H_1(h, \zeta_1, \zeta_2, \dots, \zeta_n, \eta_1, \eta_2, \dots, \eta_n) \quad (10)$$

Compute ζ_i, η_i using Equation (9), and check if Equation (10) holds. If it does, the signature image I_π in the signature is not used, and the signature is valid. If it does not, the signature image I_π is used and the signature is invalid.

5. Security Analysis

5.1. Correctness Analysis

The verifier checks the transaction signature σ using Equation (10), and if it holds, the signature is valid. When $i \neq \pi$, the conversion of ζ_i is given as Equation (11) and η_i is given as Equation (12):

$$\begin{aligned} \zeta_i &= c_i * pk_i + d_i * G \\ &= (t_i + r_i) * pk_i + v_i * G \\ &= Y_i \end{aligned} \quad (11)$$

$$\begin{aligned} \eta_i &= d_i * H_0(pk_i) + c_i * I_\pi \\ &= v_i * H_0(pk_i) + (t_i + r_i) * I_\pi \\ &= A_i \end{aligned} \quad (12)$$

When $i = \pi$, the conversion of ζ_i, η_i is as follows:

$$\begin{aligned} \zeta_i &= c_i * pk_i + d_i * G \\ &= c_i * pk_i + [(v_i + t_i) - c_i * sk_i] * G \\ &= v_i * G + t_i * G \\ &= Y_i \end{aligned} \quad (13)$$

$$\begin{aligned} \eta_i &= d_i * H_0(pk_i) + c_i * I_\pi \\ &= [(v_i + t_i) - c_i * sk_i] * H_0(pk_i) + c_i * sk_\pi * H_0(pk_\pi) \\ &= v_i * H_0(pk_i) + t_i * H_0(pk_i) \\ &= A_i \end{aligned} \quad (14)$$

Therefore, based on the above relationship, the validity of our proposed scheme can be verified according to the following equation.

$$\begin{aligned}
 & H_1(h, \zeta_1, \zeta_2, \dots, \zeta_n, \eta_1, \eta_2, \dots, \eta_n) \\
 &= H_1(h, Y_1, Y_2, \dots, Y_\pi, \dots, Y_n, A_1, A_2, \dots, A_\pi, \dots, A_n) \\
 &= c_\pi + \sum_{i=1, i \neq \pi}^n c_i \\
 &= \sum_{i=1}^n c_i
 \end{aligned} \tag{15}$$

5.2. Unforgeability Analysis

Theorem 1. Under a randomized oracle model, the messages m can be chosen adaptively by adversary \mathcal{A} in Game II to attack. If there exists an algorithm that can win the ECDLP game in polynomial time T , then it is shown that the ECDLP hard problem can be broken with a non-negligible probability.

Proof. The purpose of challenger \mathcal{R} is to compute the value of a when provided with a random instance of the discrete logarithm problem (P, aP) . The challenger \mathcal{R} sets the public key of the signer U_* as: $pk_{i^*} = aP$. In this scenario, \mathcal{R} acts as a subroutine within \mathcal{A} and takes on the role of the challenger in Game II. To simplify the discussion, let us assume that all queries made by the attacker \mathcal{A} are distinct. Next, we elaborate on how the challenger \mathcal{R} deals with \mathcal{A} 's query:

Step 1, initialization: The challenger \mathcal{R} proceeds to execute the initialized algorithm with a security parameter of l to obtain the system parameters. Subsequently, these system parameters are transmitted from \mathcal{R} to the adversary \mathcal{A} .

Step 2, hash query: This step consists of the challenger \mathcal{R} creating an empty table L , where L holds pairs of two values, such as (x_i, y_i) , where the challenger \mathcal{R} randomly selects y_i and sets $H_1(x_i) = y_i$. When the adversary \mathcal{A} queries $H_1(x_i)$, \mathcal{R} hands over y_i to \mathcal{A} and appends (x_i, y_i) to list L .

Step 3, public key query: When the adversary \mathcal{A} queries the public key of a user, the challenger \mathcal{R} halts if $sk_i = sk_{i^*}$; otherwise, the challenger \mathcal{R} gives the matching user public key pk_i to adversary \mathcal{A} .

Step 4, private key query: When the adversary \mathcal{A} queries the private key of a user, if $pk_i = pk_{i^*}$, then \mathcal{R} stops operating, in the absence of this, the \mathcal{R} sends the appropriate user private key sk_i back to adversary \mathcal{A} .

Step 5, ring signature query: The adversary \mathcal{A} transmits information m and a public key collection L of N users to \mathcal{R} , which returns a corresponding signature σ . Suppose there is a user identity $pk_\pi \in L$ such that $pk_s \neq pk_{i^*}$, then the adversary \mathcal{A} signs the message using pk_π as the real signer and gives the signature σ . Alternatively, the adversary \mathcal{A} will conduct the next steps:

- Randomly choose $v_i, t_i, r_i, s \in \mathbb{Z}_q^*$, and compute:

$$Y_i = \begin{cases} (v_i + t_i) * G, & \text{if } i = \pi \\ v_i * G + (t_i + r_i) * pk_{i^*} & \text{if } i \neq \pi \end{cases} \tag{16}$$

$$A_i = \begin{cases} (v_i + t_i) * H_0(pk_{i^*}), & \text{if } i = \pi \\ v_i * H_0(pk_{i^*}) + (t_i + r_i) * I_\pi^* & \text{if } i \neq \pi \end{cases} \tag{17}$$

$$h = H_2(m || s), \tag{18}$$

$$c_i = \begin{cases} H_1(h, Y_1, \dots, Y_n, A_1, \dots, A_n) - \sum_{i=1, i \neq \pi}^n c_i & \text{if } i = \pi \\ t_i + r_i & \text{if } i \neq \pi \end{cases} \tag{19}$$

$$d_i = \begin{cases} (v_i + t_i) - c_i * sk_{i^*} & \text{if } i = \pi \\ v_i, & \text{if } i \neq \pi \end{cases} \quad (20)$$

- The ring signature is given as $\sigma^* = (I_\pi^*, c_1, c_2, \dots, c_\pi^*, \dots, d_1, d_2, \dots, d_\pi^*, \dots, d_n)$.

Step 6, forgery: At last, the adversary \mathcal{A} provides the signer pk_{i^*} with a signature with different information m^* . This same result can be obtained by the challenger \mathcal{R} , while both signatures σ and σ^* are valid: $\sigma = (I_\pi, c_1, c_2, \dots, c_\pi, \dots, d_1, d_2, \dots, d_\pi, \dots, d_n)$, $\sigma^* = (I_\pi^*, c_1, c_2, \dots, c_\pi^*, \dots, d_1, d_2, \dots, d_\pi^*, \dots, d_n)$.

The challenger \mathcal{R} returns the value corresponding to the private key $a = sk_\pi$.

Hence, the adversary \mathcal{A} for the instantiation (P, aP) can be found for $a = sk_\pi$, which means that the ECDLP is solved. \square

Assuming that \mathcal{A} can forge valid ring signatures with a non-negligible probability, there exists an algorithm \mathcal{R} that addresses the ECDLP in polynomial time. However, the ECDLP is known to be hard, so the probability of forging the ring signatures in our scheme will be negligible with a random oracle model. For the ring signature $\sigma = (I_\pi, c_1, c_2, \dots, c_\pi, \dots, d_1, d_2, \dots, d_\pi, \dots, d_n)$, the adversary needs to obtain the signer's private key in the computation even if the adversary randomly selects v_i, t_i, r_i to forge Y_i, A_i, Y_π, A_π , and d_π . In the proposed scheme, the user's public and private keys are calculated from the system's master key pair, which is jointly generated by the authorized user group according to the distributed key-generation algorithm and is not issued by the trusted third party. Therefore, there is no third-party attack, so the user's public and private keys are safe. Without knowing the key, it is infeasible to compute the key image $I_\pi = sk_\pi H_0(pk_\pi)$, so an attacker cannot create the signature σ . Therefore, the scheme in this paper is unforgeable.

5.3. Unconditional Anonymity

Theorem 2. *In the signature scheme proposed in this paper, the signer has unconditional anonymity, i.e., for any algorithm \mathcal{T} , any participant ensemble $L = pk_1, pk_2, \dots, pk_n$, and any $pk_\pi \in L$, the probability $P_r[pk = pk']$ is always $\frac{1}{2}$, where the signer of π creates a ring signature: $\sigma = (I_\pi, c_1, c_2, \dots, c_\pi, \dots, d_1, d_2, \dots, d_\pi, \dots, d_n)$.*

Proof. Step 1: The challenger \mathcal{R} computes the system parameters and gives them to the adversary \mathcal{A} .

Step 2: The adversary \mathcal{A} performs polynomially restricted ring signature queries adaptively.

Step 3: The adversary \mathcal{A} outputs a message m , two different public keys pk_1, pk_2 selected from the set of public keys L consisting of authorized users in the challenge phase, and delivers all of this information to \mathcal{R} . Next, \mathcal{R} randomly chooses one of the two public keys to generate the ring signature and transmits the ring signature $\sigma = (m, L, sk_u)$ to the adversary \mathcal{A} .

Step 4: The adversary \mathcal{A} performs polynomially restricted ring signature queries adaptively.

Step 5: Finally, the adversary \mathcal{A} gives a public key $pk' \in \{0, 1\}$.

Step 6: The adversary \mathcal{A} succeeds in this game if and only if $pk = pk'$.

The output signature cannot be seen by any third party until the signer has voluntarily disclosed all information himself/herself. In the ring signature generation, the signer computes the Y_i and A_i values needed to obtain c_i and d_i by randomly picking the corresponding $t_i, v_i, r_i \in Z_q^*$ and also obtains the private key by randomly choosing $x_i \in Z_q^*$ and computing $sk_i = H_2(x_i \cdot D_i) \in Z_q^*$. Therefore, the ring signature σ is uniformly distributed in G . The chance that a non-member can guess the real signer is at most $1/(n+1)$, and the chance that a member of the ring group can guess the real signer is at most $1/n$, so the ring signature scheme meets unconditional anonymity. \square

6. Performance Evaluation

In this section, the computational efficiency of the secure ring signature scheme based on distributed key generation proposed in this paper is analyzed. To achieve a credible security level, we adopted the experiment that has been performed for the computation evaluation in [25]. The experimental environment was: an i58500CPU@3.00GHz, 8GBRAM on an HP desktop, based on the Windows 10 operating system, under the Eclipse development environment, using JAVA Version 1.8.0 and JPBC Version 2.0.0 for the implementation, which uses the library Type A class curves to construct symmetric prime-order bilinear groups and performs Type A pairing on the super-singular elliptic curve E . The equation $y^2 \equiv x^3 + x \pmod{p}$ defines E , where $p \equiv 3 \pmod{4}$, the embedding degree is two, and the order of G_1 is q . The order of the group is 512 bit, and the order of the Galawa domain is 160 bit. The signatures of our scheme in the cryptographic operations can be found in Table 2. We define the execution time of some notations of the cryptography operations in milliseconds (ms) in Table 3.

Table 2. Notations of cryptography operation.

Notation	Crypto-Operation
SM_E	ECC-based scalar multiplication operation.
A_E	ECC-based point addition operation.
H_P	Map-to-point operation.
H	One-way hash function operation, which is negligible.
M	Multiplication operation.
P	Bilinear pair operation.
E	Exponential calculation time.

Table 3. Cryptography operations' time in milliseconds.

Cryptography Operation	T_{SM_E}	T_{A_E}	T_{H_P}	T_M	T_P	T_E
Execution time (ms)	1.7090	0.0075	4.406	0.042	5.071	8.31

In our proposed scheme, the user signature requires the ECC-based scalar multiplication operation, a hash operation mapping to points on elliptic curves, a multiplication operation, and a one-way hash operation mapping to a finite field of prime numbers, with the last one having negligible computational overhead. Therefore, the computational overhead of generating a signature is:

$$T_{Sign} = (4n - 2)T_{SM_E} + (2n - 1)T_{H_P} + T_M$$

The signature verification process of this program mainly requires the ECC-based scalar multiplication operation. Thus, the computational cost is:

$$T_{Verify} = 4nT_{SM_E}$$

The signature communication cost is $(2n + 1)L$, where L is the bit length of the group G_1 , based on the signature-generation phase of our scheme. This shows that the signature length increases linearly with the number of users.

To conclude, the time consumption of the three different ring signature schemes in the signature-generation and signature-verification steps is summarized in Table 4. It was observed that our scheme took less time than the other schemes, both in the signature-generation phase and the signature-verification phase. This indicates that our signature scheme possesses higher signing and verification efficiency.

Table 4. Efficiency analysis of ring signature algorithms.

Algorithm	Signature Generation	Signature Verification
[27]	$6nT_P + 4nT_E$	$nT_E + 2nT_P$
[28]	$(2n - 1)T_M + 4nT_E$	$2nT_E + 2TP$
[25]	$(4n - 1)T_M + (4n + 6)T_E$	$nT_E + 2T_P$
Ours	$(4n - 2)T_{SM_E} + (2n - 1)T_{H_P} + T_M$	$4nT_{SM_E}$

7. Conclusions

To address the privacy leakage problem faced by users in blockchain systems, this paper proposes a ring signature scheme suitable for blockchain. The design of the scheme is based on the elliptic curve cryptography and distributed key generation ideas. First of all, by generating the system master key through the DKG, the risk of key leakage when a trusted authorizer (TA) is attacked can be effectively reduced. Furthermore, the security analysis of the ring signature showed that the scheme enhances the unforgeability and anonymity of the signature. Under the same security parameter length, the elliptic-curve-based ring signature design is more secure than the traditional bilinear pairing design. Finally, by comparing with related signature schemes, our scheme had a shorter signature generation and verification time and higher efficiency. In view of the problem of the efficiency and communication overhead of the scheme increasing with the increase of the users, we will study the aggregation scheme of the ring signature in the future to improve the verification and communication efficiency.

Author Contributions: Conceptualization, C.P. and W.T.; Writing—original draft, L.W.; Supervision, C.P. and W.T.; Project administration, C.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the following projects: The National Key Research and Development Program of China (No. 2022YFB2701400), the National Natural Science Foundation of China (No. 62272124, No. 62361010), the Research Project of Guizhou University for Talent Introduction (No. [2020]61), the Cultivation Project of Guizhou University (No. [2019]56), and the Open Fund of Key Laboratory of Advanced Manufacturing Technology, Ministry of Education (GZUAMT2021KF[01]).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. *Decentralized Business Review*. 2008; p. 21260. Available online: <https://bitcoin.org/en/bitcoin-paper> (accessed on 5 August 2023).
2. Kano, Y.; Nakajima, T. A novel approach to solve a mining work centralization problem in blockchain technologies. *Int. J. Pervasive Comput. Commun.* **2018**, *14*, 15–32. [CrossRef]
3. Buterin, V. A next-generation smart contract and decentralized application platform. *White Pap.* **2014**, *3*, 1–2.
4. Schollmeier, R. A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications. In *Proceedings of the First International Conference on Peer-to-Peer Computing*, Linköping, Sweden, 27–29 August 2001; pp. 101–102.
5. DeGroot, M.H. Reaching a consensus. *J. Am. Stat. Assoc.* **1974**, *69*, 118–121. [CrossRef]
6. Yuan, Y.; Wang, F.Y. Blockchain: The state of the art and future trends. *Acta Autom. Sin.* **2016**, *42*, 481–494.
7. Burniske, C.; White, A. Bitcoin: Ringing the Bell for a New Asset Class. *Ark Invest* (January 2017). 2017. Available online: https://research.ark-invest.com/hubfs/1_Download_Files_ARK-Invest/White_Papers/Bitcoin-Ringing-The-Bell-For-A-New-Asset-Class.pdf (accessed on 5 August 2023).
8. Agbo, C.C.; Mahmoud, Q.H.; Eklund, J.M. Blockchain technology in healthcare: A systematic review. *Healthcare* **2019**, *7*, 56. [CrossRef] [PubMed]

9. Wang, Q.; Su, M. Integrating blockchain technology into the energy sector—from theory of blockchain to research and application of energy blockchain. *Comput. Sci. Rev.* **2020**, *37*, 100275. [[CrossRef](#)]
10. Ron, D.; Shamir, A. Quantitative analysis of the full bitcoin transaction graph. In Proceedings of the Financial Cryptography and Data Security: 17th International Conference, FC 2013, Okinawa, Japan, 1–5 April 2013; pp. 6–24.
11. Fleder, M.; Kester, M.S.; Pillai, S. Bitcoin transaction graph analysis. *arXiv* **2015**, arXiv:1502.01657.
12. Rivest, R.L.; Shamir, A.; Tauman, Y. How to leak a secret. In Proceedings of the Advances in Cryptology—ASIACRYPT 2001: 7th International Conference on the Theory and Application of Cryptology and Information Security, Gold Coast, Australia, 9–13 December 2001; pp. 552–565.
13. Chaum, D.; Van Heyst, E. Group signatures. In Proceedings of the Advances in Cryptology—EUROCRYPT’91: Workshop on the Theory and Application of Cryptographic Techniques, Brighton, UK, 8–11 April 1991; pp. 257–265.
14. Shamir, A. How to share a secret. *Commun. ACM* **1979**, *22*, 612–613. [[CrossRef](#)]
15. Feldman, P. A practical scheme for non-interactive verifiable secret sharing. In Proceedings of the 28th Annual Symposium on Foundations of Computer Science, Washington, DC, USA, 12–14 October 1987; pp. 427–438.
16. Bonneau, J.; Narayanan, A.; Miller, A.; Clark, J.; Kroll, J.A.; Felten, E.W. Mixcoin: Anonymity for bitcoin with accountable mixes. In Proceedings of the Financial Cryptography and Data Security: 18th International Conference, FC 2014, Christ Church, Barbados, 3–7 March 2014; pp. 486–504.
17. Valenta, L.; Rowan, B. Blindcoin: Blinded, accountable mixes for bitcoin. In Proceedings of the Financial Cryptography and Data Security: FC 2015 International Workshops, BITCOIN, WAHC, and Wearable, San Juan, Puerto Rico, 30 January 2015; pp. 112–126.
18. Ziegeldorf, J.H.; Matzutt, R.; Henze, M.; Grossmann, F.; Wehrle, K. Secure and anonymous decentralized Bitcoin mixing. *Future Gener. Comput. Syst.* **2018**, *80*, 448–466. [[CrossRef](#)]
19. Goldwasser, S.; Micali, S.; Rackoff, C. The knowledge complexity of interactive proof-systems. In *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*; ACM: New York, NY, USA, 2019; pp. 203–225.
20. Li, B.; Wang, Y. RZKPB: A privacy-preserving blockchain-based fair transaction method for sharing economy. In Proceedings of the 2018 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/12th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE), New York, NY, USA, 1–3 August 2018; pp. 1164–1169.
21. Haines, T.; Boyen, X. Votor: Conceptually simple remote voting against tiny tyrants. In Proceedings of the Australasian Computer Science Week Multiconference, Canberra, Australia, 2–5 February 2016; pp. 1–13.
22. Patil, K.; Wasnik, C.T. An ID-based block ring signature system for secret sharing of data. In Proceedings of the 2017 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 5–7 January 2017; pp. 1–5.
23. Li, X.; Mei, Y.; Gong, J.; Xiang, F.; Sun, Z. A blockchain privacy protection scheme based on ring signature. *IEEE Access* **2020**, *8*, 76765–76772. [[CrossRef](#)]
24. Wang, Z.; Fan, J. Flexible threshold ring signature in chronological order for privacy protection in edge computing. *IEEE Trans. Cloud Comput.* **2020**, *10*, 1253–1261. [[CrossRef](#)]
25. Lai, C.; Ma, Z.; Guo, R.; Zheng, D. Secure medical data sharing scheme based on traceable ring signature and blockchain. *Peer-Netw. Appl.* **2022**, *15*, 1562–1576. [[CrossRef](#)]
26. Samra, B.; Fouzi, S. New efficient certificateless scheme-based conditional privacy preservation authentication for applications in VANET. *Veh. Commun.* **2022**, *34*, 100414. [[CrossRef](#)]
27. Cheng, X.; Guo, R.; Cheng, Y. Construction of efficient ring signature scheme with revocation of anonymity. *Commun. Eng. Des. Mag.* **2015**, *36*, 857–861.
28. Mao, M.; Zhou, Z.X. A forward-secure anonymity signature scheme based on ring signature idea. *Microcomput. Inf.* **2010**, *26*, 62–63.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.