Blockchain-based Outsourced Secure Multi-party Privacy-preserving Verifiable Computation Scheme

Feng Xi[1]

[1]*Yunnan University of Finance and Economics, Information Center, South Campus, Yunnan University of Finance and Economics, Wuhua District, Kunming City, Yunnan Province, China*

Keywords
Blockchain; Outsourcing Computing; Secure Multi-party; Private Set Intersection; Verifiable Computing

Correspondence Email:
202302110719@stu.ynufe.edu.cn

Abstract
With the increasing demand for data privacy protection, outsourcing secure multi-party privacy intersection is faced with the dual challenges of privacy leakage and computational credibility verification. This paper proposes a blockchain-based secure multi-party privacy intersection verifiable computing scheme, which achieves the synergistic enhancement of privacy protection and computational credibility by integrating Paillier encryption algorithm, DH protocol key negotiation, HMAC-RSA authentication, Shamir secret sharing, AES encryption and zero-knowledge proof. In view of the core requirements of verifiable computing, zero-knowledge proof technology is introduced and verification logic is automatically executed through smart contracts to ensure that the calculation results are accurate and cannot be tampered with. In view of the credibility of computing nodes, a dynamic reward and punishment mechanism based on trust function and utility function is designed to constrain the honest behavior of computing parties through economic incentives. The experimental results verify the effectiveness of the scheme.

1. Introduction

In the big data era, data has become a vital asset, making privacy protection paramount. While organizations and individuals enjoy the convenience of data, they also face significant risks of privacy breaches. Outsourced secure multi-party privacy intersection computation, a key data processing method, allows multiple participants to collaboratively compute intersection results without disclosing their private data. This technology has extensive and urgent demands across fields like healthcare, finance, and social media [2][5].

However, current outsourced secure multi-party privacy intersection computation faces dual challenges: on one hand, privacy leakage concerns where participants fear their data might be accessed during computations; on the other hand, the difficulty in verifying computational result credibility, as participants struggle to confirm that results haven't been maliciously tampered with.

This paper proposes a blockchain-based secure multi-party computation scheme for outsourced privacy-preserving intersection queries. The innovative approach integrates blockchain technology with other privacy protection techniques, leveraging its tamper-proof and traceable characteristics to provide verifiable records for computational processes. To ensure privacy protection, the scheme combines encryption algorithms and secret sharing techniques to maintain data confidentiality

throughout computations. For computational reliability, smart contracts automate verification procedures while implementing dynamic incentive mechanisms to effectively prevent malicious actions from compromising results. This provides a more secure, trustworthy, and efficient solution for outsourced secure multi-party privacy-preserving intersection queries, promoting the widespread adoption of this technology in practical applications [3][10].

1.1 Literature Review

The academic community has conducted multi-dimensional research on the challenge of ensuring the credibility and verifiability of outsourced computing results while protecting data privacy. Three main technical approaches have emerged: First, verifiable computing schemes based on traditional cryptography. These studies primarily rely on advanced cryptographic tools such as zero-knowledge proofs (Zero-Knowledge Proof) and polynomial commitments (e.g., KZG commitments) to enable computing nodes to prove the correctness of their computing processes to task initiators without revealing the original data. Second, decentralized trust architecture based on blockchain. Blockchain provides a natural technical foundation for building trusted computing environments with its decentralized, tamper-proof, and traceable characteristics. Existing research has attempted to combine MPC with blockchain, using smart contracts to automatically execute computing tasks and result verification, thereby eliminating the need for centralized authorities. Third, integrated frameworks that combine privacy protection and access control. To further strengthen privacy protection, some studies have integrated technologies such as homomorphic encryption (e.g., Paillier), secret sharing (e.g., Shamir secret sharing), and message authentication codes (HMAC) into the MPC framework to achieve multi-level data protection. In summary, although existing research has made significant progress in privacy protection and verifiable computing, three core contradictions remain to be reconciled: the contradiction between privacy protection and computational efficiency; the contradiction between result verifiability and verification cost; and the contradiction between decentralized architecture and node behavior incentives. Most current solutions tend to focus on optimizing a single dimension, making it difficult to achieve among the three.

Against this background, this paper proposes a "Blockchain-based Secure Multi-party Private Set Intersection Verifiable Computing Scheme." This proposal is not a simple combination of existing technologies but a systematic innovation oriented towards practical application scenarios. Its core research motivation stems from a profound insight into the aforementioned research gaps: In terms of technology integration, the scheme innovatively integrates Paillier encryption, DH key agreement, HMAC-RSA authentication, Shamir secret sharing, AES encryption, and zero-knowledge proofs to build a depth defense privacy protection system; In terms of verification mechanisms, it abandons the traditional model relying on centralized verifiers and instead uses smart contracts to automatically execute zero-knowledge verification logic based on KZG polynomial commitments, achieving complete decentralization and automation of the verification process.

2. Research Methods

The overall architecture of the system is illustrated in Fig. 1. In addition to data, the system comprises three roles: PSI Task Initiator, Computing Node, and Verification Node. PSI Task Initiator: As the

service recipient, the PSI Task Initiator utilizes secure computing services provided by blockchain nodes while paying transaction fees and deposits. They transmit data through confidential sharing mechanisms to service provider nodes and supply verified data to the Verification Node. The participant pool must consist of at least two members who pre-process and encrypt data to ensure security. Computing Node: This node participates in PSI data computation by retrieving task parameters from the blockchain (e.g., participant public key sets) and collaborating with the Verification Node to validate smart contracts regarding data integrity and computational accuracy. Verification Node: The Verification Node reviews the PSI Task Initiator's computational tasks, verifying whether transaction fees and deposits meet requirements. It determines which computing nodes participate in the computation, selects specific nodes for execution, and participates in incentive contracts based on final validation results to decide on rewards or penalties for computing nodes.

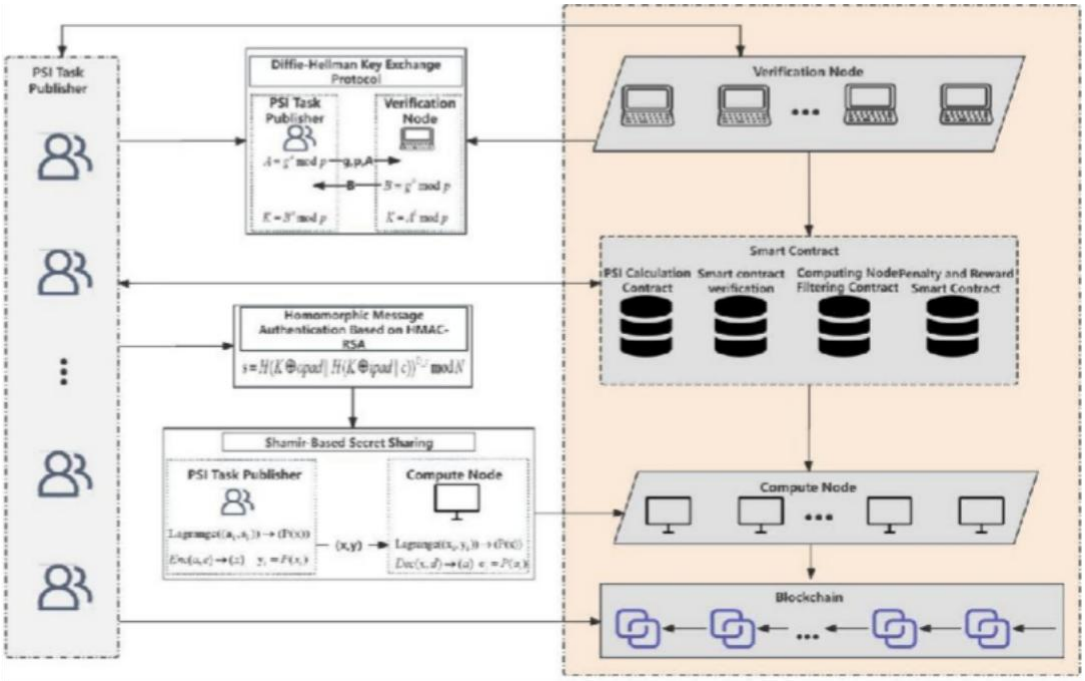

Fig. 1 General Structure Diagram

2.1 Data Preprocessing

In the PSI task, each initiator (Initiatori) selects two distinct large primes p, q and defines $N = p \cdot q$. They then choose two integers E and Di satisfying: $\gcd(E, \varphi(N)) = 1$, $E \cdot Di = 1 \bmod \varphi(N)$, $g = N + 1$, $\lambda = \varphi(N)$, $\mu = (\varphi(N) - 1) \bmod N$. This generates a public key pair (N, g, E) and a private key pair (λ, μ, Di). The public keys N and E are shared by all participants. Each participant encrypts their data using the generated public key pair (N, g) through the Paillier encryption algorithm to obtain the corresponding ciphertext c.

The initiator and verifier of the task generate a shared key K through key negotiation based on the Diffie-Hellman (DH) protocol. Using this shared key, they compute the HMAC value of the ciphertext and encrypt it with the private key Di via RSA to obtain the ciphertext s.

2.2 Shared Secret Phase

The PSI task initiator (Initiatori) selects a set of n+1 random numbers a = (a0, a1, ..., an) and the participant's encrypted data to form n+1 points: (a0, s0), (a1, s1), ..., (an, sn). Using the Lagrange interpolation formula, a polynomial P(x) is constructed: $P(x) = \Sigma(i=0 \text{ to } n)\ si \cdot \Pi(j=0, j\neq i \text{ to } n)\ (x - aj) / (ai - aj)$. The random number is encrypted using the shared public key of the computing node to generate a unique identifier. This identifier is substituted into the polynomial to obtain the corresponding y-value, which forms the secret-sharing share sent to the computing node. A random integer t (where $1 < t < n$) is selected, and t random plaintexts s′ are obtained by summing the corresponding ai values. The task initiator and verifier share a key K, which is used to encrypt the bi = AES_E(ai, K), S = AES_E(s′, K) and xi = ai mod N values through AES encryption before sending them to the verification node. The polynomial P(x) can be reconstructed when at least n+1 computing nodes provide their received shares.

The computational node selects two distinct large primes p, q, defining $N = p \cdot q$. It generates a shared public key e and private keys d_i satisfying: $\gcd(e, \varphi(N)) = 1$, $e \cdot d_i = 1 \bmod \varphi(N)$. Using the private key, the node decrypts ai = xi^d_i mod n the random number selected by the PSI task initiator, Initiatori, and substitutes it into the polynomial to obtain the correct ciphertext $ci = \Sigma(i=0 \text{ to } n)\ si \cdot \Pi(j=0, j\neq i \text{ to } n)\ (ai - aj) / (ai - aj)$.

2.3 Verification Phase

Verifiable computation is implemented through KZG polynomial commitments. Let b denote the maximum degree bound of the polynomial P(x) to be supported, and let SRS = (g, g^s, ..., g^(s^b)) be the commitment. The prover constructs the polynomial $F(x) = \Pi(i=0 \text{ to } n)\ (x - ci)$, $P(x) = \Pi(i=0 \text{ to } n)\ (x - psi\_result\_i)$, $Q(x) = P(x) / F(x)$ using the intersection result psi_result_i. The prover then reconstructs the polynomial using the secret [s_i] to form the commitment KZG([s_i], P(x), Q(x), F(x))
→ (CP, CQ, CF). These commitments CP, CF, CQ are sent to the verifier, who selects k points (z0, ..., z_{k−1}) as the challenge and transmits them to the prover.

The proving party evaluates the polynomial at k points to obtain the corresponding y-values: yPi = P(zi), yFi = F(zi), yQi = Q(zi), then generates polynomials IP(x), IF(x), IQ(x) using the corresponding point sets. Define a zero polynomial $Z(x) = \Pi(i=0 \text{ to } k-1)\ (x - zi)$, and compute: $p(x) = (P(x) - IP(x)) / Z(x)$, $f(x) = (F(x) - IF(x)) / Z(x)$, $q(x) = (Q(x) - IQ(x)) / Z(x)$. IP(x), IF(x), IQ(x) are constructed using Lagrange interpolation: $\Sigma(i=0 \text{ to } k-1)\ y\_i \cdot \Pi(j=0, j\neq i \text{ to } k-1)\ (x - zj) / (zi - zj)$.

The proving party then defines three Kate multiproofs for the evaluation of these points: (zi, yPi): πP = [p(s)]1, (zi, yFi): πF = [f(s)]1, and (zi, yQi): πQ = [q(s)]1, sending πP, πF, πQ and k pairs of (z, y) points to the verifier. The verifier uses k points to construct Z(x), IP(x), IF(x), IQ(x), and verifies the following equations: e(πP, [Z(s)]2) = e(CP − [IP(s)]1, h), e(πF, [Z(s)]2) = e(CF − [IF(s)]1, h), e(πQ, [Z(s)]2) = e(g, CQ − [IQ(s)]2). If the equations hold, the verifier accepts the commitment and proceeds to the next step; otherwise, the verifier rejects the commitment and returns "verification failed". The verifier sends zi to the prover, receives the returned computation result yi = p(zi), and verifies the subset relationship and data consistency.

The verification is completed by checking: ComResult = Σ(i=0 to n) (AES_D(bi, K) ⊕cp || AES_D(S, K) ⊕cp || ...). In this solution, the verification process will be implemented through smart contracts.

3. Result and Discussion

3.1 Experimental Design

This study proposes a blockchain-based outsourced secure multi-party privacy-preserving verifiable computation scheme (SMP-PSI). To systematically evaluate the scheme's performance across different data scales, the experiment employs three dataset configurations: 5x, 7x, and 10x sizes of the intersection dataset. By dynamically adjusting the intersection data points (10^4 to 1.1×10^5), we conduct systematic tests on the time complexity of the computational nodes (provers) and verification nodes (verifiers). The experimental framework utilizes the KZG polynomial commitment mechanism and smart contract technology, focusing on efficiency differences between the proof and verification phases. Notably, the verification nodes automate the execution of verification logic through smart contracts.

3.2 Performance Comparison

The experiment was repeated 10 times to eliminate outliers, with the results shown in Fig. 6. The left panel displays Proof Time Consumption, while the right panel shows Verification Time Consumption.

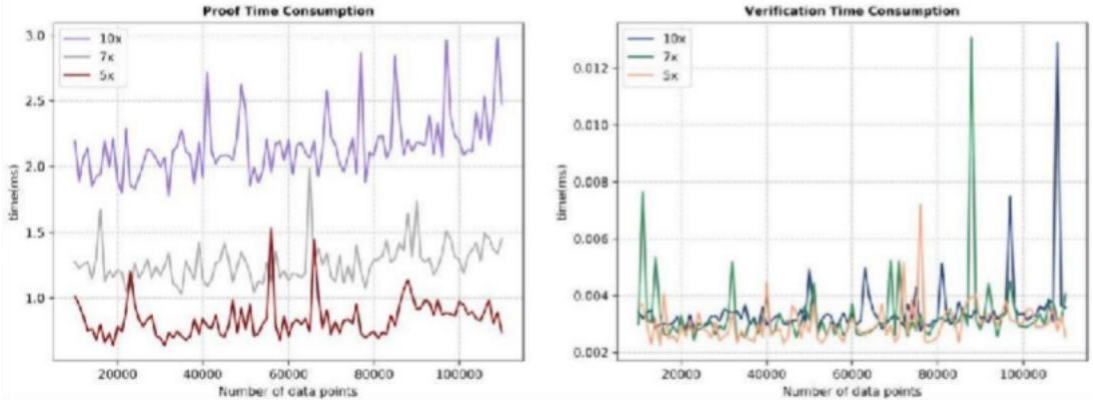

Fig. 6 Experimental Results

3.3 Interpretation of Results

The left figure demonstrates that proof time exhibits significant fluctuations with increasing data volume. Higher data multiples result in longer proof times and greater volatility. Specifically, when data volume is 10 times the original, proof time ranges from 1.72 to 2.98 milliseconds; at 7 times, it fluctuates between 1.03 and 1.99 milliseconds; while at 5 times, it remains relatively stable between
0.64 and 1.53 milliseconds. The proof process requires constructing polynomials and generating KZG commitments. As data volume increases, the computational complexity of polynomial operations and ciphertext processing grow linearly, leading to intensified resource consumption. The high volatility may stem from resource contention among computing nodes or random anomalies. The verification time shown in the right figure is significantly shorter than the proof time and remains insensitive to variations in data volume. Under three scenarios (10x, 7x, and 5x), the verification time consistently

stays within the range of 0.002 to 0.013 milliseconds, demonstrating minimal fluctuations. Overall, the prover's time grows approximately linearly with increasing set size, while the verifier's time remains stable across different scales. The verification process leverages the efficiency of KZG polynomial commitments through elliptic curve pairing operations, where complexity depends solely on the number of verification points rather than data volume. Experimental results not only validate the scheme's verification efficiency but also reveal performance differences between systems with varying set sizes.

4. Conclusions

This study addresses the dual technical challenges of data privacy protection and computational result verification in outsourced secure multi-party private set intersection (PSI) scenarios. We propose a blockchain-based framework for verifiable PSI computation that establishes a distributed trust mechanism to ensure tamper-proof verification records. The framework integrates multiple cryptographic techniques including Paillier homomorphic encryption, Diffie-Hellman key negotiation, HMAC-RSA message authentication, Shamir secret sharing, and AES encryption to build a comprehensive privacy protection system. By combining zero-knowledge proofs with smart contracts, the framework enables automated verification of computational results, ensuring the accuracy and integrity of PSI outcomes.

Future research will focus on exploring the following directions: the implementation mechanism of verifiable computation processes, performance optimization strategies under large-scale datasets and complex network environments, as well as integration and application paths with emerging privacy computing technologies such as federated learning.

5. References

[1] Kate, A. P. (2010). Distributed Key Generation and its applications. In UWSpace (University of Waterloo). http://hdl.handle.net/10012/5285

[2] Groth, J. (2016). On the Size of Pairing-Based Non-interactive Arguments. In Lecture notes in computer science (pp. 305–326). https://doi.org/10.1007/978-3-662-49896-5_11

[3] Li, P., Xu, H., S Ji, Y. (2016). Multi-client outsourced computation. In Lecture notes in computer science (pp. 397–409). https://doi.org/10.1007/978-3-319-38898-4_23

[4] Yao, A. C. (1982). Protocols for secure computations. Foundations of Computer Science, 160–164. https://doi.org/10.1109/sfcs.1982.88

[5] Goldreich, O., Micali, S., S Wigderson, A. (2019). How to play any mental game, or a completeness theorem for protocols with honest majority. In Association for Computing Machinery eBooks. https://doi.org/10.1145/3335741.3335755

[6] Tong Wei, Dong Xuewen, Shen Yulong, Zhang Yuanyu, Jiang Xiaohong S Tian Wensheng. (2022). CHChain: Secure and parallel crowdsourcing driven by hybrid blockchain. Future Generation Computer Systems. https://doi.org/10.1016/J.FUTURE.2022.01.023.

[7] Xue, K.-P., Fan, M., Wang, F., S Luo, X.-Y. (2024). Data verification and controllable anonymity scheme in blockchain privacy crowdsourcing. Journal of Electronics and Information Technology, 46(2), 748–756.

[8] Zhang Chen, Guo Yu, Jia Xiaohua, Wang Cong S Du Hongwei. (2021). Enabling Proxy-Free Privacy-Preserving and Federated Crowdsourcing by Using Blockchain. IEEE Internet of Things Journal, 8(8), 6624–6636. https://doi.org/10.1109/JIOT.2021.3051295.

[9] Guo, Z., Zhang, Y., An, F.-L., Zhao, K.-J., Zhang, W.-J., S Ye, J. (2021). Blockchain-based polynomial outsourcing computation scheme with privacy protection. Journal of Information Security, 6(1), 78–89.
[10] Feng, T., Kong, F.-Q., Liu, C.-Y., Ma, R., S Albettar, M. (2021). A blockchain-based dual-verifiable cloud storage scheme. Journal on Communications, 42(12), 192–201.
[11] Zhuang, C.-Y., Guo, R., S Yang, G. (2022). Verifiable outsourced decryption with anonymity attribute-based encryption scheme in blockchain. Computer Engineering and Applications, 58(19), 124–134.
[12] Wang, T., Ma, W.-P., S Luo, W. (2019). A blockchain-based information sharing and secure multi-party computation model. Computer Science, 46(9), 162–168.
[13] Xia, H., Tian, W., Gao, J.-B., Zhang, T.-Y., Gao, R., S Xia, Q. (2024). A verifiable privacy-preserving scheme for outsourced secure multi-party statistical computation based on blockchain. Radio Engineering, 54(4), 835–847.