VOTING USING HOMOMORPHIC ENCRYPTION VS. VOTING USING MULTI-PARTY COMPUTATION

Artur Jakubski [1]***, Jacek Piątkowski*** [2]***, Robert Perliński*** [3]

[1,2,3]*Department of Computer Science, Czestochowa University of Technology, Czestochowa, Poland*

[1]*artur.jakubski@icis.pcz.pl@icis.pcz.pl,* [2]*jacek.piatkowski@icis.pcz.pl,* [3]*robert.perlinski@icis.pcz.pl*

Keywords: **Homomorphic Encryption (HE), Multi-Party Computation (MPC), e-voting, vote tallying, secret sharing, threshold cryptography.**

Modern electronic voting systems require ensuring both the correctness of results and strong privacy guarantees. Traditional cryptographic approaches often require decrypting data before processing it, which creates potential attack vectors.

In response to these limitations, two main paradigms have been developed: homomorphic encryption (HE) and secure multi-party computation (MPC). Both approaches enable processing encrypted or otherwise hidden data without revealing its content; however, they differ in their operational models, computational costs, and scalability characteristics.

Homomorphic Encryption in Voting Systems

Homomorphic encryption allows algebraic operations to be performed directly on encrypted data. In the context of voting, this means that votes can be encrypted by voters and then aggregated without being decrypted. In particular, partially homomorphic schemes (e.g., additive ones) enable summing votes, which corresponds to the fundamental tallying operation. From an efficiency perspective, HE offers several important advantages. These include the lack of need for interaction between multiple parties during computation, a relatively simple implementation model, and, in some cases, the possibility of using a central server for data aggregation. However, the computational cost of operations on ciphertexts is significantly higher than on plaintext data. Operations such as multiplication or more complex logical functions (e.g., comparisons, rankings) in fully homomorphic encryption (FHE) schemes are particularly expensive. Moreover, ciphertext sizes are much larger than the original input data, which increases transmission and storage costs.

In practical voting systems, HE performs best in scenarios where operations are limited to summation or simple linear functions. In such cases, it is possible to achieve relatively good scalability, even for large populations of voters.

Secure Multi-Party Computation (MPC)

MPC is based on the collaboration of multiple parties that jointly compute a function over their inputs without revealing those inputs to each other. Data is typically divided into shares (secret sharing), which are distributed among protocol participants. Computations are performed on these shares, and the final result is reconstructed only at the end of the process. In the context of voting, MPC enables the implementation of more complex procedures than HE, including verification of vote correctness and process integrity. The efficiency of MPC strongly depends on the communication model and the number of participants. Unlike HE, MPC requires multiple rounds of communication between parties, which can lead to significant delays in high-latency environments. Communication cost often dominates computational cost.

On the other hand, MPC offers better performance for non-linear operations compared to FHE. Many MPC protocols are optimized for specific operations (e.g., comparisons, multiplications), making them more practical in complex voting scenarios.

Efficiency Comparison

When comparing HE and MPC in terms of efficiency, several key dimensions should be considered: computational cost, communication cost, scalability, function complexity, as well as security and trust model. HE is characterized by a high cost of operations on individual data items, especially in the case of FHE. In contrast, MPC distributes the computational workload among participants and often achieves better execution times for complex functions. HE requires minimal communication—votes are transmitted and aggregated in a largely one-directional manner. MPC requires intensive multi-directional communication, which can become a bottleneck. HE scales well with the number of voters, as each vote can be processed independently. MPC scales less efficiently as the number of protocol participants increases, particularly when each voter is an active participant in the computation. MPC outperforms HE in the case of complex voting algorithms, while HE is more efficient for simple aggregations. HE typically relies on a single public key and requires trust in the decryption process (or the use of threshold decryption techniques). MPC eliminates a single point of trust but requires assumptions about the honesty of a majority of participants.

Conclusions

There is no single universally optimal solution for all voting scenarios. Homomorphic encryption is more efficient in simple voting systems where summation operations dominate and minimizing communication is crucial. MPC, on the other hand, is better suited for complex systems requiring rich computational logic and a high degree of decentralization.

In practice, hybrid approaches combining HE and MPC are increasingly being considered in order to leverage the advantages of both techniques. For example, HE can be used for vote aggregation, while MPC can be applied for verification or for implementing more advanced voting procedures.

Our research focuses on optimizing the performance of both approaches, particularly by reducing communication costs in MPC and accelerating operations in FHE. Advances in hardware, its effective utilization, and the development of new protocols may significantly reshape the current efficiency landscape of these technologies in the future.

References

[1] Jakubski A, Piątkowski J, Perliński R., Efficient vote encoding and counting protocols based on the Chinese remainder theorem and multiparty computation, 16th Conference on Mathematical Modeling in Physics and Engineering, 2025.

[2] Yuan, K., Sang, P., Zhang, S., Chen, X., Yang, W., & Jia, C. (2023). An electronic voting scheme based on homomorphic encryption and decentralization. PeerJ Computer Science, 9, e1649. https://doi.org/10.7717/peerj-cs.1649.

[3] Paillier, P. (1999). Public-key cryptosystems based on composite degree residuosity classes. In Stern, J., editor, Advances in Cryptology - EUROCRYPT ’99, pp. 223–238, Berlin, Heidelberg. Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-48910-X_16.

[4] Li, J., Wang, X., Huang, Z., Wang, L., & Xiang, Y. (2019). Multi-level multi-secret sharing scheme for decentralized e-voting in cloud computing. Journal of Parallel and Distributed Computing, 130:91–97. https://doi.org/10.1016/j.jpdc.2019.04.003

[5] Fan, X., Wu, T., Zheng, Q., Chen, Y., Alam, M., & Xiao, X. (2020). Hse-voting: A secure high-efficiency electronic voting scheme based on homomorphic signcryption. Future Generation Computer Systems, 111:754–762. https://doi.org/10.1016/j.future.2019.10.016

[6] Damgard, I., Pastro, V., Smart, N., & Zakarias, S. (2012). Multiparty computation from somewhat homomorphic encryption. In Annual Cryptology Conference (pp. 643-662). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-32009-5_38

[7] Cortier, V., Gaudry, P., & Yang, Q. (2022). A toolbox for verifiable tally-hiding e-voting systems. In European Symposium on Research in Computer Security, (pp. 631–652). Springer. https://doi.org/10.1007/978-3-031-17146-8_31