**Overview**

**Introduction:**

The Idea Concept Paper will identify and focus a research direction in the field of Asynchronous Consensus and Permissionless Systems. Specifically, the research will incorporate the Aleph protocol (Gągol et al., 2019) as the basis of the consensus protocol targeted for improvement to enhance its efficiency and scalability by reducing communication complexity. The paper will address a performance problem with Alephs communication complexity and propose an improvement based on the evaluation and integration of published improved Reliable Broadcast Communications (RBC), replacing Aleph’s merkle tree based RBC with one based on RSA accumulators (Hussein & Al-Gailani, 2022).

The Aleph protocol is important because it is credited to being one of the first asynchronous consensus protocols to operate in a permissionless setting by removing the need of Distributed Key Generation (DKG) and operating without a trusted dealer (Guo et al., 2022). However, performance is throttled by the use of merkle tree based RBC, which becomes resource intensive as the network scales. This is because merkle trees provide cryptographic proofs of data integrity, but require significant computational resources for construction and verification leading to increased latency and reduced throughput in blockchain systems. (Hussein & Al-Gailani, 2022).

To address these issues, this research proposes replacing Aleph's merkle tree based RBC with a more efficient system based on RSA accumulators. RSA accumulators offer a cryptographic alternative that can significantly reduce communication and computational complexity (Reddy, 2021). By integrating RSA accumulator based RBC, the protocol aims to decrease communication complexity, leading to an enhanced performance.

**Problem:** *Aleph Scalability Communication Complexity*

The Aleph research focuses on the theoretical aspects of the protocol rather than empirical evaluation. The research does not detail how workloads are generated and does not mention the collection of raw data from simulations, but Aleph however does provide mathematical proofs of the algorithmic properties. In the mathematical proof of Lemma F.1 Aleph provides the scalability of communication complexity problem.

The communication complexity problem is based upon the evidence of the F.1 Lemma that provides a mathematical proof demonstrating how the Aleph implementation of RBC has high message overhead, exacerbating its communication complexity. Alephs implementation of RBC is called Chain Reliable Broadcast (ch-RBC) and has a message complexity that is defined in the lemma as O(Tr+(N⌃2)log⁡N)). The (Tr) portion represents the total number of transaction inputs (T) in honest units of round (r). Independent of the transaction input rounds (Tr) is the communication overhead ((N⌃2)log⁡N) that this research will focus on. The communication overhead has four parts. The first three are Propose, Prevote, and Commit which all grow quadratically O(N⌃2) because each node communicates with every other node (Gągol et al., 2019). The 4th part is the merkle-tree validation phase and it grows logarithmically O(logN). Each node needs to verify the integrity of the shares using merkle tree branches, and this verification involves O(logN) operations due to the properties of merkle trees (Kharangate, 2023). The problem becomes apparent in large networks such as when N=1000, the communication complexity of O(N⌃2log⁡N) implies that approximately 9.97 million operations or message exchanges are needed. That is a problem in comparison to the first synchronous permissionless ledger Bitcoin (Nakamoto, 2008) and the first asynchronous permissioned ledger Honey Badger BFT (Miller et al., 2016). Unlike BTC’s linear communication complexity O(N) and HBFT’s quadratic O(N^2), Aleph suffers from both quadratic and logarithmic growth in message exchanges and validations. This is a problem because large networks are needed to handle increased transaction volumes and to ensure decentralization and security, which are critical for the robustness of blockchain systems (Gencer, Basu, Eyal, van Renesse, & Sirer, 2018).

By addressing the communication complexity problem, the research enhances the scalability of the Aleph protocol, making it more resilient to network congestion and better equipped to contribute to the field of asynchronous permissionless systems.

**Goal:**

The goal of this research is to reduce the communication complexity of the Aleph protocol by implementing an improved RBC protocol using RSA accumulators instead of merkle trees and conducting comprehensive simulations with comparative analyses to obtain quantifiable metrics to support this claim.

As stated in the F.1 Lemma of Aleph, implementing RSA accumulators to replace merkle trees in the RBC protocol can significantly improve scalability. The reason being that RSA accumulators provide a cryptographic method for aggregating multiple values into a single, fixed-size accumulator (Reddy, 2021). Research has also supported that RSA accumulators allow for more efficient verification and communication processes in blockchain networks compared to the logarithmic and quadratic complexity of merkle trees (Hussein & Al-Gailani, 2022). This is important because RSA accumulators offer efficient aggregation of multiple values into fixed-size accumulators, allowing the protocol to achieve more streamlined verification and communication processes.

The simulations and comparative analyses will involve replicating similar development environments and benchmarking methodologies used in previous asynchronous consensus research like HBFT, BEAT, DUMBO, and the Asynchronous Byzantine Fault Tolerance (ABFT) protocols to ensure consistent and reliable performance data (Miller et al., 2016, Knudsen et al., 2021, Duan et al., 2018, and Guo, Lu, Tang, Xu, & Zhang, 2020).

The goal will be to obtain quantifiable metrics to support the claim of a reduction in communication complexity of Aleph by reducing the computational and communication overhead of the ch-RBC with RSA accumulators. Thereby improving the overall scalability and communication complexity of the Aleph protocol.

**References**

Abraham, I., Jovanovic, P., Maller, M., Meiklejohn, S., Stern, G., & Tomescu, A. (2021). Reaching consensus for asynchronous distributed key generation. In *Proceedings of the 2021 ACM Symposium on Principles of Distributed Computing* (pp. 363–373).

Baird, L., & Luykx, A. (2020). The Hashgraph Protocol: Efficient Asynchronous BFT for High-Throughput Distributed Ledgers. *International Conference on Omni-layer Intelligent Systems* (pp. 1-7).<https://doi.org/10.1109/COINS49042.2020.9191430>

Boldyreva, A. (2002). Threshold signatures, multisignatures and blind signatures based on the gap-diffie-hellman-group signature scheme. In *Public key cryptography–PKC 2003* (pp. 31–46). Springer.

Das, S., Xiang, Z., & Ren, L. (2020). Asynchronous data dissemination and its applications. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*.

Djari, A., Anceaume, E., & Tucci-Piergiovanni, S. (2022). An extensive agent-based simulation study of sycomore++, a DAG-based permissionless ledger. In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing* (pp. 334–336). Association for Computing Machinery.<https://doi-org.ezproxylocal.library.nova.edu/10.1145/3477314.3507245>

Duan, S., Reiter, M., & Zhang, H. (2018). Beat: Asynchronous BFT Made Practical. In *ACM SIGSAC Conference on Computer and Communications Security* (pp. 2028–2041).<https://doi.org/10.1145/3243734.3243812>

Fischer, M. J., Lynch, N. A., & Paterson, M. S. (1982). Impossibility of distributed consensus with one faulty process (Technical Report No. MIT/LCS/TR-728). Massachusetts Institute of Technology, Laboratory for Computer Science, Cambridge.

Gągol, A., Leśniak, D., Straszak, D., & Świętek, M. (2019). Aleph: Efficient atomic broadcast in asynchronous networks with Byzantine nodes. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies (AFT '19)* (pp. 214–228). Association for Computing Machinery.<https://doi.org/10.1145/3318041.3355467>

Gao, Y., Lu, Y., Lu, Z., Tang, Q., Xu, J., & Zhang, Z. (2022). Dumbo-NG: Fast Asynchronous BFT Consensus with Throughput-Oblivious Latency. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security (CCS '22)* (p. 1187–1201). Association for Computing Machinery.<https://doi.org/10.1145/3548606.3559379>

[Gencer, A. E., Basu, S., Eyal, I., van Renesse, R., & Sirer, E. G. (2018). Decentralization in Bitcoin and Ethereum Networks. In *Proceedings of the 22nd International Conference on Financial Cryptography and Data Security* (pp. 439-457). Springer. https://doi.org/10.1007/978-3-662-58387-6\_25](https://doi.org/10.1007/978-3-662-58387-6_25)

Gennaro, R., Jarecki, S., Krawczyk, H., & Rabin, T. (2003). Secure Applications of Pedersen’s Distributed Key Generation Protocol. In *Topics in Cryptology - CT-RSA 2003, The Cryptographers’ Track at the RSA Conference 2003, San Francisco, CA, USA, April 13-17, 2003, Proceedings* (pp. 373–390).<https://doi.org/10.1007/3-540-36563-X_26>

Gennaro, R., Jarecki, S., Krawczyk, H., & Rabin, T. (2007). Secure Distributed Key Generation for Discrete-Log Based Cryptosystems. *Journal of Cryptology, 20*(1), 51–83.<https://doi.org/10.1007/s00145-006-0347-3>

Guo, B., Lu, Z., Tang, Q., Xu, J., & Zhang, Z. (2020). Dumbo: Faster Asynchronous BFT Protocols. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security (CCS '20)* (pp. 803–818). Association for Computing Machinery.<https://doi.org/10.1145/3372297.3417262>

Guo, B., Lu, Y., Lu, Z., Tang, Q., Xu, J., & Zhang, Z. (2022). Speeding Dumbo: Pushing Asynchronous BFT Closer to Practice. doi:10.14722/ndss.2022.24385

Hood, K., Oglio, J., Nesterenko, M., & Sharma, G. (2021). Partitionable Asynchronous Cryptocurrency Blockchain. *IEEE International Conference on Blockchain and Cryptocurrency* (pp. 1-9).<https://doi.org/10.1109/ICBC51069.2021.9461080>

Hussein, K. M., & Al-Gailani, M. F. (2022). An Efficient Bandwidth Based on the Cryptographic Technique of the RSA Accumulator in Block Chain Networks. In 2022 Fifth College of Science International Conference of Recent Trends in Information Technology (CSCTIT) (pp. 164-168). Baghdad, Iraq. doi:10.1109/CSCTIT56299.2022.10145614

Kharangate, A. (2023). Asynchronous merkle trees. arXiv. https://ar5iv.labs.arxiv.org/html/2311.17441

Kogias, E. K., Malkhi, D., & Spiegelman, A. (2020). Asynchronous Distributed Key Generation for Computationally-Secure Randomness, Consensus, and Threshold Signatures. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security (CCS '20)* (pp. 1751–1767). Association for Computing Machinery.<https://doi.org/10.1145/3372297.3423364>

Knudsen, H., Li, J., Notland, J., Haro, P., & Ræder, T. (2021). High-Performance Asynchronous Byzantine Fault Tolerance Consensus Protocol. *IEEE International Conference on Blockchain* (pp. 476-483).<https://doi.org/10.1109/Blockchain53845.2021.00073>

Lauinger, J., Ernstberger, J., Regnath, E., Hamad, M., & Steinhorst, S. (2021). A-PoA: Anonymous Proof of Authorization for Decentralized Identity Management. In *2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)* (pp. 1-9). Sydney, Australia. doi:10.1109/ICBC51069.2021.9461082.

Miller, A., Xia, Y., Croman, K., Shi, E., & Song, D. (2016). The Honey Badger of BFT Protocols. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (pp. 31–42).<https://doi.org/10.1145/2976749.2978399>

Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. Retrieved from https://bitcoin.org/bitcoin.pdf

Ramakrishna Kotla, Lorenzo Alvisi, Michael Dahlin, Allen Clement, and Edmund L. Wong. (2009). Zyzzyva: Speculative Byzantine fault tolerance. *ACM Transactions on Computer Systems (TOCS), 27*(4), 7:1–7:39.<https://doi.org/10.1145/1658357.1658358>

Reddy, B. S. (2021). securePrune: Secure block pruning in UTXO based blockchains using Accumulators. In *2021 International Conference on COMmunication Systems & NETworkS (COMSNETS)* (pp. 174-178). Bangalore, India. doi:10.1109/COMSNETS51098.2021.9352892.

Silva, P., Matos, M., & Barreto, J. (2023). NimbleChain: Speeding up Cryptocurrencies in General-purpose Permissionless Blockchains. *Distributed Ledger Technology, 2*(1), Article 8.<https://doi-org.ezproxylocal.library.nova.edu/10.1145/3573895>

Woznica, A., & Kedziora, M. (2022). Performance and scalability evaluation of a permissioned Blockchain based on the Hyperledger Fabric, Sawtooth and Iroha. *Comput. Sci. Inf. Syst., 19*, 659-678.

Zhou, Q., Huang, H., Zheng, Z., & Bian, J. (2020). Solutions to Scalability of Blockchain: A Survey. *IEEE Access*. Advance online publication.<https://doi.org/10.1109/ACCESS.2020.2967218>