**Introduction:** The exploratory research for the Winter 2024 ISEC 885 course aims to identify and address key challenges in the field of Asynchronous Consensus and Permission-less Systems. Specifically, the research endeavors to utilize insights from an established protocol called Aleph to enhance its own efficiency and scalability by reducing communication complexity. Improving communication complexity in an asynchronous permissionless environment will not only streamline the communication process within blockchain networks but also enhance their scalability and efficiency. By investigating the integration of Rivest–Shamir–Adleman (RSA) accumulators into the Reliable Broadcast (RBC) protocol instead of using Merkle trees, this research seeks to achieve a significant reduction in communication complexity from O(N^2 log N) to O(N^2), thereby addressing a critical bottleneck in the performance of the Aleph permissionless asynchrnous blockchain system. This enhancement is expected to bolster the network's capacity to handle larger transaction volumes and accommodate growing network sizes.

In the realm of network protocols and by abstraction of consensus protocols, the debate between synchronous and asynchronous approaches has been that synchronous protocols operate under the assumption of predictable timing, while asynchronous protocols allow for greater flexibility by not requiring strict timing assumptions. More specifically, asynchronous consensus ensures that systems can reach an agreement even when there is no bound on the time it takes for messages to be delivered. A bound time refers to a maximum limit or constraint on the duration within which a message can be transmitted or delivered between nodes in a network (Miller et al., 2016). This research favors the asynchronous setting because of its decentralization, instantaneous progress, and resilience to network variability. This has been supported in the previous research of asynchronous protocols such as the Honey Badger Byzantine Fault Tolerant Protocol (HBFT), Beat, Dumbo, and the Asynchronous Byzantine Fault Tolerance (ABFT) protocol (Miller et al., 2016, Knudsen et al., 2021, Duan et al., 2018, and Guo, Lu, Tang, Xu, & Zhang, 2020).

Permissioned and permissionless blockchains represent two different paradigms in the realm of distributed ledgers. When it comes to the blockchain trilemma, the advantage of having a permissioned system is that they typically make gains in security and scalability while giving up decentralization. On the other hand, permissionless systems typically have a higher decentralization factor with weaker scalability and security (Woznica & Kedziora, 2022). This research favors permissionless over permissioned because of the higher decentralization in the blockchain trilemma. The reason being, decentralization promotes the core principle of blockchain technology. For instance, Bitcoin, a prominent example of a permissionless blockchain, operates without the need for central authority, allowing anyone to participate in the network without requiring permission (Nakamoto, 2008). Permissionless blockchains are important because it allows community-driven governance, ensuring that decisions are made through decentralized processes involving network participants. The permissionless approach fosters inclusivity and ownership among stakeholders, while enhancing the resilience and sustainability of the network when compared to permissioned systems (Silva, Matos, & Barreto, 2023).

When it comes to asynchronous consensus using permissioned vs permissionless ledgers, the majority of asynchronous consensus research has involved permissioned ledgers. This being that the initial research in the field of asynchronous consensus is implemented with trusted dealers for Distributed Key Generation (DKG) protocols (Abraham et al., 2021; Das et al., 2020). Having a trusted dealer for DKG, and by definition making the blockchain permissioned, has been seen throughout the conception of asynchronous consensus, such as in the initial works of HBFT, Beat, Dumbo, and the Asynchronous Byzantine Fault Tolerant (ABFT) protocols (Miller et al., 2016, Duan et al., 2018, Guo et al., 2020, and Knudsen et al., 2021). Although these protocols lay the foundation for the field of asynchronous consensus, they do not enable a fully permission-less system and even call for additional research to advance the field of permission-less asynchronous consensus without DKG (Knudsen et al., 2021).

The lack of asynchronous consensus without DKG led to the conception of the Aleph protocol (Gągol et al., 2019).The creation of the Aleph protocol addressed the critical gap in the field of asynchronous consensus by enabling permissionless systems without the need for trusted dealers. It represents a significant advancement in blockchain technology by providing a solution that improved decentralization, security, and resilience in asynchronous consensus protocols.The same way HBFT touts itself as the first practical asynchronous consensus protocol (Miller et al., 2016). Aleph touts itself as the first practical asynchronous consensus protocol without a trusted dealer (Gągol et al., 2019).The Aleph protocol provides qualitative value by enabling a more permissionless environment for consensus without the need for trusted dealers, thereby fostering greater decentralization, inclusivity, and resilience in blockchain networks. Its removal of the trusted dealer requirement enhances trust among network participants and promotes community-driven governance, aligning with the core principles of decentralization and ownership in blockchain technology.Aleph not only lacks a trusted dealer to make it permission-less, but also contributes to the field of asynchronous consensus by using a Direct Acyclic Graph (DAG) for consensus instead of sequential Asynchronous Common Subset (ACS), thus providing a theoretical alternative for implementing atomic Reliable Broadcast (RBC) in lieu of sequentially executing ACS (Guo et al., 2022). Aleph provides a reputable protocol for this research because it executes consensus asynchronously without giving up its permission-less properties as seen in the past with trusted dealers used in previous asynchronous consensus protocols.

**Problem:** *Scalability of Communication Complexity*

The scalability of communication complexity of the Aleph protocol is the critical issue addressed in this research. The communication complexity refers to the amount of communication required among nodes in a network to achieve consensus or reliably broadcast messages. In the context of Aleph, the RBC protocol is a fundamental component responsible for disseminating messages across the network. The research identifies a bottleneck in the scalability of the Aleph protocol due to the communication complexity of its RBC protocol. Specifically, the protocol's communication complexity is currently O(N^2 log(N)), where N represents the number of nodes in the network (Guo et al., 2022). This complexity indicates that as the network grows in size, the communication overhead increases quadratically and logarithmically, potentially limiting the protocol's scalability.The value of this scalability improvement lies in its ability to enable the Aleph protocol to handle larger transaction volumes and accommodate network growth effectively. As blockchain networks continue to expand and attract more participants, scalability becomes increasingly crucial for ensuring the network's performance and sustainability. By comparison Bitcoins communication complexity is roughly linear with the number of nodes (O(N)) (Nakamoto, 2008). Therefore, by comparing Aleph's O(N^2 log N) complexity with Bitcoin's O(N) complexity, we can quantify the scalability gap and emphasize the need for improvement in Aleph's communication efficiency to achieve a level comparable to established synchronous permissioned blockchain networks like Bitcoin. By reducing communication complexity, the research enhances the scalability of the Aleph protocol, making it more resilient to network congestion and better equipped to support the growing demands of decentralized applications and transactions.

**Goal:** *Improving Communication Complexity*

The goal of this research is to enhance the scalability of the Aleph protocol by implementing an improved RBC protocol using RSA accumulators instead of Merklee trees. The key goal is to reduce the communication complexity from its current state of O(N^2 log N) to O(N^2). To achieve this objective, the research aims to improve transaction throughput, network latency, and resource utilization. The goals of improvement include defining enhanced transaction throughput by evaluating the number of transactions processed per unit of time before and after the implementation of the improved protocol. Additionally, the aim is to decrease network latency by determining the time taken for message propagation across the network both before and after the implementation. Lastly, the goal is to decrease resource utilization by assessing the computational and network resources consumed during message dissemination before and after the implementation. In conclusion, this research endeavors to significantly enhance the scalability and efficiency of the Aleph protocol by implementing an improved RBC protocol using RSA accumulators. With the goals of reducing communication complexity and improving transaction throughput, network latency, and resource utilization.

**Review of the Literature**

The evolution of asynchronous consensus protocols included seminal works like the HBFT, Beat, Dumbo, and the ABFT protocol. These protocols initially focused on permissioned ledgers, requiring trusted dealers for DKG. While these protocols laid the foundation for asynchronous consensus, they lacked the ability to enable fully permission-less systems, highlighting the need for further research in the area of asynchronous permissionless ledgers .

Following the conception of the field of asynchronous consensus was the first asynchronous consensus protocol without a trusted dealer, the Aleph protocol. Being the first truly asynchronous permissionless ledger compared to its permissioned counterparts. Aleph being permissionless fosters decentralization and resilience but lacks scalability with the communication complexity from its current state being O(N^2 log N) and even writes for the need to improve the communication by potentially employing RSA accumulators.

Using RSA accumulators instead of Merkle trees in blockchain has been supported in the recent works of (Hussein & Al-Gailani, 2022) but does not explicitly dive into if the research is on a permissioned or permissionless ledger. Similar research such as (Reddy, 2021) proposes a scheme called securePrune to reduce the storage space and synchronization time of nodes joining a Peer-to-Peer (P2P) network in a blockchain like Bitcoin, using RSA accumulators but is done in a synchronous environment. Similarly another example of using RSA accumulators in a decentralized fashion is seen in the works of (Lauinger et al., 2021). The paper introduces the Anonymous Proof of Authorization (A-PoA) protocol, designed to tackle the trust relation challenge within Self-Sovereign Identity Management (SSIM) frameworks reliant on Verifiable Credentials (VCs). A-PoA is tailored for permissioned environments, facilitating decentralized and anonymous authorization of CIAs. It harnesses RSA accumulators, alongside Non-Interactive Zero-Knowledge Proofs, to authorize CIAs for credential issuance. From the above research we can see RSA accumulators being used for synchronous and permissioned environments but not in an asynchronous permissionless manner.

**Approach**

The approach for this research focuses on addressing the scalability challenges faced by the Aleph protocol through the integration of RSA accumulators into the RBC protocol. The goal is to enhance the protocol's efficiency and scalability by reducing communication complexity. Firstly, the research will implement an improved reliable broadcast protocol using RSA accumulators, replacing Merkle Trees, to achieve a significant reduction in communication complexity from O(N^2 log N) to O(N^2). This approach targets the bottleneck in the performance of the Aleph permissionless asynchronous blockchain system, aiming to bolster the network's capacity to handle larger transaction volumes and accommodate growing network sizes.

To assess the effectiveness of the scalability improvement, the research will employ several metrics. Transaction throughput will be measured to evaluate the number of transactions processed per unit of time before and after the implementation of the improved protocol. A higher transaction throughput would indicate improved scalability. Network latency will also be evaluated to determine the time taken for messages to propagate across the network. A reduction in network latency suggests improved efficiency and scalability. Furthermore, the research will assess resource utilization, focusing on the computational and network resources consumed during message dissemination. Lastly, the growth rate of the Aleph network in terms of the number of nodes and transactions over time will be monitored to measure growth. This metric provides insights into the protocol's ability to effectively scale to accommodate increasing demand and network growth.

In conclusion, the research will compare the original Aleph protocol with the new RSA accumulator-based RBC protocol to gather and compare metrics related to transaction throughput, network latency, resource utilization, and network growth. This comparative analysis will help evaluate the effectiveness of the scalability improvement and provide valuable insights into the protocol's performance and scalability.

**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).

Adam Gągol, Damian Leśniak, Damian Straszak, and Michał Świętek. (2019). Aleph: Efficient Atomic Broadcast in Asynchronous Networks with Byzantine Nodes. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies* (pp. 214–228).

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.

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>

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

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>