Paper Review: Dumbo- Faster Asynchronous BFT Protocols

1 Introduction

This paper describes two new asynchronous Byzantine Fault Tolerant protocols, Dumbo1 and Dumbo2, that aim to enhance the efficiency of atomic broadcast in asynchronous networks. The protocols are based on HoneyBadgerBFT, which relies on asynchronous common subset (ACS) to achieve consensus. Dumbo1 and Dumbo2 introduce redesigned ACS protocols to minimize the use of asynchronous binary agreement instances, a major performance bottleneck in HoneyBadgerBFT.

2 Strong Points

- **Scalability**: Dumbo1 and Dumbo2 achieve considerably improved scalability since their high-performance behavior is maintained as the number of nodes increases. Reduced instances of ABA make it capable of being deployed in larger networks where latency does not grow at a rate necessary to justify its use in real-world applications.
- **Use of MVBA**: Dumbo2 utilizes Multi-Valued Validated Byzantine Agreement (MVBA) to achieve constant running time, challenging previous perspectives about the inefficiency of using MVBA for the actual construction of ACS protocols.
- **Asymptotic Efficiency**: Both Dumbo1 and Dumbo2 have significant asymptotic improvements in running time compared to HoneyBadgerBFT. Dumbo1 reduces the expected running time from O(log n) to O(log κ) wherein κ is a security parameter independent of the number n of nodes. Finally, Dumbo2 pushes this even to an optimal O(1), which is asymptotically optimal.

3 Weak points

- A threshold coin-tossing protocol, such as the one described in the selection of CE, introduces a critical weakness: reliance on a trusted third party, the dealer, for committee election.

- The paper does not discuss in detail how to optimally select the value of κ for Dumbo1; this could significantly affect its performance and security.
- The risk of having no honest node in the committee can be mitigated by choosing a sufficiently large κ value. However, this introduces a trade-off with performance: a larger κ increases the number of ABA instances and slows down the protocol.

4 Detailed Feedback

- **Committee Election Weakness:** This reliance on a centralized entity to generate and distribute private functions - $CShare_i$ to each node, and public functions ($CShareVerify$, $CToss$) introduces dependency on a centralized entity. A compromised/malicious dealer may easily manipulate these functions to affect committee selection and, in turn, enable the selection of a committee dominated by malicious nodes. Such a case directly undermines Dumbo1 security because the protocol relies, for its correct operation and security arguments, on having at least one honest committee node.
- **Parameter Selection:** The paper does not provide precise guidance on selecting an optimal κ value. The authors suggest that the system designer can choose κ based on a desired error parameter ϵ_0, ensuring $(1/3)^{\kappa_0} \leq \epsilon_0$. However, this choice involves a trade-off between security and performance, and the specific requirements of the application should be considered.
- **Tradeoff in CE:** The larger the committee size κ is in Dumbo1, the higher the probability that at least one honest node will be included, which strengthens security. However, a larger κ requires more concurrent executions of ABA, which directly influences performance. Each additional ABA instance adds complexity and communication overhead, slowing down the protocol and potentially increasing latency. This trade-off between security (higher κ) and performance (lower κ) requires careful consideration based on the specific application's needs.

5 Conclusion

This paper represents an important advance in asynchronous BFT protocols. The novelty of ABA instance reduction and practical performance gains make this work important. Further discussion into potential vulnerabilities and ways to mitigate them, could prove to be useful.