

Toward Great Machine Replication: Revising Leader-Based Consensus One Building Block at a Time



Richard von Seck ¹ and Georg Carle ¹

Abstract

Fault-tolerant systems can be constructed, using the State Machine Replication (SMR) approach [Sc90]. SMR offers a method to increase reliability in face of e.g., crash [SS83] or byzantine [LSP82] faults. While modern systems with Byzantine Fault Tolerance (BFT) demonstrate increasing efficiency, the involved overhead limits their applicability. In the last four decades, a plethora of work has been dedicated to the study and optimization of this problem space [Zh24]. The state of the art is complex and identifying a suitable solution for a target scenario is non-trivial. Conceptual modification of existing algorithms typically requires a rework of safety and liveness proofs, and the modified protocol becomes yet another choice in the set of candidates. Furthermore, with increasing performance of BFT-SMR systems, components can become a limiting factor for some scenarios [Ca23].

We propose a different approach toward improving practical BFT-SMR [Se22]. The study of BFT-SMR components – building blocks – holds optimization potential, while preserving proven algorithm guarantees. We identify frequently assumed components from related work as candidates and study selected building blocks in theory and practice. The seminal *HotStuff* [Yi19] algorithm is used as a representative BFT-SMR system. We analyze, implement, and experimentally verify modification of the chosen building blocks, using the freely available HotStuff codebase. As candidate building blocks, we study (1) the underlying network transport [Se24], including four transport protocols and two secure channel implementations, and (2) underlying cryptographic primitives such as signature schemes [vSRC24], including 3 threshold signature schemes, one regular digital signature scheme, and additional metrics to support detailed evaluation. In the experiments, we vary a range of typical BFT-SMR and building-block-specific parameters.

We observe varying results. Depending on the target building block, parameters, and scenario, both performance improvement and decline are possible. For (1), secure channel usage incurs a small performance overhead in our scenarios. We regard their usage as functional. Furthermore, we consider TCP to be the best default transport protocol choice, except for edge cases. We observe performance improvements of up to ~ 20% between transport protocols and configurations for some scenarios. These differences are primarily governed by protocol logic for e.g., handling of adverse network conditions such as loss.

¹ Technical University of Munich, Chair of Network Architectures and Services, Boltzmannstr. 3, 85748 Garching, Germany, seck@net.in.tum.de,  <https://orcid.org/0009-0006-6060-3163>; carle@net.in.tum.de,  <https://orcid.org/0000-0002-2347-1839>

For (2), interactiveness limits the applicability of signature schemes in BFT-SMR context. We find BLS threshold signatures to be the most compatible from the set of studied schemes, while they incur elevated cryptographic base operation costs. Studied semi-interactive schemes may have performance improvement potential if presigning can be executed fast enough to not limit consensus speed. For small to medium network sizes, we do not observe performance improvements through usage of threshold signatures and consider simple digital signatures to be the best default solution here. Throughput differences between schemes can be alleviated by suitable choice of batch size, while latency differences remain.

The study of building blocks offers a helpful method to explore the optimization potential of BFT-SMR systems. Impact and significance vary with building block, scenario, and upper-layer algorithm semantics. We contribute to the state of the art by extending studied schemes and parameters in the chosen areas in number and detail.

Bibliography

- [Ca23] Camaioni, Martina; Guerraoui, Rachid; Monti, Matteo; Roman, Pierre-Louis; Vidigueira, Manuel; Voron, Gauthier: Chop Chop: Byzantine Atomic Broadcast to the Network Limit. arXiv preprint arXiv:2304.07081, 2023.
- [LSP82] Lamport, Leslie; Shostak, Robert; Pease, Marshall: The Byzantine Generals Problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 4(3):382–401, 1982.
- [Sc90] Schneider, Fred B: Implementing Fault-Tolerant Services Using the State Machine Approach: A tutorial. *ACM Computing Surveys (CSUR)*, 22(4):299–319, 1990.
- [Se22] von Seck, Richard; Rezabek, Filip; Jaeger, Benedikt; Gallenmüller, Sebastian; Carle, Georg: BFT-Blocks: The Case for Analyzing Networking in Byzantine Fault Tolerant Consensus. In: 2022 IEEE 21st International Symposium on Network Computing and Applications (NCA). volume 21, pp. 35–44, 2022.
- [Se24] von Seck, Richard; Rezabek, Filip; Gallenmüller, Sebastian; Carle, Georg: On the Impact of Network Transport Protocols on Leader-Based Consensus Communication. In: 6th ACM International Symposium on Blockchain and Secure Critical Infrastructure (BSCI:’24). July 2024.
- [SS83] Schlichting, Richard D; Schneider, Fred B: Fail-Stop Processors: An Approach to Designing Fault-Tolerant Computing Systems. *ACM Transactions on Computer Systems (TOCS)*, 1(3):222–238, 1983.
- [vSRC24] von Seck, Richard; Rezabek, Filip; Carle, Georg: Thresh-Hold: Assessment of Threshold Cryptography in Leader-Based Consensus. In: 2024 IEEE 49th Conference on Local Computer Networks (LCN). IEEE, pp. 1–8, 2024.
- [Yi19] Yin, Maofan; Malkhi, Dahlia; Reiter, Michael K; Gueta, Guy Golan; Abraham, Ittai: HotStuff: BFT Consensus with Linearity and Responsiveness. In: *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing*. ACM, 2019.
- [Zh24] Zhang, Gengrui; Pan, Fei; Mao, Yunhao; Tijanac, Sofia; Dang’ana, Michael; Motepalli, Shashank; Zhang, Shiquan; Jacobsen, Hans-Arno: Reaching Consensus in the Byzantine Empire: A Comprehensive Review of BFT Consensus Algorithms. *ACM Computing Surveys*, 56(5):1–41, 2024.