Byzantine Fault Tolerant Protocols with Near-Constant Work per Node without Signatures

Philipp Schneider[1]

Abstract: Numerous distributed tasks have to be handled in a setting where a fraction of nodes behaves Byzantine, that is, deviates arbitrarily from the intended protocol. Resilient, deterministic protocols rely on the detection of majorities to avoid inconsistencies if there is a Byzantine minority, which requires individual nodes to handle a communication workload that is proportional to the size of the network — an intolerable disadvantage in large networks. Randomized protocols circumvent this by probing only small parts of the network, thus allowing for consistent decisions quickly and with a high level of confidence with communication that is near-constant in the network size. However, such protocols usually come with the drawback of limiting the fault tolerance of the protocol, for instance, by severely restricting the number or type of failures that the protocol can tolerate.

We present randomized protocols to reliably aggregate and broadcast information, form consensus and compute common coins that tolerate a constant fraction of Byzantine failures, do not require cryptographic signatures and have a near-constant time and message complexity per node. Our main technique is to compute a system of witness committees as a pre-computation step almost optimally. This pre-computation step allows to solve the aforementioned distributed tasks repeatedly and efficiently, but may have far reaching further applications, e.g., for sharding of distributed data structures.

Keywords: Byzantine Fault Tolerance, Randomized Protocols, Reliable Broadcast, Consensus

Introduction. Distributed algorithms like reliable broadcast and consensus are central to applications such as replicated databases, large-scale cloud systems, and blockchains. Such system must maintain correctness in the event of node failures, including adversarial (Byzantine) behavior. Existing solutions typically rely on large quorums, leading to linear communication overhead for each node. This is problematic for sensor networks, industrial IoT, or decentralized energy grids, with huge numbers of nodes but limited bandwidth and computational power per node.

Randomized approaches can reduce each node's load to near-constant (i.e., terms that scale negligibly in n), but they often limit the adversary model or assume cryptographic methods that incur a significant computational burden or come with other disadvantages. Our work addresses core tasks such as reliable broadcast, consensus, and common coin computation that achieves near-constant communication overhead per node while offering resilience to an adaptive adversary controlling a constant fraction of nodes.

In large-scale networks, distributed systems repeatedly solve tasks like consensus and reliable broadcast, which suggests a two-phase strategy: a one-off *pre-computation* step and a much more efficient *execution* step for each problem instance. In the pre-computation step, we build a linear number of small witness committees each with an honest majority and publish them network-wide. This structure enables parallelism and reduces both round complexity and per-node workload in the execution step. Furthermore, this approach confines adversary assumptions to the pre-computation step, whereas in the execution step we only rely on

[1] University of Bern, Bern, Switzerland, philipp.schneider2@unibe.ch, https://orcid.org/0000-0001-9660-1270

the pre-computed system of witness committees, even allowing new nodes to be added after the pre-computation step. This makes our approach modular and adaptable to varying requirements or models.

Summary of Contributions. *(1) Pre-computing a System of Witness Committees.* Our main technical contribution is the introduction of an efficient, randomized procedure to compute a *system of witness committees* in networks with a constant fraction of Byzantine nodes. The number of committees is linear in the network-size n, each committee is near-constant in size, contains a majority of honest nodes, and is recognized by all honest participants. By splitting the construction into phases that build on top of each other, we avoid expensive all-to-all communication and do not rely on cryptographic signatures. Further, the required number of rounds is $\widetilde{O}(n)$ [2] and the communication workload per node is near-constant, which is almost optimal considering the information-theoretic lower bounds.

(2) Reliable Broadcast. With the system of witness committees in place, we design a deterministic protocol for reliable broadcast in an asynchronous network. We connect the committees in a broadcast tree, and a single message is efficiently disseminated in the tree having each node handle only near-constant work. The degree δ of the broadcast tree influences the trade-off between per-node workload $\widetilde{O}(\delta)$ and round complexity $O(\log_\delta n)$. For instance, choosing $\delta = \frac{\log n}{\log \log n}$ yields $O(\log \log n)$ rounds, demonstrating sublogarithmic broadcast complexity with near-constant workload per node under strong adversarial conditions and without using signatures.

(3) Reliable Aggregation. We adapt the reliable broadcast routine to conduct *reliable aggregation* of data values. A subset of nodes has input values for an aggregation function (such as sum, minimum, or XOR). Each node commits its value to a witness committee, which then aggregates the inputs securely. The aggregation continues up the broadcast tree, and the final outcome is delivered back down with per-node workload $\widetilde{O}(\delta)$ and round complexity $O(\log_\delta n)$. This procedure assumes synchrony as aggregation inputs must be collected within a time bound, but does not require cryptographic signatures.

(4) Common Coin and Consensus. Using reliable aggregation, we construct a *common coin* beacon in $O(\log_\delta n)$ rounds and $\widetilde{O}(\delta)$ communication per node. Each node contributes a random local coin, and as soon as at least one honest coin is included, the resulting global coin remains out of the adversary's control (under an adversary who has access to all information with a 1-round delay). For multi-value consensus, we rely on reliable aggregation identify whether one proposal has an absolute majority or not and to break symmetry. Notably, this bypasses classical lower bounds by shifting the challenging requirements to the pre-computation phase.

For further details and related work see [Sc25].

Bibliography

[Sc25] Schneider, Philipp: Byzantine Fault Tolerant Protocols with Near-Constant Work per Node without Signatures, 2025. https://arxiv.org/abs/2501.05377.

[2] The $\widetilde{O}(\cdot)$ notation masks terms that are polynomial in $\log n$ and $\widetilde{O}(1)$ can be interpreted as *near-constant*.