

An Asymmetric DAG-based Consensus Algorithm

Ignacio Amores-Sesar ¹, Christian Cachin ², Juan Villacis ², and Luca Zanolini ³

Abstract: We introduce the first asymmetric directed acyclic graph (DAG)-based consensus algorithm. To achieve this, we adapt key building-blocks from the well-known DAG-Rider protocol, namely reliable broadcast, common coin, and gather, by replacing them with asymmetric equivalents. During the process of replacing gather, we explore the problem of obtaining common cores in the asymmetric world. We show with a counterexample that existing symmetric approaches are not applicable in these settings. This finding implies that DAG-based consensus protocols that rely on common cores for the liveness analysis of their commit rules, like DAG-Rider, Bullshark, and Tusk, must be adapted to work effectively in asymmetric contexts. Furthermore, we propose the first asymmetric common core primitive—a gather equivalent that operates with asymmetric quorums.


Keywords: Asymmetric Trust, Directed Acyclic Graphs, Atomic Broadcast, Gather



1 Talk Overview


In recent years, some research on consensus protocols has moved beyond linear, single-chain blockchains and toward *Directed Acyclic Graphs* (DAGs), enabling parallelization of transaction dissemination and ordering. Several works [Ke21, Sp22] leverage DAGs to improve throughput (by concurrently batching transactions) and latency (by reducing bottlenecks inherent in a single leader or chain). Broadly, these protocols separate the act of distributing transactions from the act of finalizing their order.

Despite the clear performance benefits of DAG-based protocols, *existing proposals rely on symmetric trust assumptions*. In other words, they employ global adversarial thresholds (e.g., assuming that at most f of n participants are faulty) that remain uniform across all nodes. This model may fail to capture the nuanced and heterogeneous trust relationships that naturally arise in open and decentralized networks.

Asymmetric distributed trust, introduced by Damgård et al. [Da07] and further developed by Alpos et al. [Al24], refers to trust configurations where participants have differing views of the trustworthiness of other nodes. Unlike traditional symmetric models, where trust is uniformly and globally shared, asymmetric trust enables nodes to individually specify their trust relationships, leading to more flexible but complex trust configurations. Alpos *et al.* [Al24] highlight how such trust assumptions can be leveraged to design protocols such as reliable broadcast and consensus that remain resilient under varying participant perspectives, offering greater adaptability in heterogeneous environments.

¹ Aarhus University, amores-sesar@cs.au.dk,  <https://orcid.org/0000-0002-1751-1515>

² University of Bern, christian.cachin@unibe.ch,  <https://orcid.org/0000-0001-8967-9213>;
juan.villacis@unibe.ch,  <https://orcid.org/0009-0006-0110-8613>

³ Ethereum Foundation, luca.zanolini@ethereum.org,  <https://orcid.org/0000-0003-4655-3172>

While these systems demonstrate that subjective trust can be integrated into consensus, their designs are predominantly built on linear ledgers. The interplay between asymmetric trust assumptions and DAG-based concurrency remains uncharted.

Building on the line of research on asymmetric trust [Al24, Da07] we investigate how to reconcile subjective fail-prone systems with the concurrency and efficiency offered by DAG-based consensus protocols. Asymmetric trust allows each node to define its own conditions under which it deems other parties faulty, thereby relaxing the need for a single global adversarial threshold. Such flexibility more accurately represents the varied trust relationships observed in real-world settings, where different participants may have fundamentally different notions of risk, reliability, or expertise.

We present the first asymmetric DAG-based consensus protocol by extending the DAG-Rider [Ke21] protocol to accommodate locally defined fail-prone systems. More specifically, each node in the system determines its own assumptions about which parties may be Byzantine, and the DAG structure is employed to order transactions despite these potentially divergent views. In order to do this, we also explore the asymmetric gather problem, show why existing symmetric approaches [Ke21] are not suited for asymmetric environments, and introduce the first asymmetric gather protocol.

By unifying asymmetric quorum systems with DAG-based protocol design, our construction harnesses the throughput advantages of concurrent transaction processing while granting nodes the freedom to adopt individualized trust configurations. The result is a trust-flexible consensus approach that applies naturally to open, decentralized, and diverse environments in which a single adversarial threshold fails to capture the reality of distributed trust. Specifically, DAG-based approaches are well-suited to the asymmetric model, as their leaderless structure aligns with the freedom of choice and independent trust inherent to such systems.

Bibliography

- [Al24] Alpos, Orestis; Cachin, Christian; Tackmann, Björn; Zanolini, Luca: Asymmetric distributed trust. *Distributed Comput.*, 37(3):247–277, 2024.
- [Da07] Damgård, Ivan; Desmedt, Yvo; Fitzi, Matthias; Nielsen, Jesper Buus: Secure Protocols with Asymmetric Trust. In: *ASIACRYPT*. volume 4833 of *Lecture Notes in Computer Science*. Springer, pp. 357–375, 2007.
- [Ke21] Keidar, Idit; Kokoris-Kogias, Eleftherios; Naor, Oded; Spiegelman, Alexander: All You Need is DAG. In (Miller, Avery; Censor-Hillel, Keren; Korhonen, Janne H., eds): *PODC '21: ACM Symposium on Principles of Distributed Computing, Virtual Event, Italy, July 26-30, 2021*. ACM, pp. 165–175, 2021.
- [Sp22] Spiegelman, Alexander; Giridharan, Neil; Sonnino, Alberto; Kokoris-Kogias, Lefteris: Bullshark: DAG BFT Protocols Made Practical. In: *CCS*. ACM, pp. 2705–2718, 2022.