Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

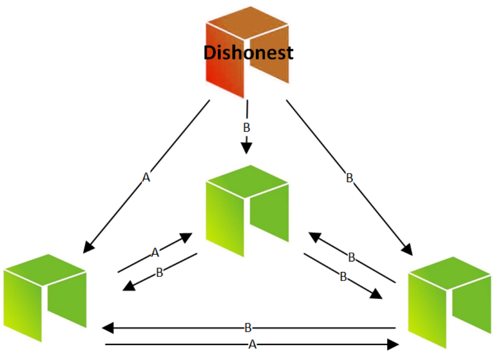
Prof. R. Wattenhofer

Byzantine Agreement: Optimistic Analysis

A distributed system consists of n nodes. The system is byzantine fault tolerant (BFT) if it can tolerate at most $f < \frac{n}{3}$ arbitrarily malicious (*byzantine*) nodes. BFT protocols have been studied in great detail since many decades, both in theory and practice. Nowadays, BFT protocols are the key to building "permissioned blockchains", an area traditionally known as "state machine replication" [9, 10].

In practice, BFT protocols have many applications ranging from online shopping to credit card transactions, cryptocurrencies and stock market trades; whenever a set of clients makes concurrent requests for (or with) limited resources, the service providers have an interest to both prevent fraudulent and tolerate faulty behaviour in the system.

From a research perspective, the interest in BFT systems has first been reignited by Castro and Liskov when they presented their "Practical" BFT (PBFT) system [2]. After PBFT, a large number of other BFT systems emerged [8, 11, 5, 12, 13, 1, 7, 3, 4, 6]. Many of these systems try to minimize the delay until transactions are committed. Unlike otherwise common in computer science, however, they do not optimize for the worst possible case. The aim of this research domain is to design systems that are *practical*, that is, systems that provide strong theoretic safety guarantees (e.g. may cope with arbitrary bad networking conditions), but do not incur an unproportional overhead during "normal operation" [2]. To that end, the aforementioned protocols make a varying degree of optimistic assumptions (e.g., no message timeouts occur, leader is not byzantine, no node is byzantine, ...).



In this project we aim to analyze what limitations come with different considered models as, depending on the optimistic assumptions, the optimal latency may differ. In this way, we aim to give an objectively justifiable ground for reasoning why different models should be considered to design optimal systems for real-world applications.

Requirements: The expected outcome of this project is of theoretic nature; hence, you should be willing to formulate both theorems and proofs independently. Progress, open problems and new ideas will be discussed in collaborative (at least) weekly meetings!

Interested? Please contact us for more details!

Contacts

- Roland Schmid: roschmi@ethz.ch, ETZ G94

References

[1] E. Buchman, J. Kwon, and Z. Milosevic. The latest gossip on bft consensus. *arXiv preprint arXiv:1807.04938*, 2018.

[2] M. Castro, B. Liskov, et al. Practical byzantine fault tolerance. In *OSDI*, volume 99, pages 173–186, 1999.

[3] T. H. Chan, R. Pass, and E. Shi. Pala: A simple partially synchronous blockchain, 2018.

[4] Y. Gilad, R. Hemo, S. Micali, G. Vlachos, and N. Zeldovich. Algorand: Scaling byzantine agreements for cryptocurrencies. In *Proceedings of the 26th Symposium on Operating Systems Principles*, pages 51–68. ACM, 2017.

[5] G. Golan-Gueta, I. Abraham, S. Grossman, D. Malkhi, B. Pinkas, M. K. Reiter, D.-A. Seredinschi, O. Tamir, and A. Tomescu. SBFT: a scalable decentralized trust infrastructure for blockchains. *arXiv preprint arXiv:1804.01626*, 2018.

[6] M. M. Jalalzai, C. Busch, and G. Richard III. Proteus: A scalable BFT consesus protocol for blockchains. *arXiv preprint arXiv:1903.04134*, 2019.

[7] A. Kiayias and A. Russell. Ouroboros-BFT: A simple byzantine fault tolerant consensus protocol. Technical report, Cryptology ePrint Archive, Report 2018/1049, 2018. https://eprint.iacr.org, 2018.

[8] R. Kotla, L. Alvisi, M. Dahlin, A. Clement, and E. Wong. Zyzzyva: speculative byzantine fault tolerance. In *ACM SIGOPS Operating Systems Review*, volume 41, pages 45–58. ACM, 2007.

[9] L. Lamport. Using time instead of timeout for fault-tolerant distributed systems. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 6(2):254–280, 1984.

[10] F. B. Schneider. Implementing fault-tolerant services using the state machine approach: A tutorial. *ACM Computing Surveys (CSUR)*, 22(4):299–319, 1990.

[11] Y. J. Song and R. van Renesse. Bosco: One-step byzantine asynchronous consensus. In *International Symposium on Distributed Computing*, pages 438–450. Springer, 2008.

[12] J. Sousa, E. Alchieri, and A. Bessani. State machine replication for the masses with bft-smart. 2013.

[13] M. Yin, D. Malkhi, M. K. Reiter, G. G. Gueta, and I. Abraham. Hotstuff: Bft consensus in the lens of blockchain. *arXiv preprint arXiv:1803.05069*, 2018.