Errors found in distributed protocols

Protocol	Reference	Violation	Counter-example
PBFT[1]	[CL99]	liveness	[BRB21]
Chord	[Sto+01; LBK02]	liveness[2]	[Zav12; Zav17]
Pastry	[RD01]	safety	[AMW16; AMW18]
Generalised Paxos	[Lam05]	non-triviality[3]	[SS10]
FaB Paxos	[MA05; MA06]	liveness	[Abr+17]
Multi-Paxos[4]	[CGR07]	safety	[Mic+17]
Zyzzyva	[Kot+07; Kot+10]	safety	[Abr+17]
CRAQ	[TF09]	safety[5]	[Whi20]
JPaxos	[Koń+11]	safety	[Mic+17]
VR Revisited	[LC12]	safety	[Mic+17]
EPaxos	[MAK13]	safety	[Sut20]
EPaxos	[MAK13]	safety	[Whi21]
Raft	[OO14]	liveness[6]	[Hoc14]
Raft	[Ong14]	safety[7]	[AZ15; Ong15]
Raft	[OO14; Ong14]	liveness	[HA20; JHM21]
hBFT	[DPL15]	safety	[SKD19]
Tendermint	[Buc16]	liveness	[CV17]
CAESAR	[Aru+17]	liveness	[Ene+21]
DPaxos	[NAE18]	safety	[Whi+21]
Sync HotStuff	[Abr+19]	safety & liveness	[MC19]
Gasper	[But+20]	safety & liveness	[NTT21]

(Reference list starts on the next page.)

[1] With the read-only optimisation.

[2] Eventual reachability is Chord's key correctness property.

[3] Acceptors might accept commands that have not been proposed.

[4] As described in Paxos Made Live.

[5] Client reads might fail due to incorrect garbage collection.

[6] The joint consensus membership change algorithm described in the paper version of Raft had a liveness bug, which was fixed in Ongaro's PhD thesis.

[7] The bug is in the single-server membership change scheme described in Ongaro's thesis.

References

[Abr+17] Ittai Abraham et al. "Revisiting Fast Practical Byzantine Fault Tolerance". In: *arXiv:1712.01367 [cs]* (Dec. 2017). arXiv: 1712.01367. URL: `http://arxiv.org/abs/1712.01367` (visited on 2021-09-06).

[Abr+19] Ittai Abraham et al. *Sync HotStuff: Simple and Practical Synchronous State Machine Replication.* Tech. rep. 270. 2019. URL: `http://eprint.iacr.org/2019/270` (visited on 2021-09-16).

[AZ15] Brandon Amos and Huanchen Zhang. *15-812 Term Paper: Specifying and proving cluster membership for the Raft distributed consensus algorithm.* Tech. rep. 2015, p. 46. URL: `https://www.cs.cmu.edu/~aplatzer/course/pls15/projects/bamos.pdf`.

[Aru+17] Balaji Arun et al. "Speeding up Consensus by Chasing Fast Decisions". In: *2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN).* ISSN: 2158-3927. June 2017, pp. 49–60. DOI: `10.1109/DSN.2017.35`.

[AMW16] Noran Azmy, Stephan Merz, and Christoph Weidenbach. "A Rigorous Correctness Proof for Pastry". en. In: *Abstract State Machines, Alloy, B, TLA, VDM, and Z.* Ed. by Michael Butler et al. Vol. 9675. Series Title: Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016, pp. 86–101. ISBN: 978-3-319-33599-5 978-3-319-33600-8. DOI: `10.1007/978-3-319-33600-8_5`. URL: `http://link.springer.com/10.1007/978-3-319-33600-8_5` (visited on 2021-09-07).

[AMW18] Noran Azmy, Stephan Merz, and Christoph Weidenbach. "A machine-checked correctness proof for Pastry". en. In: *Science of Computer Programming* 158 (June 2018), pp. 64–80. ISSN: 01676423. DOI: `10.1016/j.scico.2017.08.003`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0167642317301612` (visited on 2021-09-07).

[BRB21] Christian Berger, Hans P. Reiser, and Alysson Bessani. "Making Reads in BFT State Machine Replication Fast, Linearizable, and Live". In: *arXiv:2107.11144 [cs]* (July 2021). arXiv: 2107.11144. URL: `http://arxiv.org/abs/2107.11144` (visited on 2021-11-10).

[Buc16] Ethan Buchman. "Tendermint: Byzantine Fault Tolerance in the Age of Blockchains". PhD thesis. Guelph, Ontario, Canada: University of Guelph, June 2016. URL: `https://atrium.lib.uoguelph.ca/xmlui/handle/10214/9769`.

[But+20] Vitalik Buterin et al. "Combining GHOST and Casper". In: *arXiv:2003.03052 [cs]* (May 2020). arXiv: 2003.03052. URL: `http://arxiv.org/abs/2003.03052` (visited on 2021-09-16).

[CV17] Christian Cachin and Marko Vukolić. "Blockchain Consensus Protocols in the Wild". In: *arXiv:1707.01873 [cs]* (July 2017). arXiv: 1707.01873. URL: http://arxiv.org/abs/1707.01873 (visited on 2021-09-16).

[CL99] Miguel Castro and Barbara Liskov. "Practical Byzantine Fault Tolerance". In: *3rd Symposium on Operating Systems Design and Implementation (OSDI 99)*. New Orleans, LA: USENIX Association, Feb. 1999. URL: https://www.usenix.org/conference/osdi-99/practical-byzantine-fault-tolerance.

[CGR07] Tushar D. Chandra, Robert Griesemer, and Joshua Redstone. "Paxos made live: an engineering perspective". en. In: *Proceedings of the twenty-sixth annual ACM symposium on Principles of distributed computing - PODC '07*. Portland, Oregon, USA: ACM Press, 2007, pp. 398–407. ISBN: 978-1-59593-616-5. DOI: 10.1145/1281100.1281103. URL: http://dl.acm.org/citation.cfm?doid=1281100.1281103 (visited on 2021-01-19).

[DPL15] Sisi Duan, Sean Peisert, and Karl N. Levitt. "hBFT: Speculative Byzantine Fault Tolerance with Minimum Cost". In: *IEEE Transactions on Dependable and Secure Computing* 12.1 (Jan. 2015). Conference Name: IEEE Transactions on Dependable and Secure Computing, pp. 58–70. ISSN: 1941-0018. DOI: 10.1109/TDSC.2014.2312331.

[Ene+21] Vitor Enes et al. "Efficient replication via timestamp stability". en. In: *Proceedings of the Sixteenth European Conference on Computer Systems*. Online Event United Kingdom: ACM, Apr. 2021, pp. 178–193. ISBN: 978-1-4503-8334-9. DOI: 10.1145/3447786.3456236. URL: https://dl.acm.org/doi/10.1145/3447786.3456236 (visited on 2021-09-16).

[Hoc14] Ezra Hoch. *Configuration changes*. Feb. 2014. URL: https://groups.google.com/g/raft-dev/c/xux5HRxH3Ic/m/mz_PDK-qMJgJ (visited on 2021-09-09).

[HA20] Heidi Howard and Ittai Abraham. *Raft does not Guarantee Liveness in the face of Network Faults*. Dec. 2020. URL: https://decentralizedthoughts.github.io/2020-12-12-raft-liveness-full-omission/ (visited on 2021-05-07).

[JHM21] Chris Jensen, Heidi Howard, and Richard Mortier. "Examining Raft's behaviour during partial network failures". In: *Proceedings of the 1st Workshop on High Availability and Observability of Cloud Systems*. HAOC '21. New York, NY, USA: Association for Computing Machinery, Apr. 2021, pp. 11–17. ISBN: 978-1-4503-8336-3. DOI: 10.1145/3447851.3458739. URL: https://doi.org/10.1145/3447851.3458739.

[Koń+11] Jan Kończak et al. *JPaxos: State machine replication based on the Paxos protocol*. Tech. rep. EPFL-REPORT-167765. 2011.

[Kot+07] Ramakrishna Kotla et al. “Zyzzyva: Speculative Byzantine Fault Tolerance”. In: *SIGOPS Oper. Syst. Rev.* 41.6 (Oct. 2007), pp. 45–58. issn: 0163-5980. doi: `10.1145/1323293.1294267`. url: `https://doi.org/10.1145/1323293.1294267`.

[Kot+10] Ramakrishna Kotla et al. “Zyzzyva: Speculative Byzantine Fault Tolerance”. In: *ACM Trans. Comput. Syst.* 27.4 (Jan. 2010). issn: 0734-2071. doi: `10.1145/1658357.1658358`. url: `https://doi.org/10.1145/1658357.1658358`.

[Lam05] Leslie Lamport. *Generalized Consensus and Paxos*. Tech. rep. MSR-TR-2005-33. Microsoft Research, Mar. 2005. url: `https://www.microsoft.com/en-us/research/publication/generalized-consensus-and-paxos/`.

[LBK02] David Liben-Nowell, Hari Balakrishnan, and David Karger. “Analysis of the Evolution of Peer-to-Peer Systems”. In: *Proceedings of the Twenty-First Annual Symposium on Principles of Distributed Computing*. PODC ’02. Monterey, California: Association for Computing Machinery, 2002, pp. 233–242. isbn: 1581134851. doi: `10.1145/571825.571863`. url: `https://doi.org/10.1145/571825.571863`.

[LC12] Barbara Liskov and James Cowling. *Viewstamped Replication Revisited*. en. Tech. rep. MIT-CSAIL-TR-2012-021. July 2012, p. 16.

[MA05] J.-P. Martin and L. Alvisi. “Fast Byzantine Consensus”. In: *2005 International Conference on Dependable Systems and Networks (DSN’05)*. ISSN: 2158-3927. June 2005, pp. 402–411. doi: `10.1109/DSN.2005.48`.

[MA06] Jean-Philippe Martin and Lorenzo Alvisi. “Fast Byzantine Consensus”. English. In: *IEEE Transactions on Dependable and Secure Computing* 3.3 (Sept. 2006). Num Pages: 202-215 Place: Washington, United States Publisher: IEEE Computer Society, pp. 202–215. issn: 15455971. doi: `http://dx.doi.org.libproxy1.nus.edu.sg/10.1109/TDSC.2006.35`. url: `http://www.proquest.com/docview/206534931/abstract/A91EECC1018D4A46PQ/1`.

[Mic+17] Ellis Michael et al. “Recovering Shared Objects Without Stable Storage”. en. In: (Aug. 2017). Appendix B, p. 27.

[MC19] Atsuki Momose and Jason Paul Cruz. *Force-Locking Attack on Sync Hotstuff*. Tech. rep. 1484. 2019. url: `http://eprint.iacr.org/2019/1484` (visited on 2021-09-16).

[MAK13] Iulian Moraru, David G. Andersen, and Michael Kaminsky. “There is more consensus in Egalitarian parliaments”. In: *Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles*. SOSP ’13. New York, NY, USA: Association for Computing Machinery, Nov. 2013, pp. 358–372. isbn: 978-1-4503-2388-8. doi:

10.1145/2517349.2517350. URL: http://doi.org/10.1145/2517349.2517350.

[NAE18] Faisal Nawab, Divyakant Agrawal, and Amr El Abbadi. "DPaxos: Managing Data Closer to Users for Low-Latency and Mobile Applications". In: *Proceedings of the 2018 International Conference on Management of Data*. SIGMOD '18. New York, NY, USA: Association for Computing Machinery, May 2018, pp. 1221–1236. ISBN: 978-1-4503-4703-7. DOI: 10.1145/3183713.3196928. URL: https://doi.org/10.1145/3183713.3196928 (visited on 2021-09-07).

[NTT21] Joachim Neu, Ertem Nusret Tas, and David Tse. "Ebb-and-Flow Protocols: A Resolution of the Availability-Finality Dilemma". In: *arXiv:2009.04987 [cs]* (Feb. 2021). arXiv: 2009.04987. URL: http://arxiv.org/abs/2009.04987 (visited on 2021-09-16).

[Ong14] Diego Ongaro. "Consensus: Bridging Theory and Practice". AAI28121474. PhD thesis. Stanford, CA, USA, 2014. ISBN: 9798662514218.

[Ong15] Diego Ongaro. *bug in single-server membership changes*. July 2015. URL: https://groups.google.com/g/raft-dev/c/t4xj6dJTP6E/m/d2D9LrWRza8J (visited on 2021-09-01).

[OO14] Diego Ongaro and John Ousterhout. "In Search of an Understandable Consensus Algorithm". In: *Proceedings of the 2014 USENIX Conference on USENIX Annual Technical Conference*. USENIX ATC'14. Philadelphia, PA: USENIX Association, 2014, pp. 305–320. ISBN: 9781931971102.

[RD01] Antony Rowstron and Peter Druschel. "Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems". en. In: *Middleware 2001*. Ed. by Rachid Guerraoui. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2001, pp. 329–350. ISBN: 978-3-540-45518-9. DOI: 10.1007/3-540-45518-3_18.

[SKD19] Nibesh Shrestha, Mohan Kumar, and SiSi Duan. "Revisiting hBFT: Speculative Byzantine Fault Tolerance with Minimum Cost". In: *arXiv:1902.08505 [cs]* (Apr. 2019). arXiv: 1902.08505. URL: http://arxiv.org/abs/1902.08505 (visited on 2021-09-16).

[Sto+01] Ion Stoica et al. "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications". In: *SIGCOMM Comput. Commun. Rev.* 31.4 (Aug. 2001), pp. 149–160. ISSN: 0146-4833. DOI: 10.1145/964723.383071. URL: https://doi.org/10.1145/964723.383071.

[Sut20] Pierre Sutra. "On the correctness of Egalitarian Paxos". en. In: *Information Processing Letters* 156 (Apr. 2020). ISSN: 0020-0190. DOI: 10.1016/j.ipl.2019.105901. URL: https://www.sciencedirect.com/science/article/pii/S002001901930184X (visited on 2021-04-01).

[SS10] Pierre Sutra and Marc Shapiro. *Fast Genuine Generalized Consensus*. Tech. rep. (corrected August 2010). Section 6.3. Feb. 2010, p. 62. URL: https://drive.google.com/open?id=0BwFkGepvBDQoRjNYRGJTdWQ0SzA.

[TF09] Jeff Terrace and Michael J. Freedman. "Object Storage on {CRAQ}: High-Throughput Chain Replication for Read-Mostly Workloads". In: 2009. URL: https://www.usenix.org/conference/usenix-09/object-storage-craq-high-throughput-chain-replication-read-mostly-workloads (visited on 2021-09-09).

[Whi20] Michael Whittaker. "CRAQ Bug". original-date: 2020-06-13T18:44:33Z. June 2020. URL: https://github.com/mwhittaker/craq_bug (visited on 2021-09-09).

[Whi21] Michael Whittaker. *EPaxos Dependency Set Compaction Bug*. original-date: 2018-11-03T04:31:20Z. Sept. 2021. URL: https://github.com/mwhittaker/bipartisan_paxos/blob/cbd99cc735215d18c163dc41cb0a05edcb55437d/epaxos_bugs/epaxos_dependency_bug.pdf (visited on 2021-09-16).

[Whi+21] Michael Whittaker et al. "Matchmaker Paxos: A Reconfigurable Consensus Protocol". en. In: *Journal of Systems Research* (2021), p. 22.

[Zav12] Pamela Zave. "Using lightweight modeling to understand chord". en. In: *ACM SIGCOMM Computer Communication Review* 42.2 (Mar. 2012), pp. 49–57. ISSN: 0146-4833. DOI: 10.1145/2185376.2185383. URL: https://dl.acm.org/doi/10.1145/2185376.2185383 (visited on 2021-09-06).

[Zav17] Pamela Zave. "Reasoning About Identifier Spaces: How to Make Chord Correct". In: *IEEE Transactions on Software Engineering* 43.12 (Dec. 2017). Conference Name: IEEE Transactions on Software Engineering, pp. 1144–1156. ISSN: 1939-3520. DOI: 10.1109/TSE.2017.2655056.