Research Plan for CSE3000 Research Project

Using Weighted Voting to Accelerate Blockchain Consensus

Diana Micloiu

April 28, 2024

Background of the research

Blockchain consensus algorithms lay at the basis of distributed ledger technologies. Classified into permissioned and permissionless, they distinguish themselves by either limiting participation to a predetermined set of nodes or allowing anyone to join.

Initially, permissionless systems gained popularity with the seminal Nakamoto consensus relying on proof-of-work. However, its significant impact on energy consumption revealed the system's limitations and urged researchers to look for alternative consensus algorithms. In contrast, permissioned systems bring higher efficiency in terms of throughput, latency and finality. Thus, the focus shifted towards finding ways to optimise their performance. Strategies such as system size reduction and leader selection mechanisms have been explored to enhance scalability and resilience. Notably, the first-generation Practical Byzantine Fault Tolerance (PBFT) algorithm [4] has been a focal point of research in permissioned systems.

The idea of changing the consensus algorithms to use a weight metric as voting power occurred. Using this kind of mechanism, WHEAT [13] achieved higher performance for state machine replication in geographically distributed settings. Next, researchers put together an enhanced version of PBFT, namely BFT-SMaRt [2] and the weighted voting mechanism behind WHEAT to create AWARE [1], a deterministic, self-monitoring and self-optimising algorithm for optimising the latency of the blockchain.

So far, research on the benefits of weighted voting has only studied first-generation algorithms such as PBFT. This research project seeks to address this gap in the literature by investigating the impact of weighted voting on streamlined consensus algorithms, such as the Hotstuff[15] family ones. By extending the principles established by AWARE to newer generations of consensus mechanisms and evaluating their robustness in the face of node failures, this study aims to contribute to the broader understanding of weighted voting's efficacy in accelerating consensus and fostering distributed trust in blockchain systems.

Research Question

The research question that the project aims to answer is:

How can weighted voting improve the performance of streamlined algorithms (2nd generation[15])?

Over the years, research has been done on how consensus can be improved in blockchain consensus algorithms, with asynchronous Byzantine fault tolerance becoming the most popular approach. However, the main disadvantages, namely that they are slow and expensive to run, supported the research of the streamlined and cluster-based algorithms. In this sense, researchers from VMware Research developed Hotstuff [15], a protocol that achieves partial synchrony by using leader rotation on each command to shift the communication burden from the leader. By using a star-type communication network, the protocol achieves linear message complexity and faster response times.

Additionally, current research is being conducted to optimise the features of streamlined algorithms[11], such as Pili[7], Pala[6], Streamlet [5], Tendermint[3] and, previously mentioned, Hotstuff[15]. For instance, DAMYSUS [8] improves on top of Hotstuff by reducing the number of communication phases using trusted components, thus achieving better performance.

Going back to the idea of weighted voting and analysing the major improvements that the development of AWARE showcases, the question of how weighted voting could impact latency optimisation in streamlined protocols arises. Given the extensive research that has been done over the years on the Hotstuff family protocols, together with the weighted voting mechanism presented in papers such as WHEAT and AWARE, the research question posed by the project seems feasible in the provided time frame. That is mainly due to the large availability of resources that can support the research on the impact of assigning some importance measure in the voting process.

The project can be structured by answering the following sub-questions, which ultimately comprise the whole idea behind the research being conducted.

R1: What is weighted voting in the context of streamlined consensus algorithms?

This aims to emulate the concept of weighted voting in the steps of streamlined algorithms. Namely, the focus would be on how weights are assigned and how they influence the decision-making process during consensus.

R2: How can weighted voting be applied to a specific 2nd generation algorithm?

This sub-question shifts the focus to one streamlined algorithm (possibly Hotstuff or DAMYSUS) to discover how weighted voting can be integrated into its design. It implies modifying protocol in order to support weighted voting in the consensus step.

R3: What are the challenges and limitations imposed by implementing weighted voting in streamlined algorithms?

Implementing weighted voting in streamlined algorithms comes with some possible challenges, such as node coordination. An analysis should be conducted to report the potential biases introduced by weight assignment and elaborate on the challenges of implementation complexity.

R4: How do empirical or simulation experiments support the effectiveness of weighted voting in 2nd generation algorithms?

For this sub-question, the algorithm the paper focuses on should be simulated on different inputs and in various settings to analyse performance on latency, throughput and finality. This would comprise the *Evaluation* section of the paper.

Method

The **method of research** used in this project will be **experiments conducted by writing code** that emulates the desired behaviour and uses a benchmark to analyse its performance compared with the original version. To elaborate, I will break apart each sub-question and describe the steps needed to achieve the expected outcomes.

Firstly, in **week 1**, the focus will be on a literature review on weighted voting to better understand the concepts of streamlined algorithms and weighted voting. This will ultimately represent the basis for the first sub-question. Transitioning towards **week 2**, all the information regarding background and related work should be summarised in the corresponding sections of the research plan. Moreover, building upon the required knowledge, I will develop possible optimisation methods to integrate weighted voting in the context of 2nd generation algorithms. In the end, I will have a clear idea of what could be a feasible approach, hence answering the first sub-question.

Next, by looking over the multiple choices of streamlined algorithms, I will stick with one to apply the weighted voting optimisation. This decision will be made by considering implementation complexity, time limitation imposed by the 10-week frame of the research project and advantages/disadvantages of each protocol.

Afterwards, there are two possibilities for moving forward:

1. Simplified implementation using Python. Write a simplified, local version of the blockchain algorithm and tweak it to use weighted voting. Compare the performance of the two implementations to observe the advantages/disadvantages of using this mechanism on streamlined algorithms.

2. Fork the original implementation of the algorithm and adapt it. This is the more advanced approach since it requires writing the modified voting mechanism in Rust (if we are talking about Hotstuff

[12]) or C++ (in the case of DAMYSUS [14]). Moreover, it requires a better understanding of the code bases that support the corresponding research papers. Plus, the inherent use of libraries for networking (such as Salticidae[9]) and Docker containers for running experiments must be explored.

In the period of **week 3 - week 6**, one of the implementation approaches described above will be completed and comprised in the *Contribution* section of the paper. In this way, the 2nd sub-question will be concluded. Moreover, in parallel, the simulations and experiments will be conducted to gather data about the new algorithm's performance compared with the original one. The goal is to execute them in various conditions such that the impact of different settings would reveal possible edge cases or limitations of the idea of improvement, thus answering the 3rd sub-question. By using Python scripts, the results will be showcased in the form of tables and graphs to point out the changes in behaviour determined by implementing weighted voting on top of the streamlined blockchain algorithm. This step, together with the interpretation of results and data analysis, would enable identifying the performance improvements, which will be described in the *Experimental setup and Result sections*.

Having the results, the next step (**week 7 - week 8**) would be analysing the results of a fellow peer and comparing their weighting voting approach to the one implemented to highlight possible advantages/disadvantages. Moreover, having in mind the best research practices, ethics and responsible research will be considered.

Ultimately (**week 9**), the research will come together in the form of a paper. The report should be well-structured, offering a clear outline of the research methodology, results, conclusions, responsible research techniques and any future work recommendations.

Planning of the research project

Note that the "Project Meeting" is a weekly meeting together with the supervisor, responsible professor and peers to discuss project advancements and provide/receive feedback.

Week	Task	Estimated Time
Week 1	Kick-off lecture	1h
	Assignment: Research plan - Draft	2h
	Project Meeting	1h
	Identify and add to the reference list the 10 most relevant research papers	3h
	Study the AWARE paper [1] and conduct literature review ([4], [2], [10] and [13])	20h
	Study the streamlined algorithms: Hotstuff[15] and DAMYSUS[8]	6h
	Look over the code base of Hotstuff [15] and DAMYSUS[8]	2h
	Assignment: Research plan final	5h
Week 2	**Lecture:** Session Responsible Research	1h
	Project Meeting	1h
	Complete "Background and Related work" section of the paper	5h
	Complete "Problem description" section of the paper	5h
	Define what is weighted voting in the context of streamlined algorithms	20h
	Try validating the idea with a toy example	8h
Week 3	**Lecture:** Session Responsible Research	1h
	Project Meeting	1h
	Apply weighted voting on a specific streamlined algorithm and its chained version	20h
	Identify relevant performance metrics to be observed when running the experiments	5h
	Complete the "Contribution" section of the paper	6h
	Assignment (ACS 1)	2h

Week	Task	Estimated Time
Week 4	**Session:** ACS 1 (Paper)	1h
	Project Meeting	1h
	Write draft of the "Abstract" section	2h
	Write draft of the "Introduction" section	2h
	Get a working version of the algorithm	10h
	Start the experiment phase to gather data	10h
	Data analysis on results of the simulations	5h
	Assignment (ACS 2a, 2b)	2h
	Session: ACS 2 (Poster)	1h
	Prepare poster and midterm presentation	8h
Week 5	Midterm presentation	2h
	Project Meeting	1h
	Incorporate feedback from the midterm presentation	5h
	Write the "Responsible Research" (Ethics) section	5h
	Tweak the algorithm and perform simulations in various settings	25h
	Assignment (ACS 3)	1h
Week 6	**Session:** ACS 3 (Paper)	1h
	Project Meeting	1h
	Finalise experiment phase	10h
	Gather all data in form of graphical illustrations	10h
	Get feedback on results and incorporate it in the paper	5h
	Finalise the "Abstract" and "Introduction" sections	3h
	Write the "Experimental setup" and "Results" sections	8h
Week 7	Check the paper and improve sections	15h
	Assignment: Paper Draft v1	1h
	Review paper of fellow student and provide feedback	5h
	Project Meeting	1h
	Assignment: Peer Review	1h
	Incorporate peer review feedback	5h
	Write "Discussion" section based on the results presented in the reviewed paper	5h
Week 8	Complete "Conclusion & Limitations" and "Future work" sections	10h
	Revise "Abstract" section to make it comprehensive enough, concise and precise	2h
	Project Meeting	1h
	Discuss possible improvement	1h
	Review paper citations to ensure correctness and completeness	5h
	Assignment: Paper draft 2	4h
	Additional literature review and revise paper correctness	8h
Week 9	Reiterate a sample of the experiments to ensure reproducibility	10h
	Project Meeting	1h
	Clean up and document code	4h
	Properly document experiments and prepare as deliverable	5h
	Incorporate feedback received	5h
	Proofread and submit final version the paper	5h
	Prepare final presentation and poster	10h
Week 10	Rehearse final presentation	5h
	Final proofread of the paper	5h
	Project Meeting	1h
	Session: ACS (Poster 4)	1h
	Provide deliverables on TUDelft repository	1h
	Attend final presentation	2h

References

[1] Christian Berger et al. "AWARE: Adaptive wide-area replication for fast and resilient Byzantine consensus". In: *IEEE Transactions on Dependable and Secure Computing* 19.3 (2020), pp. 1605–1620.

[2] Alysson Bessani, João Sousa, and Eduardo EP Alchieri. "State machine replication for the masses with BFT-SMART". In: *2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*. IEEE. 2014, pp. 355–362.

[3] Ethan Buchman, Jae Kwon, and Zarko Milosevic. "The latest gossip on BFT consensus". In: *arXiv preprint arXiv:1807.04938* (2018).

[4] Miguel Castro, Barbara Liskov, et al. "Practical byzantine fault tolerance". In: *OsDI*. Vol. 99. 1999. 1999, pp. 173–186.

[5] Benjamin Y Chan and Elaine Shi. "Streamlet: Textbook streamlined blockchains". In: *Proceedings of the 2nd ACM Conference on Advances in Financial Technologies*. 2020, pp. 1–11.

[6] TH Hubert Chan, Rafael Pass, and Elaine Shi. "Pala: A simple partially synchronous blockchain". In: *Cryptology ePrint Archive* (2018).

[7] TH Hubert Chan, Rafael Pass, and Elaine Shi. "Pili: An extremely simple synchronous blockchain". In: *Cryptology ePrint Archive* (2018).

[8] Jérémie Decouchant et al. "DAMYSUS: streamlined BFT consensus leveraging trusted components". In: *Proceedings of the Seventeenth European Conference on Computer Systems*. 2022, pp. 1–16.

[9] Determinant. *Salticidae*. `https://github.com/Determinant/salticidae`.

[10] Fred B Schneider. "Implementing fault-tolerant services using the state machine approach: A tutorial". In: *ACM Computing Surveys (CSUR)* 22.4 (1990), pp. 299–319.

[11] Elaine Shi. "Streamlined blockchains: A simple and elegant approach (a tutorial and survey)". In: *Advances in Cryptology–ASIACRYPT 2019: 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8–12, 2019, Proceedings, Part I 25*. Springer. 2019, pp. 3–17.

[12] A. Sonnino et al. *HotStuff*. `https://github.com/asonnino/hotstuff`.

[13] João Sousa and Alysson Bessani. "Separating the WHEAT from the chaff: An empirical design for geo-replicated state machines". In: *2015 IEEE 34th Symposium on Reliable Distributed Systems (SRDS)*. IEEE. 2015, pp. 146–155.

[14] V. Vrahli et al. *Damysus*. `https://github.com/vrahli/damysus`.

[15] Maofan Yin et al. "HotStuff: BFT consensus with linearity and responsiveness". In: *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing*. 2019, pp. 347–356.