Short Paper: Ballot Secrecy for Liquid Democracy

Mahdi Nejadgholi, Nan Yang, and Jeremy Clark(✉)

Concordia University, Montreal, Canada
j.clark@concordia.ca

Abstract. Certain advances in election technology, such as online voting, promise to reduce the administrative overhead of running an election. This has breathed new life into the idea of direct democracy, where voters play a more active role in setting legislation. However it is anticipated that a steady stream of referendums would generate voter fatigue. To combat this fatigue, voters could be allowed to delegate their votes to other (more knowledgeable) voters. This idea is old but has been recently reinvented under the name liquid democracy. In this paper, we consider how ballot secrecy should be defined for liquid democracy. We first show that a natural definition of full secrecy leads to several undesirable outcomes. We then show that these are very difficult to address without enabling voter coercion and vote buying. The purpose of the paper is not to affirm liquid democracy; rather, it is to raise awareness of unseen complexity hiding under our initial presumption that liquid democracy could effortlessly support a secret ballot.

1 Introductory Remarks

A *liquid democracy* voting system allows each voter the option of delegating their vote to another voter. This helps offload the burden of informing yourself about every issue and position at stake. While it can be used in any election, it is well-suited in direct forms of democracy, where many or all issues are put to a referendum. In liquid democracy, if Alice and Bob delegate to Carol, Carol's vote carries the weight of three voters. The defining feature of liquid democracy is that Carol can in turn delegate to David. If David votes directly, he can effectively cast four ballots (his own, Carol's, Alice's, and Bob's) in addition to whatever other delegations he has received. In this paper, we note several challenges in defining ballot secrecy for liquid democracy.

2 Preliminaries

2.1 Systems of Democracy

Consider a nation-state where legislation is set through a process of voting on bills. A common system is *representative democracy* where citizens (or legal residents) elect a parliamentary member to represent their region and vote on bills.

M. Bernhard et al. (Eds.): FC 2021 Workshops, LNCS 12676, pp. 306–314, 2021.
https://doi.org/10.1007/978-3-662-63958-0_26

In a system based on *direct democracy*, citizens would vote on the bills themselves. A critical issue with direct democracy is voter fatigue. Given the large number and wide variety of bills, it is difficult for voters to inform themselves on every issue and cast meaningful votes.

One solution to voter fatigue is allowing voters to delegate their votes to someone else (a proxy). Proxy voting is a general term that includes cases where: (a) the voter directs the proxy how to vote, and (b) the voter lets the proxy decide how to vote. Clearly, voter fatigue is only addressed by (b) and not (a). Systems of type (b) are called *delegative democracy*. While forms of delegative democracy have been discussed for centuries, *liquid democracy* is a re-branding of it that has become popular since the late 2000s [3]. We resist calling liquid democracy a 'silicon valley' invention because its early popularity stemmed from Europe, however it has been embraced and amplified by a similar demographic of young technologists. These technologists already advocate for more direct democracy. Complimentary tools include: online voting, which reduces the friction of conducting frequent elections; random sample voting, a competing solution to voter fatigue [1,6]; and blockchain technology (like liquid democracy, proposed in the 2000s), a technological platform for decentralized computation. Liquid democracy is suggested as a governance mechanism for decentralized organizations in the Ethereum whitepaper [4]. Note that experts argue that blockchain offers more hype than merit in the specific case of voting technology for governmental elections [21,22].

Liquid democracy's pseudonymous inventor Sayke maintains that liquid democracy is distinct from delegative democracy. For example, in liquid democracy, a voter could delegate to a set of other voters and have the system cast a ballot in favour of the plurality of opinions [24]. This has been analyzed as *statement voting* in the literature [27]. However all notable software implementations of liquid democracy are limited to delegations only, and that is what we study in this paper.

2.2 Past Experiments and Uses

The open-source *Liquid Feedback* [2] system is likely the best-known Liquid Democracy implementation. It was used by Germany's Pirate Party by ~15K members in 2013 [18]. It is still used by Italy's *MoVimento 5 Stelle (M5S)*; a party which received 25% of Italy's parliament seats in 2013. Liquid Feedback is an open vote system and does not implement a secret ballot. Further, despite offering the feature, data from its main trials find less than 5% of voters actually delegated their votes in practice [23].

Google Votes was an experimental implementation of liquid democracy used internally by Google from 2012–2015 through its Google+ social network [15]. The uses were relatively non-significant (*e.g.,* decisions on the Mountain View Microkitchen food fair in California, or the GoogleServe logo). Like Liquid Feedback, there is no ballot secrecy in Google Votes (in fact, voters can do 'biased sharing' by advertising their vote and obtaining delegations on it). Only 3.6% of voters delegated.

Low profile examples include *LiquidFriesland* for voting on municipal initiatives (less than 30 average voters for each initiative) [9], and *Civicracy* whose pilot study for a school council fell through [14]. For more abandoned or unfinished liquid democracy projects, see Paulin's retrospective [23].

Academic contributions from the computer science (and security) community include a coercion-resistant proxy voting scheme [20] that issues voters fake credentials. A second paper introduces statement voting, an end-to-end verifiable voting system that implements a generalization of liquid democracy but does not address coercion [27]. Both of these designs implement *full ballot secrecy* (all votes and delegations are protected). Our paper illustrates several ways in which this level of secrecy is undesirable; unfortunately, relaxing it tends to create coercion issues. This idea is extended for governing decentralized applications running on blockchain technologies like Ethereum [26]. Another blockchain-based solution proposes an efficient algorithm for self-tallying, cycle-resistant liquid democracy for Ethereum, however, the authors do not consider vote secrecy [10]. Another work considers increasing participation in an open liquid democracy system without ballot secrecy [19].

Last, a series of papers from Ford (the most recent and representative work is [11]) dives deeply into different design parameters of liquid democracy, and critically analyzes them. The Ford paper considers many different aspects, technical and social, while our paper does a deep dive on one specific issue: ballot secrecy. Even so, we identify some specific overlaps in Subsect. 4.3.

3 Assumptions

We are interested in what liquid democracy would look like for a governmental election. Most of the systems mentioned in Subsect. 2.2 were developed for transparent, open vote elections with a rolling tally (*i.e.,* realtime updates). It is difficult to imagine all the election law changes that would pave the road for liquid democracy, but we assume two basic principles of elections would still be required: ballot secrecy, and an announcement of the final result only at the conclusion of the election. We also make the following assumptions about a hypothetical liquid democracy system for governmental elections and referendums:

- **Referendums.** Liquid democracy could be used for either elections of individuals or referendums on issues. For simplicity, we will refer to referendums throughout the rest of this paper but it is not without loss of generality.
- **Online.** We assume the referendum is conducted with online voting. Online voting is incredibly problematic from a security perspective, but we will assume that the system has end-to-end verification (E2E), mitigates the untrusted platform problem (*cf.* [5,25]), and provides some basic coercion-resistance (*cf.* [7,8,12,17])—however we will revisit the degree to which such coercion resistance protection is even possible.
- **Phases.** We assume the referendum is conducted in two phases. In phase 1, voters can delegate their votes, change their delegation, or remove their delegation. Phase 1 might happen over the course of weeks or months depending

Table 1. Issues with a full secrecy ballot and proposed features to solve them. In this paper, we evaluate the privacy consequences of these features.

Issue	Proposed feature
Delegation cycle	Real-time cycle detection
Unexpected delegations	Expose incoming weight
Unaccountable or non-responsive delegates	Expose voting action

on the lead-up time to the referendum. In phase 2, voters can cast their ballots, and can no longer delegate or change their delegation. A voter that has delegated in phase 1 can still vote in phase 2 and this action overrides their delegation. We will refer to phase 1 as the delegation phase and phase 2 as the voting phase.

- **Multi-Referendums.** In the case of multiple concurrent referendums, we assume that the delegation phase is on-going but for each specific referendum, the delegation status will freeze at a certain announced time as that referendum moves into the voting stage.

4 Ballot Secrecy for Liquid Democracy

Full Secrecy and Its Shortcomings.
The most natural definition of ballot secrecy for a liquid democracy election is to hide everything except the final tally [20,27]. This includes all votes and all delegations. This approach could however lead to one of three unintended consequences, recapped in Table 1 and explored in each of the following sections of the paper.

4.1 Delegation Cycles

A delegation cycle occurs when Alice delegates to Bob and Bob delegates to Alice (or any longer chain that cycles back to the initial voter). If ballots are secret, there is no directly way for Alice and Bob to discover the cycle within the system. It is important to note that delegation cycles can form without any voter behaving maliciously. If they do not discover the cycle out-of-band, their votes will not be counted. This issue is mentioned without solution by Zhang and Zhou[1] who use full ballot secrecy for their cryptographic design [27].

A straight-forward solution is to offer an 'oracle' in the design that would either (a) answer any voter's query of whether their own vote is in a cycle or not, or (b) prevent a voter from delegating to another voter if that delegation forms a cycle by displaying a failure message to the voter. By the term 'oracle,' we assume this information would be made available only to the voter (or more

[1] Authors' note: at our suggestion.

precisely, would only be convincing to the voter and could not be convincingly shown to a coercer; *cf.* designated verifier signatures and proofs [16]).

Consequences for Ballot Secrecy. The security issue with either of these oracles is that their inclusion breaks the *coercion-resistance* of the system. Assume that Mallory, a coercer or vote buyer, uses undue influence to convince Alice to delegate to her. She can check compliance (at any time) by delegating her own vote to Alice and confirming that it forms a cycle. If it does not, Alice did not comply. Alice can try to rush Mallory and undelegate at the last minute, or overwrite the delegation by casting a ballot. In both cases, the coercion evasion strategy is akin to a voting system that lets you vote as many times as you want (revoting or multiple cast). The same simple coercion techniques, such as Mallory retaining Alice's voter ID card, can also thwart these defences in liquid democracy.

Potential Mitigations. A variant on the *fake credential* design pattern—used in many coercion-resistant voting systems [7,8,17], including proxy voting [20]—could be applied here. In these systems, if Alice is coerced, she can make up a fake credential (or have prepared one in advance) to give to the adversary that operates exactly like her real voting credential. During tallying, all votes cast with fake credentials are obliviously removed. This design pattern can work for liquid democracy except that Alice needs to create a fake *identity* or *persona* that can create delegations that are seen by the delegates and are indistinguishable from real identities. One straightforward composition with existing protocols is to consider public keys as identities. Alice can convincingly lie about what public key she registered as her real identity. She use fake keys to cast fake votes or create fake delegations. While this solves the delegation cycle problem by providing a coercion resistance mechanism that is not thwarted by the introduction of a cycle detection oracle, two issues remain: (1) how to cryptographically realize the cycle detection oracle (we do not solve that problem here; this paper is about ideal functionalities), and (2) it does not solve the two additional issues that follow.

4.2 Unexpected Delegations

It seems natural that Alice would like to know if others have delegated to her. For example, knowing that she has a large number of delegations could increase her efforts in informing herself and completing the task of voting. Very popular delegates could find themselves the target of individualized attacks (cybersecurity or otherwise) to modify their vote or to prevent them from voting. First, it seems sensible that serving as a delegate should be opt-in, and that voters who do not want delegations can remain as default voters. A second design feature could offer to each voter an oracle service that reports the number of voters who have delegated to them (*incoming weight oracle*).

Consequences for Ballot Secrecy. Like the cycle-detection oracle, the incoming weight oracle can be used for coercion. Consider Mallory influencing Alice to delegate her vote to her. She checks the incoming weight oracle before and after the purported delegation, which should increase by one delegation, to ensure Alice's compliance.

Potential Mitigations. Weights could be given in ranges and/or with noise added to thwart coercion, however this requires further attention. For instance, if a single delegation (*e.g.,* Alice's delegation is Mallory's 100th) moves the incoming weight from one range (*e.g.,* 10–99) to the next highest range (*e.g.,* 100–999), it can be used for coercion. While noise can provide provable 'differential privacy' when used once, liquid democracy allows the coercer to dynamically add/remove weights and re-query the oracle as many times as she likes, taking statistics over all the results.

The fake persona design pattern suggested for cycle detection can be used to thwart coercion, however it defeats the original goal of providing Alice with a sense of her ballot's weight—while a number can be displayed, there is telling if it consists of real or fake delegations. Delegates can be easily misled in terms of the number of delegations they are actually receiving. Were the design to use both noisy counts and fake personas, providing the coercer with a mechanism to add/remove any number of fake delegations makes it more difficult to disguise the count.

4.3 Unaccountable Delegates

It could be argued that when a voter delegates, there is absolute trust in the delegate. With a secret ballot, if the delegate fails to vote, or purposefully misleads its delegators as to how it will vote, there is no way to hold them accountable. This issue and its consequence for coercion-resistance is already explored by Ford [11], however we include it for completeness.

A *voting action oracle* could be introduced to let voters see the full delegation path to the final vote (Google Votes [15]). Or more simply, the design could make all delegate votes (and further delegations) public information along with the tally (Ford [11]).

Consequences for Ballot Secrecy. As pointed out by Ford, adding accountability harms coercion resistance. If Mallory apply undue influence on Alice, she can have Alice opt-in as a delegate, delegate her own vote to Alice, instruct Alice on how to vote, and then use the voting action oracle to learn if Alice complied.

Potential Mitigations. In the fake persona design pattern, Alice could create a fake identity and give it to Mallory for delegation. First, note that this situation is different from the earlier coercion example in cycle detection. There, Alice was

a voter and Mallory was the delegate. Here Alice is the delegate and Mallory is the voter. If fake personas can be created by voters and delegates equally, then the coercion issue here is solved.

However there are good reasons why voters might be allowed to create fake identities, but once a voter opts into becoming a delegate, they can no longer create fake identities. Consider an attack where Mallory becomes a high profile celebrity for having a certain political ideology. In reality, she actually holds a different ideology. If she amasses a large number of delegations from supporters of her fake public ideology, she could decide to cast all the votes in favour of her real ideology, or simply not vote at all—both actions harm the support of her fake public ideology. However if a voting action oracle is provided, both actions will get her caught.

Instead, she could give out a fake identity for other voters to delegate to. She could vote for her fake public ideology using this fake identity to satisfy the voters. In the end, all the votes would be canceled during tallying but the cancelation is done without revealing which votes are being cancelled—therefore, she could avoid getting caught and do this attack indefinitely. For this reason, we consider this issue an open research problem.

5 Concluding Remarks

The problem of defining ballot secrecy for liquid democracy presents a set of four desirable properties with no obvious way of achieving them all: (1) coercion resistance, (2) no cycles, (3) knowledge of incoming weight, and (4) accountability for delegates. Full ballot secrecy alone does not provide any of these [27]. Fake personas have been applied to achieve (1) [20], however we show it can provide both (1) and (2) with a cycle detection oracle.

An alternative to liquid democracy comes close to providing all of (2)–(4). The idea is to restructure the election into multiple rounds. In the first round, voters can only vote directly for issues. In the subsequent rounds, voters can delegate to any voter who has already voted in a prior round (names will be made public) or they can vote directly. This system cannot have cycles by definition (which would require a voter to delegate to someone who has not voted yet). It sidesteps the impact of a delegate receiving a large number of delegations as the delegations are collected only after a ballot is already cast. It is impossible to delegate to someone who will not vote, however it is still possible to delegate to someone who will vote differently from their public political views, leaving the voter with no knowledge or recourse. Finally, if (1) is achieved using fake personas, a malicious delegate could collect delegations using a fake persona knowing these votes will all be discarded during tallying.

A prerequisite to designing an end-to-end verifiable voting system is deciding how the system should operate; more formally captured by defining its ideal functionality. Debates have been had over ideal functionalities for simple first-past-the-post schemes [13] (*e.g.,* should only the winner be declared, or should the final tally of votes be declared?). In this paper, we informally discuss what

the ideal functionality of a liquid democracy system should be, particularly as it relates to ballot secrecy. We hope to have demonstrated that it is not a simple or obvious choice, but rather it is an important research question to consider before proposing new designs in this space.

Acknowledgements. We thank the reviewers who helped to improve our paper. J. Clark acknowledges support for this research project from (i) the National Sciences and Engineering Research Council (NSERC), Raymond Chabot Grant Thornton, and Catallaxy Industrial Research Chair in Blockchain Technologies, and (ii) NSERC through a Discovery Grant.

References

1. Basin, D., Radomirovic, S., Schmid, L.: Alethea: A provably secure random sample voting protocol. In: IEEE CSF (2018)
2. Behrens, J., Kistner, A., Nitsche, A., Swierczek, B.: The Principles of LiquidFeedback. Interaktive Demokratie, Berlin (2014)
3. Behrens, J.: The origins of liquid democracy. Liquid Democracy J. **5** (2017)
4. Buterin, V.: Ethereum whitepaper. Technical report, Online (2013)
5. Chaum, D.: SureVote: technical overview. In: WOTE (2001)
6. Chaum, D.: Random-sample voting (2012). Online
7. Clark, J., Hengartner, U.: Selections: Internet voting with over-the-shoulder coercion-resistance. In: FC (2011)
8. Clarkson, M.R., Chong, S., Myers, A.C.: Civitas: toward a secure voting system. In: IEEE Symposium on Security and Privacy, pp. 354–368 (2008)
9. Eisel, S.: Liquidfriesland - ein gescheitertes experiment. https://internetunddemokratie.wordpress.com/2014/05/22/liquidfriesland-ein-gescheitertes-experiment/. Accessed May 2014
10. Fan, X., Li, P., Zeng, Y., Zhou, X.: Implement liquid democracy on Ethereum: a fast algorithm for realtime self-tally voting system. CoRR abs/1911.08774 (2019). http://arxiv.org/abs/1911.08774
11. Ford, B.: A liquid perspective on democratic choice. arXiv:2003.12393 [cs.CY] (2018)
12. Grewal, G.S., Ryan, M.D., Bursuc, S., Ryan, P.Y.A.: Caveat coercitor: coercion-evidence in electronic voting. In: IEEE Symposium on Security and Privacy (2013)
13. Groth, J.: Evaluating security of voting schemes in the universal composability framework. In: Jakobsson, M., Yung, M., Zhou, J. (eds.) ACNS 2004. LNCS, vol. 3089, pp. 46–60. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24852-1_4
14. Hainisch, R., Paulin, A.: Civicracy: establishing a competent and responsible council of representatives based on liquid democracy. In: 2016 Conference for E-Democracy and Open Government (CeDEM), pp. 10–16 (2016). https://doi.org/10.1109/CeDEM.2016.27
15. Hardt, S., Lopes, L.R.: Google votes: a liquid democracy experiment on a corporate social network (2015)
16. Jakobsson, M., Sako, K., Impagliazzo, R.: Designated verifier proofs and their applications. In: Maurer, U. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 143–154. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-68339-9_13

17. Juels, A., Catalano, D., Jakobsson, M.: Coercion-resistant electronic elections. In: Chaum, D., et al. (eds.) Towards Trustworthy Elections. LNCS, vol. 6000, pp. 37–63. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-12980-3_2
18. Kling, C.C., Kunegis, J., Hartmann, H., Strohmaier, M., Staab, S.: Voting behaviour and power in online democracy: a study of LiquidFeedback in Germany's pirate party. http://arxiv.org/abs/1503.07723
19. Kotsialou, G., Riley, L.: Incentivising participation in liquid democracy with breadth-first delegation. arXiv:1811.03710 [cs, econ] (February 2019)
20. Kulyk, O., Neumann, S., Marky, K., Budurushi, J., Volkamer, M.: Coercion-resistant proxy voting. Comput. Secur. **71**, 88–99 (2017)
21. Nasser, Y., Okoye, C., Clark, J., Ryan, P.Y.A.: Blockchains and voting: somewhere between hype and a panacea (2017). Online
22. Park, S., Specter, M., Narula, N., Rivest, R.L.: Going from bad to worse: from internet voting to blockchain voting (2020). Online
23. Paulin, A.: An overview of ten years of liquid democracy research. In: The 21st Annual International Conference on Digital Government Research (2020)
24. Sayke: Liquid democracy is not delegative democracy (2006). Blog post
25. Zagórski, F., Carback, R., Chaum, D., Clark, J., Essex, A., Vora, P.L.: Remotegrity: design and use of an end-to-end verifiable remote voting system. In: ACNS (2013)
26. Zhang, B., Oliynykov, R., Balogun, H.: A treasury system for cryptocurrencies: enabling better collaborative intelligence. In: 26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, 24–27 February 2019. The Internet Society (2019)
27. Zhang, B., Zhou, H.-S.: Statement voting. In: Goldberg, I., Moore, T. (eds.) FC 2019. LNCS, vol. 11598, pp. 667–685. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-32101-7_38