Quantum Voting Protocols

Monika Getsova

December 2019

The inspiration for this project topic arose while watching a woman picking the lock of a voting machine and messing with the hardware for a few minutes after which she gained administrative privileges on said machine. Over the time that voting machines have been used, reports of hacks (including remote hacks) have increased and most experts are now advocating for the use of paper ballots [2][3]. However, paper ballots come with many issues as well. In a purely paper based system, voters must submit their ballots to some central authority to be counted such that there is no way to prove to any voter that their vote was counted and that it was not altered. Relying on people to count ballots is absurd in the modern era, but machines that scan or otherwise utilize paper ballots have also been found to be unreliable (ex. the famous hanging chad problem).

As a result, the use of a blockchain system has become a popular idea in recent years since it would theoretically address the problems of paper ballots described above. While this is still a controversial topic and most security experts warn against adopting the technology, some localities including West Virginia and the entire country of Sierra Leone (both in 2018) have held elections on a blockchain[9][10][3]. Regardless of whether one believes that the widespread adoption of such a voting protocol is an inevitability or not, the benefits it offers, including the ability for each voter to verify that their vote was counted correctly, are of great importance to a democratic society and thus research into the development of secure voting protocols that offer at least some of the benefits of a publicly distributed ledger system will likely never go away. For the purpose of the following discussion, I will assume that the adoption of a blockchain voting protocol on classical computers is an inevitability. However, if that is not the case, secure quantum voting protocols still offer socially important benefits that are lacking in the current status quo and the study of such protocols appears to have taken off in the past few years.

In [12], Sun lists the following properties that an ideal voting protocol should have:

- Anonymity: No one is able to find out how any other voter voted.
- Binding: Ballots cannot be changed after submission.

- Non-reusability: It is not possible for a dishonest voter to vote more than once.
- Verifiability: Each voter has the ability to obtain proof that their vote was counted and not altered.
- Eligibility: Only eligible voters can participate.
- Fairness: It is not possible to count any subset of the submitted votes before all votes are submitted and counted.
- Self-tallying: Anyone who is interested in the voting result can tally the votes themselves. Or alternatively, any participant in the election has a means to prove to themselves that the outcome of the election is being reported accurately.

A classical voting protocol utilizing a blockchain is able to meet most or all of these requirements (generally, it is all but in [4] the binding condition is sacrificed with the presumption that allowing voters to change their votes will help mitigate voter coercion). The anonymity and security of a voting protocol on a classical blockchain is ensured by the use of digital signatures and cryptographic hash functions[4][1][7]. The digital signatures on most blockchains are generated using the elliptic curve signature scheme which ensures security due to the computational complexity of computing discrete logarithms on a classical computer. However, it is estimated that a quantum computer may be able to do this within a reasonable time frame by 2027. It is also theorized that Grover's search algorithm may be able to perform the proof of work necessary to add a new block to the ledger 100x times faster than a classical computer in the coming decades [1]. This would allow a quantum adversary to succeed in gaining control of the blockchain via a 51% attack, essentially allowing the adversary to decide the outcome of an election [7]. Thus, in a post-quantum world, one needs quantum voting protocols!

1 Wang's Protocol

[14] For m candidates we have a basis $|j\rangle_C$ where $j = 0, 1, ..., m-1$. $|j'\rangle_F$ is the Fourier basis defined by

$$|j'\rangle_F = \frac{1}{\sqrt{m}} \sum_{k=0}^{m-1} exp(\frac{2\pi ijk}{m}) |k\rangle_c$$

. For n voters, we define a state

$$|\chi_n\rangle = \frac{1}{m^{\frac{n-1}{2}}} \sum |j_0\rangle_C |j_1\rangle_C ... |j_{n-1}\rangle_C = \frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} |j'\rangle_F^{\otimes n}$$

where the first sum is over all possible states where $(j_0+j_1+...+j_{n-1})\text{mod}m = 0$. $|\chi_n\rangle$ has the property that when each particle is measured in the computational basis, the sum of all outcomes $\text{mod}m = 0$ and in the Fourier basis the measurement outcomes for all particles are the same. Secret ballots are distributed to each voter $V_i, i = 0, 1, ..., n-1$ as follows:

1. V_0 prepares $n + n\delta_0$ states $|\chi_n\rangle$ and sends the i^{th} particle of each state to V_i st. a particle matrix is generated of dim $n + n\delta_0$ x n where the element $p_{i,j}$ is the j^{th} entangled particle in the i^{th} copy of $|\chi_n\rangle$ which voter V_i has access to.

2. V_0 chooses δ_0 of his $n + n\delta_0$ uniformly at random. V_0 also chooses between the computational and Fourier measurement basis uniformly at random. V_0 tells the chosen row indices and basis to all other V_i

3. Voters $V_i, (i \neq 0)$ all perform the same measurements as V_0 and send their measurement results to V_0

4. V_0 now knows the state of all particles in rows with the chosen indices. He checks that they satisfy the conditions for $|\chi_n\rangle$ and ABORT the protocol if not.

5. Each other voter performs the same test procedure: if all tests pass, the $n\delta_0$ test rows are discarded st. each V_i is left with n particles which they measure in the computational basis to obtain their secret ballots.

Now consider the state

$$|S_n\rangle = \frac{1}{\sqrt{n!}} \sum_{S \in P_n^n} (-1)^{\tau(S)} |s_0\rangle |s_1\rangle ... |s_{n-1}\rangle$$

where P_n^n is the set of all permutations of a basis $\{0, 1, ..., n-1\}$ and $\tau(S)$ is the number of transpositions necessary to put S in canonical order. This state has the property that the measurement results of each particle will all always be different regardless of what basis the measurements are made in. This property will be exploited to distribute secret indices to each voter:

1. The procedure is similar to that for the ballot boxes except this time V_0 prepares $1 + n\delta_1$ states $|S_n\rangle$.

2. The distribution and testing procedure is the same except now the required condition to pass the security test is that each test row is a sequence $\in P_n^n$.

3. Each voter has one particle left which they measure in the computation basis to obtain a secret index.

There is a discrepancy/error in the paper regarding the procedure for casting votes. The incorrect procedure states that if d_i is the index integer for V_i and $r_{d_i i}$ is the d_i-th particle in their secret ballot, then V_i computes $r'_{d_i i} =$

$(r_{d_i i} + v_k) \mathrm{mod} m$ and replaces the old value in their ballot box with this new value but keeps all other values the same. Then, in the numerical example, they actually replace the old value with $r'_{d_i i} = (r_{d_i i} + v_k)$. Both of these procedures are wrong. Since the elements in the matrix and the votes are both less than or equal to $m-1$, simply adding one's vote v_k to $r_{d_i i}$ may cause that element in the matrix to be greater than $m-1$ which would then reveal the voter's index number as well as their vote when the final vote matrix is made publicly available. Using the property $(a+b)\mathrm{mod} m = (a \mathrm{mod} m + b \mathrm{mod} m) \mathrm{mod} m$, the correct procedure is to add the vote value as their index number as shown in their numerical example and then replace all matrix elements with their mod$m$ values. Once a voter V_i is done with that, they publicly reveal some permutation of column i to indicate that they have submitted their ballot. Once all voters have done this, all voters V_i reveal the correct order of the elements in their columns i simultaneously. The sum of each j^{th} row mod$m$ gives the vote submitted by the voter who had secret index $= j$. Thus, all voters can tally the votes because they can see each vote but they do not know who's vote it was.

Security from dishonest voters:

For n voters, m candidates, and d dishonest voters, the number of possible combinations of ballot numbers that the honest voters can have is m^{n-d-1}. In order for the d dishonest voters to determine the ballot number of a specific voter, we need

$$m^{n-d-1} = 1 => n-1 = d$$

Thus, we must have all but one dishonest voter in order to learn any specific voter's ballot number. The same is true for the ballot indices since each voter's ballot index is unique. Since the scenario of having only one honest voter is not realistic, this essentially shows that voter anonymity is guaranteed.

Security from 3rd party eavesdroppers:

The chance that an eavesdropper intercepts x particles (for $x < n$) during the distribution phase which do not get discarded is given by

$$P_e = \frac{\binom{n}{x}}{\binom{n+n\delta_0}{x}} \approx \mathcal{O}((\frac{1}{\delta_0})^x)$$

which approaches zero as δ_0 increases (and likewise for δ_1). If the eavesdropper intercepts and modifies a particle $p_{j_0,k}$, the j_0^{th} copy of $|\chi_n\rangle$ is changed to some state $|\phi_e\rangle$ and the probability that the security tests are passed is given by $1/2$ the probability of passing given a measurement in the computational basis plus $1/2$ the probability of passing given a measurement in the Fourier basis raised to the power n_0 since $n\delta_0$ tests are conducted:

$$P_e = (\frac{1}{2}P_C + \frac{1}{2}P_F)^{n_0}$$

where P_C and P_F are given by:

$$P_C = \sum_{\sum_k j_k mod m=0} |\langle \phi_e | j_0, j_i, ..., j_{n-1} \rangle_C|^2$$

$$P_F = \sum_{j=0}^{m-1} |\langle \phi_e | j, j, ..., j \rangle_F|^2$$

Where $\langle \phi_e | \chi_n \rangle < 1$ st. $P_e \to 0$ as δ_0 increases.

Thus, for large enough security factors δ, the protocol is secure from 3rd party attacks.

Other properties of the protocol:

Binding - if voter V_i decides to change their vote, they can only do so by revealing a permutation of column i that is not the correct permutation when revealing the final vote matrix. However, this error will be detected since at least one other voter will find that the vote reveled by the sum of the values in the row corresponding to their index value mod$m$ will not be equal to their vote.
Non-reusability - This property is guaranteed due to the same reasoning as the one above.
Variability - (see above)
Eligibility - eligibility is defined as participating in the protocol, we have already seen that the protocol is secure against 3rd party attacks.
Fairness - Since each column of the vote matrix is initially revealed in a different permutation, an attempt to calculate a partial tally will yield an incorrect result.
Self-tallying - Since all voters can see all submitted votes, all voters can tally the votes.

2 Comments on other protocols

The protocol above is the most optimal voting protocol that I found. Unfortunately, it is one of the many such protocols that utilize the distribution of very large numbers of entangled particles to satisfy security concerns. This is not a protocol that is technologically feasible, especially for something on the scale of a national election. Another much more technologically feasible protocol that utilizes the properties of entangled particles is described in [5]. In this "travelling ballot" protocol, there exists a central authority, T, that creates an entangled state of 2 particles. T keeps one particle and sends the other to the first voter. To vote, voters apply a unitary operation to the particle that corresponds to a vote of yes or no (where no is the identity and yes is an operation that maps states as $|0\rangle -> |1\rangle -> |2\rangle -> |0\rangle$). Each voter applies

their operation and then sends the particle to the next voter. The final state of the second particle is sent back to the authority. He measures both particles and can tally the vote by determining how many times the yes operation was applied to the second particle. This protocol lacks the fairness property since any two voters can decide to collaborate st. the first voter performs a measurement on the particle which breaks the entanglement and the second colluding voter can then determine the number of intermediate votes. This also means that any votes cast after the entanglement was broken will not be counted. If the two colluding voters only have one person between them, they are also able to determine how that person voted which violates the anonymity requirement. There is also nothing stopping a voter from voting twice (except applying the yes operation past some limit where T cannot accurately determine how many votes were cast). Finally, the protocol is not secure against a malicious T. In the paper discussing this protocol, they address some of these problems by creating a distributive protocol with large numbers of entangled states. It seems like that is the only way to make a truly secure voting protocol that satisfies all of the properties listed.

There are some protocols which attempt to utilize distribution alone to achieve high security, for example, the paper which I wrote the earlier report on[12][13] as well as this [8] conjugate coding protocol. However, these protocols get increasingly more complex the more secure they become. This is a shame given the elegance of Wang's protocol but both the conjugate coding and quantum blockchain protocols are indeed able to be implemented on some reasonable test scale (at the very least) using modern technology [7].

Other notes: The quantum blockchain and conjugate coding protocol both suggest using the 3-pass protocol for secure communication instead of the BB84 protocol. The 3-pass protocol allows communication with almost the same level of security as BB84 without the use of a shared key and is easier to implement[6][15].

References

[1] D Aggarwal, G Brennen, T Lee, M Santha, and M Tomamichel. Quantum attacks on bitcoin, and how to protect against them. *Ledger*, 3, 2017.

[2] A Appel. Which voting machines can be hacked through the internet?, 2016.

[3] E Goldberg. America faces a voting security crisis in 2020. here's why – and what officials can do about it.

[4] F Hardwick, A Gioulis, R Akram, and Markantonakis. E-voting with blockchain: An e-voting protocol with decentralisation and voter privacy. 2018.

[5] M Hillery, M Ziman, V Buzek, and M Bielikova. Towards quantum-based privacy and voting. 2005.

[6] Y Kanamori and S Yoo. Quanutm three-pass protocol: Key distribution using quantum superposition states. *International Journal of Network Security Its Applications*, 1(1), 2009.

[7] E Kiktenko, N Pozhar, M Anufriev, A Trushechkin, R Yunusov, Y Kurochkin, A Lvovsky, and A Fedorov. Quantum-secured blockchain. *Quantum Science and Technology*, 3, 2018.

[8] T Okamoto, K Suzuki, and Y Tokunaga. Quantum voting scheme based on conjugate coding. *NTT Technical Review*, 6(1), 2008.

[9] M Orcutt. Why security experts hate that "blockchain voting" will be used in the midterm elections, 2018.

[10] R Perper. Sierra leone just became the first country in the world to use blockchain during an election. online, 2018.

[11] X Sun and Q Wang. Bit commitment in categorical quantum mechanics. 2018.

[12] X Sun, Q Wang, P Kulicki, and M Sopek. A simple voting protocol on quantum blockchain. *International Journal of Theoretical Physics*.

[13] X Sun, Q Wang, P Kulicki, and X. Zhao. Quantum-enhanced logic-based blockchain i: Quantum honest-success byzantine agreement and qulogicoin. 2018.

[14] Qingle Wang, C Yu, F Gao, H Gi, and Q Wen. Self-tallying quantum anonymous voting. *Physical Review A*, 94, 2016.

[15] L Yang, L Wu, and S Liu. Quantum three-pass cryptography protocol. *Quantum Optics in Computing and Communications*, 4917, 2002.