A Comprehensive Analysis Of Quantum E-voting Protocols

M. Arapinis, N. Lamprou, E. Kashefi, A. Pappa

29 August 2018

Electronic Voting

compared to manual procedures, could provide:

- higher voter participation
- better accuracy
- enhanced security guarantees
- verification of counting against untrusted authorities

Electronic Voting

is based on computational assumptions like integer factorization and discrete log.

Why not use quantum mechanics to achieve better guarantees than classically possible, while attaining the same properties?

Electronic Voting properties

- eligibility
- vote privacy
- no double-voting
- verifiability
- receipt-freeness

Quantum Electronic Voting

We have categorised the proposed protocols in 4 groups:

1. “Two measurement bases”-based protocols
2. Traveling ballot protocols
3. Distributed ballot protocols
4. “Conjugate coding”-based protocols

“Two measurement bases”-based protocols

The ballot is an entangled state, with the following property:

- when measured in the computational basis, the sum of outcomes is equal to zero.
- when measured in the Fourier basis, all outcomes are equal.

$$|D_1\rangle = \frac{1}{\sqrt{m^{N-1}}} \sum_{\sum_{k=1}^{N} i_k = 0 \mod c} |i_1\rangle|i_2\rangle \ldots |i_N\rangle$$

[1] W. Huang, Q.-Y. Wen, B. Liu, Q. Su, S.-J. Qin, F. Gao, “Quantum anonymous ranking”, Physical Review A, vol. 89, no. 3, p. 032325, 2014.

[2] Q. Wang, C. Yu, F. Gao, H. Qi, Q. Wen, “Self-tallying quantum anonymous voting”, Physical Review A, vol. 94, no. 2, p. 022333, 2016.

“Two measurement bases”-based protocols

Protocol:

1. States are shared and tested (cut-and-choose technique)
2. Remaining are measured to create an (almost) random matrix
3. Voters add their vote to a specific place in the matrix according to the result of measuring:

$$|D_2\rangle = \frac{1}{\sqrt{N!}} \sum_{(i_1,i_2,\ldots,i_N)\in\mathcal{P}_N} |i_1\rangle|i_2\rangle\ldots|i_N\rangle$$

   and broadcast their column
4. Each vote is equal to the sum of the elements of a row in the matrix.

The cut-and-choose technique

- An untrusted party shares $N + N2^{\delta}$ states.
- Each voter checks 2^{δ} by asking the rest of the voters to measure half in computational and half in Hadamard.

Theorem (Cut-and-choose)

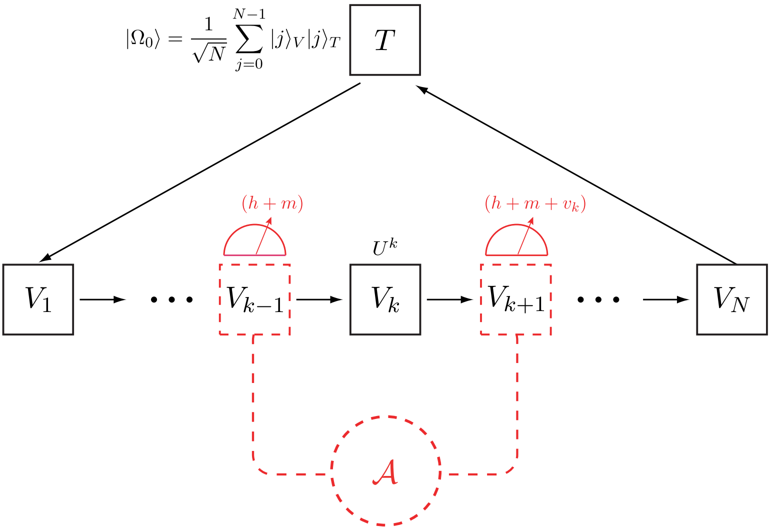
If an adversary shares the states and controls a fraction of the voters, then with non-negligible probability in δ, N *corrupted states can pass the test.*

Traveling ballot protocols

1. The Tallier prepares two entangled qudits and sends one to travel from voter to voter.
2. All voters apply an operation to the “ballot” qudit and finally it is sent back to the Tallier.
3. The Tallier measures the whole state and computes the result (of the referendum in this case).

[3] M. Hillery, M. Ziman, V. Buzek, M. Bielikova, “Towards quantum-based privacy and voting”, Physics Letters A, vol. 349, no. 1, pp. 75–81, 2006.
[4] J. A. Vaccaro, J. Spring, A. Chefles, “Quantum protocols for anonymous voting and surveying”, Physical Review A, vol. 75, no. 1, p. 012333, 2007.
[5] Y. Li, G. Zeng, “Quantum anonymous voting systems based on entangled state”, Optical review, vol. 15, no. 5, pp. 219–223, 2008.
[6] M. Bonanome, V. Buzek, M. Hillery, M. Ziman, “Toward protocols for quantum-ensured privacy and secure voting”, Physical Review A, vol. 84, no. 2, p. 022331, 2011.

Traveling ballot protocols



Problems with privacy, double-voting and verifiability!!

Distributed ballot protocols

1. T sends one qudit of the state: $|\Phi\rangle = \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} |j\rangle^{\otimes N}$ to each voter.

[6] M. Bonanome et al, Physical Review A, vol. 84, no. 2, p. 022331, 2011.

Distributed ballot protocols

1. T sends one qudit of the state: $|\Phi\rangle = \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} |j\rangle^{\otimes N}$ to each voter.
2. T also sends to each voter option qudits:

yes: $|\psi(\theta_y)\rangle = \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} e^{ij\theta_y}|j\rangle$

no: $|\psi(\theta_n)\rangle = \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} e^{ij\theta_n}|j\rangle$

[6] M. Bonanome et al, Physical Review A, vol. 84, no. 2, p. 022331, 2011.

Distributed ballot protocols

1. T sends one qudit of the state: $|\Phi\rangle = \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} |j\rangle^{\otimes N}$ to each voter.
2. T also sends to each voter option qudits:

$$\text{yes:} \quad |\psi(\theta_y)\rangle = \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} e^{ij\theta_y} |j\rangle$$

$$\text{no:} \quad |\psi(\theta_n)\rangle = \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} e^{ij\theta_n} |j\rangle$$

3. Each voter appends the option qudit to the ballot and performs a measurement and a correction operation, and sends the ballot to T.

[6] M. Bonanome et al, Physical Review A, vol. 84, no. 2, p. 022331, 2011.

Distributed ballot protocols

1. T sends one qudit of the state: $|\Phi\rangle = \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} |j\rangle^{\otimes N}$ to each voter.
2. T also sends to each voter option qudits:

yes: $|\psi(\theta_y)\rangle = \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} e^{ij\theta_y}|j\rangle$

no: $|\psi(\theta_n)\rangle = \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} e^{ij\theta_n}|j\rangle$

3. Each voter appends the option qudit to the ballot and performs a measurement and a correction operation, and sends the ballot to T.
4. (After corrections) T has the state:

$$|\Omega_m\rangle = \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} e^{ij(m\theta_y + (N-m)\theta_n)}|j\rangle^{\otimes 2N}$$

[6] M. Bonanome et al, Physical Review A, vol. 84, no. 2, p. 022331, 2011.

Distributed ballot protocols

With an appropriate mesurement, T learns the outcome m of the referendum.

- Tampering with the option qudits to learn θ_y and θ_n is detected by running the protocol many times and checking if the outcome is the same.

Distributed ballot protocols

With an appropriate mesurement, T learns the outcome m of the referendum.

- Tampering with the option qudits to learn θ_y and θ_n is detected by running the protocol many times and checking if the outcome is the same.

TRUE!

Distributed ballot protocols

With an appropriate mesurement, T learns the outcome m of the referendum.

- Tampering with the option qudits to learn θ_y and θ_n is detected by running the protocol many times and checking if the outcome is the same.

TRUE!

- However, double-voting does not require learning the actual values θ_y and θ_n.

Distributed ballot protocols: The d-transfer attack

Let's delve into more details about the protocol:

- $\theta_v = (2\pi l_v / D) + \delta$, where $l_v \in_R \{0, \ldots, D-1\}$ and $\delta \in_R [0, 2\pi/D)$.
- l_n is chosen such that $N(l_y - l_n \mod D) < D$.
- The values l_v, l_y, δ are known only to T.
- T retrieves the outcome by applying a unitary to the received state:

$$\frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} e^{ij(m\theta_y + (N-m)\theta_n)} |j\rangle^{\otimes 2N} \rightarrow \frac{1}{\sqrt{D}} \sum_{j=0}^{D-1} e^{2\pi i j m (l_y - l_n)/D} |j\rangle^{\otimes 2N}$$

Distributed ballot protocols: The d-transfer attack

Observation 1: If $l_y - l_n$ is known, then a malicious voter can transfer d votes from one option to the other.

Observation 2: We can find the difference with overwhelming probability in the number N of voters

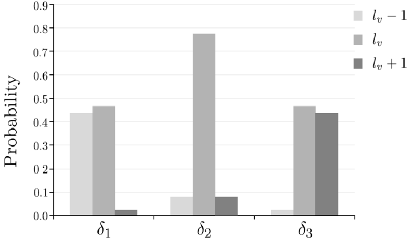
Distributed ballot protocols: Finding $l_y - l_n$

- An adversary controls ϵN of the voters, who are (all but one) instructed to vote half "yes" and half "no".
- Remaining votes are used to run Algorithm 1

Algorithm 1 Adversary's algorithm

Input: $D, |\psi(\theta_v)\rangle_1, \cdots, |\psi(\theta_v)\rangle_{\varepsilon N/2}$
Output: $\tilde{l} \in \{0, \ldots, D-1\}$

Record $= [0, \ldots, 0] \in \mathbb{N}^{1\times D}$; ▷ This vector shows us how many values are observed in each interval
Solution $= [\text{"}Null\text{"}, \text{"}Null\text{"}] \in \mathbb{N}^{1\times 2}$;
$i, l, m = 0$;
while $i \leq \varepsilon N/2$ **do**
  Measure $|\psi(\theta_v)\rangle_i$ by using POVM operator $E(\theta)$ from Eq.(2), the result is y_i;
  Find the interval for which $\frac{2\pi j}{D} \leq y_i \leq \frac{2\pi(j+1)}{D}$;
  Record$[j]$ =++;
  i++;
end while
while $l < D$ **do**
  if Record$[l] \geq 40\%(\varepsilon N/2)$ **then**
    Solution$[m] = l$;
    $m++$;
  end if
  $l++$;
end while
if Solution $== [0, D-1]$ **then**
  Solution $=$ [Solution[1], Solution[0]];
end if
return $\tilde{l} =$ Solution[0];



Distributed ballot protocols: Finding $l_y - l_n$

Theorem (Observation 2)

Algorithm 1 finds the difference $l_y - l_n$ with overwhelming probability in N:

$$\Pr\left[Algo_y - Algo_n = l_y - l_n\right] > 1 - \frac{1}{\exp(\Omega(N))}$$

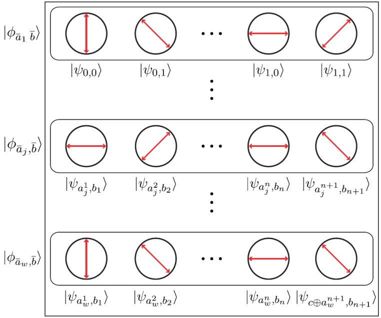
Theorem (Efficiency)

If the protocol runs less than $\exp(\Omega(N))$ times, then the attack succeeds with probability at least 25%.

“Conjugate coding”-based protocols

[7] T. Okamoto and Y. Tokunaga, “Quantum voting scheme based on conjugate coding”, NTT Technical Review, vol. 6, no. 1, pp. 18, 2008.
[8] R. Zhou, L. Yang, “Distributed quantum election scheme”, arXiv:1304.0555 [quant-ph].

1. EA creates one blank ballot for each voter.

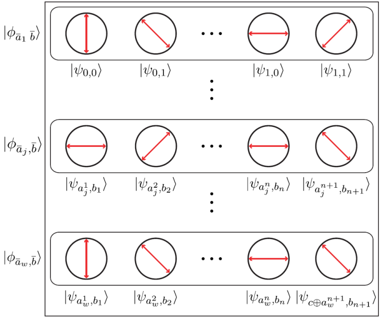


“Conjugate coding”-based protocols

[7] T. Okamoto and Y. Tokunaga, “Quantum voting scheme based on conjugate coding”, NTT Technical Review, vol. 6, no. 1, pp. 18, 2008.

[8] R. Zhou, L. Yang, “Distributed quantum election scheme”, arXiv:1304.0555 [quant-ph].

1. EA creates one blank ballot for each voter.
2. Each voter re-randomizes it.

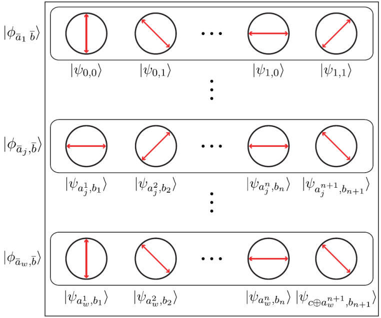


“Conjugate coding”-based protocols

[7] T. Okamoto and Y. Tokunaga, “Quantum voting scheme based on conjugate coding”, NTT Technical Review, vol. 6, no. 1, pp. 18, 2008.

[8] R. Zhou, L. Yang, “Distributed quantum election scheme”, arXiv:1304.0555 [quant-ph].

1. EA creates one blank ballot for each voter.
2. Each voter re-randomizes it.
3. Each voter encodes vote in the ballot and sends it to T.

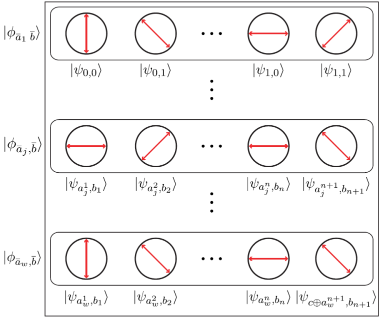


“Conjugate coding”-based protocols

[7] T. Okamoto and Y. Tokunaga, “Quantum voting scheme based on conjugate coding”, NTT Technical Review, vol. 6, no. 1, pp. 18, 2008.

[8] R. Zhou, L. Yang, “Distributed quantum election scheme”, arXiv:1304.0555 [quant-ph].

1. EA creates one blank ballot for each voter.
2. Each voter re-randomizes it.
3. Each voter encodes vote in the ballot and sends it to T.
4. EA announces bases to T.

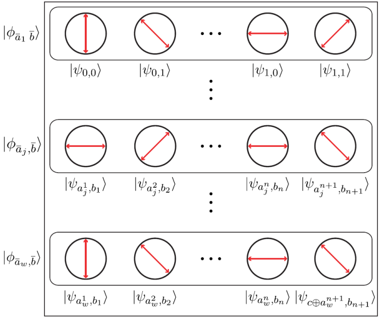


“Conjugate coding”-based protocols

[7] T. Okamoto and Y. Tokunaga, “Quantum voting scheme based on conjugate coding”, NTT Technical Review, vol. 6, no. 1, pp. 18, 2008.

[8] R. Zhou, L. Yang, “Distributed quantum election scheme”, arXiv:1304.0555 [quant-ph].

1. EA creates one blank ballot for each voter.
2. Each voter re-randomizes it.
3. Each voter encodes vote in the ballot and sends it to T.
4. EA announces bases to T.
5. T measures and announces result.



Vulnerabilities of “Conjugate coding”-based protocols

- Malleability of ballots: an adversary can change the vote.
- Violation of privacy: the EA can introduce a serial number in the blank ballot.
- One-more unforgeability: the scheme is based on a hard-to-solve problem for quantum computers. Given w blank ballot fragments, it is hard to produce $w + 1$ valid blank fragments.

Conclusion

These are great ideas!!! However...

- The cut-and-choose technique in dual-basis protocols is not working as is, and needs to be further studied.
- Unless combined with some new technique, the traveling ballot protocols do not seem to provide a viable solution, as double-voting is always possible, and there is no straightforward way to guarantee privacy.
- Distributed ballot protocols give strong privacy guarantees but cannot guarantee verifiability and the efforts to stop double voting are not yet successful.
- Except from privacy issues against a dishonest EA, the conjugate coding protocols are based on a hardness assumption that should be further analysed.

Conclusion - What is next

- Properly define the desired properties
- Improve the already identified faulty subroutines in the proposed protocols
- Study of classical e-voting protocols and identify classical subroutines that could be improved by quantum communication