A Fair and Robust Voting System by Broadcast

P. von Bergen

Bern University of Applied Sciences

March 27, 2013

Outline

1. Anonymous voting by two-round public discussion, Hao, Ryan, Zieliński 2008
2. A Fair and Robust Voting System by Broadcast, Khader, Smyth, Ryan, Hao 2012

Outline

1. Anonymous voting by two-round public discussion, Hao, Ryan, Zieliński 2008
2. A Fair and Robust Voting System by Broadcast, Khader, Smyth, Ryan, Hao 2012

- Boardroom elections
- No trusted parties
- All communication is public
- Two rounds
  1. Key publication
  2. Vote
- Challenges
  - Ballot secrecy
  - Self-tallying
  - Dispute-freeness

Preparation

- Cyclic group $(G, \cdot)$ of prime order q, where Diffie-Hellman problem is intractable
- g is a generator in G
- The n participants agree on (G, g)
- Each participant P_i select a value as secret: $x_i \in_R \mathbb{Z}_q$
- Each participant P_i computes $a_i = g^{x_i} \ mod \ p$
- Each participant P_i prove the knowledge of x_i

Round 1

- Each participant P_i publishes a_i and its ZKP

At the end of round 1

- Each participant P_i checks the validity of the ZKPs
- Each participant P_i computes

$$h_i = g^{y_i} = \frac{\prod_{j=1}^{i-1} a_j}{\prod_{j=i+1}^{n} a_j} = g^{(x_1+\ldots+x_{i-1})-(x_{i+1}+\ldots+x_n)} \tag{1}$$

Round 2

- Each participant P_i publishes $b_i = h_i^{x_i} g^{v_i} = g^{x_i y_i} g^{v_i}$
- Each participant P_i compute and publish a ZKP showing that $v_i \in \{1, 0\}$ (1 for yes, 0 for no)

Tallying

$$\prod_{i=1}^{n} b_i = \prod_{i=1}^{n} g^{x_i y_i} g^{v_i} = g^{\sum_{i=1}^{n} v_i}$$

Because

$$\prod_{i=1}^{n} g^{x_i y_i} = 1$$

Proof

$$\prod_{i=1}^{n} g^{x_i y_i} = 1 \Rightarrow \sum_{i=1}^{n} x_i y_i = 0$$

By definition (from (1))

$$y_i = \sum_{j=1}^{i-1} x_j - \sum_{j=i+1}^{n} x_j$$

Proof (continued)

Hence

$$\begin{aligned}\sum_{i=1}^{n} x_i y_i &= \sum_{i=1}^{n}\sum_{j=1}^{i-1} x_i x_j - \sum_{i=1}^{n}\sum_{j=i+1}^{n} x_i x_j \\ &= \sum\sum_{j<i} x_i x_j - \sum\sum_{i<j} x_i x_j \\ &= \sum\sum_{j<i} x_i x_j - \sum\sum_{j<i} x_j x_i \\ &= 0\end{aligned}$$

Tallying (continued)

$$\prod_{i=1}^{n} b_i = \prod_{i=1}^{n} g^{x_i y_i} g^{v_i} = g^{\sum_{i=1}^{n} v_i}$$

where $\sum_{i=1}^{n} v_i$ is the number of yes votes denoted γ.

The discrete logarithm g^{γ} can be computed, since γ is normally a small number. We can use the baby-step giant-step algorithm.

The ZKPs have also to be verified.

ZKP round 1: Schnorr's signature

Prove knowledge for the exponent of g^{x_i}:

- H: publicly agreed hash function
- Prover computes and sends (g^v, r) where $r = v - x_i z$, $v \in_R \mathbb{Z}_q$ and $z = H(g, g^v, g^{x_i}, i)$
- Verifier checks if g^v and $g^r g^{x_i z}$ are equal

Proof

$$
\begin{aligned}
g^v &= g^r \cdot g^{x_i z} \\
&= g^{v - x_i z} \cdot g^{x_i z} \\
&= \frac{g^v}{g^{x_i z}} \cdot g^{x_i z} \\
&= g^v
\end{aligned}
$$

ZKP round 2: CDS

Prove that the $v_i \in \{0, 1\}$:

- Convert terms of the protocol in a ElGamal encryption
- $h_i = g^{y_i}$ becomes the public key and x_i the randomization
- $(a, b) = (g^{x_i}, (g^{y_i})^{x_i} g^{v_i})$ where $g^{v_i} = 1$ or $g^{v_i} = g$

Sign

- Given (a, b) and v_i
- For all $k \in \{0, 1\} \backslash v_i$
- $c_k \in_R \mathbb{Z}_q^*$, $s_k \in_R \mathbb{Z}_q^*$, $w \in_R \mathbb{Z}_q^*$
- $a_k = \frac{g^{s_k}}{a^{c_k}}$, $b_k = \frac{h_i^{s_k}}{\left(\frac{b}{g^k}\right)^{c_k}}$
- Witnesses: $a_v = g^w$ and $b_v = h_i^w$
- Challenge: $c_v = H(a, b, a_0, b_0, a_1, b_1) - \sum_{i \in \{0,1\} \backslash v_i} c_i$
- Response: $s_v = w + x_i \cdot c_v$
- Output signature (a_k, b_k, c_k, s_k) for all $k \in \{0, 1\}$

Verify

- Given (a, b) and $(a_0, b_0, c_0, s_0, a_1, b_1, c_1, s_1)$
- For each $k \in \{0, 1\}$, check if $g^{s_k} = a_k \cdot a^{c_k}$ and $h_i^{s_k} = b_k \cdot (b/g^k)^{c_k}$
- Check if $H(a, b, a_0, b_0, a_1, b_1) = \sum_{k \in \{0,1\}} c_k$

This signature scheme can be extended to multiple choices.

This signature scheme also includes a challenge c_v which acts as a computationally binding commitment to values a and b, but it is not used in the above protocol.

Proof

$$\begin{aligned} g^{s_v} &= a_v \cdot a^{c_v} \\ g^{w+x_i \cdot c_v} &= g^w \cdot a^{c_v} \\ &= g^w \cdot g^{x_i c_v} \end{aligned}$$

$$\begin{aligned} h_i^{s_v} &= b_v \cdot \left(\frac{b}{g^k}\right)^{c_v} \\ h_i^{w+x_i \cdot c_v} &= h_i^w \cdot \left(\frac{b}{g^k}\right)^{c_v} \\ g^{y_i w + x_i y_i \cdot c_v} &= g^{y_i w} \cdot \left(\frac{b}{g^k}\right)^{c_v} \\ g^{y_i w + x_i y_i \cdot c_v} &= g^{y_i w} \cdot \left(\frac{g^{x_i y_i} \cdot g^{v_i}}{g^k}\right)^{c_v} \quad \text{with } g^k = g^{v_i} \end{aligned}$$

Proof (continued)

$$\begin{aligned} g^{s_k} &= a_k \cdot a^{c_k} \\ &= \frac{g^{s_k}}{a^{c_k}} \cdot a^{c_v} \end{aligned}$$

$$\begin{aligned} h_i^{s_k} &= b_k \cdot \left(\frac{b}{g^k}\right)^{c_k} \\ &= \frac{h_i^{s_k}}{\left(\frac{b}{g^k}\right)^{c_k}} \cdot \left(\frac{b}{g^k}\right)^{c_k} \end{aligned}$$

Anonymous voting by two-round public discussion

Extension to multiple candidates

- For elections with only 2 candidates, the same protocol can be used, instead of sending 'yes/no', one simply sends 'A/B'.
- For more candidates, a possibility would be to run the single-candidate protocol in parallel for k candidates
- Another way if each voter is only permitted to choose one candidate is following:
  - k independent generator are used (one for each candidate)
  - in second round P_i sends $g^{x_i y_i} \cdot \varrho_i$ with a ZKP that $\rho_i \in \{g_1, g_2, ..., g_k\}$
  - tallying: $\prod_{i=1}^{n} g^{x_i y_i} \cdot \varrho_i = g_1^{c_1} \cdot g_2^{c_2} \cdots g_k^{c_k}$ where c_1 to c_k are the counts of votes for the k candidates correspondingly

- Ballot secrecy:
  - ballot is encrypted with ElGamal $(g^{x_i}, (g^{x_i})^{y_i} \cdot g^{v_i})$
  - y_i is unknown to attackers as it is computed from all x_i which is a random value in $\mathbb{Z}_q$
  - under the decisional Diffie-Hellman assumption, an attacker cannot distinguish the encrypted ballot from a random group element.
  - zero knowledge proofs don't reveal any information more than intended
- Self-tallying: as we have seen, this requirement is satisfied.
- Dispute freeness: as the channel is public and authenticated, each voter can verify that the other voters followed the protocol. More over, the Zero Knowledges Proofs proves the respect of the rules.

Anonymous voting by two-round public discussion

Problems

- The last voter knows the result before other voters $\Rightarrow$ no fairness
- If a voter aborts in the second round, she disrupts the election $\Rightarrow$ no robustness

Outline

1. Anonymous voting by two-round public discussion, Hao, Ryan, Zieliński 2008

2. A Fair and Robust Voting System by Broadcast, Khader, Smyth, Ryan, Hao 2012

A Fair and Robust Voting System by Broadcast

Base concepts

- Resolves the problems of HRZ10
- Adds a commitment round
- Allows to recover the result when a voter has aborted a round
- Three (four) rounds
  1. Setup
  2. Commitment
  3. Vote
  4. (Recovery)

- The computationally binding in CDS signature scheme (second ZKP) is used
- b_i value is not published in commitment round
- $\Rightarrow$ So, no partial result can be compute before all voters have voted
- b_i value is publish in the third round (Voting round)

- If a voter refuses to vote, her $b_i = g^{y_i x_i} \cdot g^{v_i}$ is not published
- $\prod_{i=1}^{n} b_i = g^{\sum_{i=1}^{n} v_i}$ can't be computed

So,

- let be L the set of voter that have published a valid vote
- each voter $i \in L$ computes

$$\hat{h}_i = \frac{\prod\limits_{j \in \{i+1,\dots,n\} \setminus L} a_j}{\prod\limits_{j \in \{1,\dots,i-1\} \setminus L} a_j} = g^{\hat{y}_i}$$

- and publish $\hat{h}^{x_i}$ with a ZKP that $\log_g a_i = \log_{\hat{h}_i} \hat{h}_i^{x_i}$
  $\Rightarrow \log_g g^{x_i} = \log_{\hat{h}_i} \hat{h}_i^{x_i}$

Tallying

$$g^{\sum_{i \in L} v_i} = \prod_{i \in L} \hat{h}_i^{x_i} \cdot h_i^{x_i} \cdot g^{v_i} = \prod_{i \in L} \hat{h}_i^{x_i} \cdot b_i$$

where $\sum_{i \in L} v_i$ is the number of yes.

Proof

$$g^{\sum\limits_{i \in L} v_i} = \prod_{i \in L} g^{\hat{y}_i x_i} \cdot g^{y_i x_i} \cdot g^{v_i} = g^{\sum\limits_{i \in L} x_i y_i + x_i \hat{y}_i} \cdot g^{\sum\limits_{i \in L} v_i} = g^0 \cdot g^{\sum\limits_{i \in L} v_i}$$

With

$$y_i = \sum_{j=1}^{i-1} x_j - \sum_{j=i+1}^{n} x_j$$

$$\hat{y}_i = \sum_{j \in \{i+1,\ldots,n\} \setminus L} x_j - \sum_{j \in \{1,\ldots,i-1\} \setminus L} x_j$$

Thus

$$\sum_{j \in L} (x_i y_i) + (x_i \hat{y}_i) = 0$$

- Goal: prove that $\hat{h}^{x_i}$ has been computed correctly

$$\log_g g^{x_i} = \log_{\hat{h}_i} \hat{h}_i^{x_i}$$

- Sign:
  - given $g, \hat{h}_i, x_i$ select a random value $w \in \mathbb{Z}_q^*$
  - compute $g' = g^w$ and $\hat{h}_i' = \hat{h}_i^w$, challenge $c = H(g', \hat{h}_i')$, response $s = w + c \cdot x_i$
  - publish $(g', \hat{h}_i', s)$
- Verify:
  - given $g, \hat{h}_i, g^{x_i}, \hat{h}_i^{x_i}$ and signature $(g', \hat{h}_i', s)$ check if $g^s = g' \cdot (g^{x_i})^c$ and $\hat{h}_i^s = \hat{h}_i' \cdot (\hat{h}_i^{x_i})^c$ where $c = H(g', \hat{h}_i')$

1. Setup Round
   - choose x_i
   - compute and publish g^{x_i}
   - at the end of the round, compute $g^{y_i} = h_i$
2. Commitment Round
   - choose v_i
   - compute signature that $v_i \in \{0, 1\}$ which also works as commitment
3. Voting Round
   - publish v_i
4. Recovery Round if needed
   - compute $\hat{h}_i$
5. Tallying

Added properties:

- Fairness
- Robustness