The Big Gap Between Academic and Business World in E-Voting

Reto E. Koenig

University of Fribourg
&
Bern University of Applied Science

18.11.2010

What We Expect

Voting Phase

| Setup | Casting | Tallying |

Attributes

Time	Every voting-phase ends within a reasonable time
Simplicity	The voter has to understand the Vote-Casting phase
Operability	Every voting-phase has to be operated by the 'normal' guy
Verifiability	To be sure that the votes are cast, collected, and counted as intended by each voter
Privacy	No vote can be linked back to the voter at any time
Coercion Resistance	Every voter can express the will without fear.

Which Attributes are provided by...

Business Systems

Time	Most important
Simplicity	Most important
~~Operability~~	Obvious reasons
~~Verifiability~~	"You have to believe us"
~~Privacy~~	Organisational measures
~~Coercion Resistance~~	Organisational measures

Academic Systems

~~Time~~	"It's not exponential!"
~~Simplicity~~	Don't care
~~Operability~~	Don't care
Verifiability	Most important
(Privacy)	"50 Years should be enough for every one"
(Coercion Resistance)	"We are working on it"

Where to spend a Dollar?

Is it important ...

Time	...to know the result within a reasonable time?
Simplicity	...that the voter can vote?
Operability	...that the software is operational?
Verifiability	...that you know why the final tally is this way and not the other?
Privacy	...that your will is kept non disclosable
Coercion Resistance	...that you can not be put under pressure?

A Case-Study in respect to Large Scale Voting

Voting Phase

Setup	Casting	Tallying

A Case-Study in respect to Large Scale Voting

Voting Phase

Setup	Casting	Tallying
1 Month	1 Month	6 hours

Amount of Votes

Just as an indicator we use an example of 1 Mio countable votes.

A Case-Study in respect to Large Scale Voting

Voting Phase

Setup	Casting	Tallying
		*

The Business Side

Algorithm at the Tally-Side

1. read each vote
2. count
3. present

Timings...

$M =$ Amount of votes

$Time = M * (read + addition) + present$

This happens in a fraction of minutes even if M is fairly large ($> 1'000'000$)

The Business Side

Algorithm at the Voter-Side

1. You have to believe ... (¿But whom?)

Timings...

0

Definition

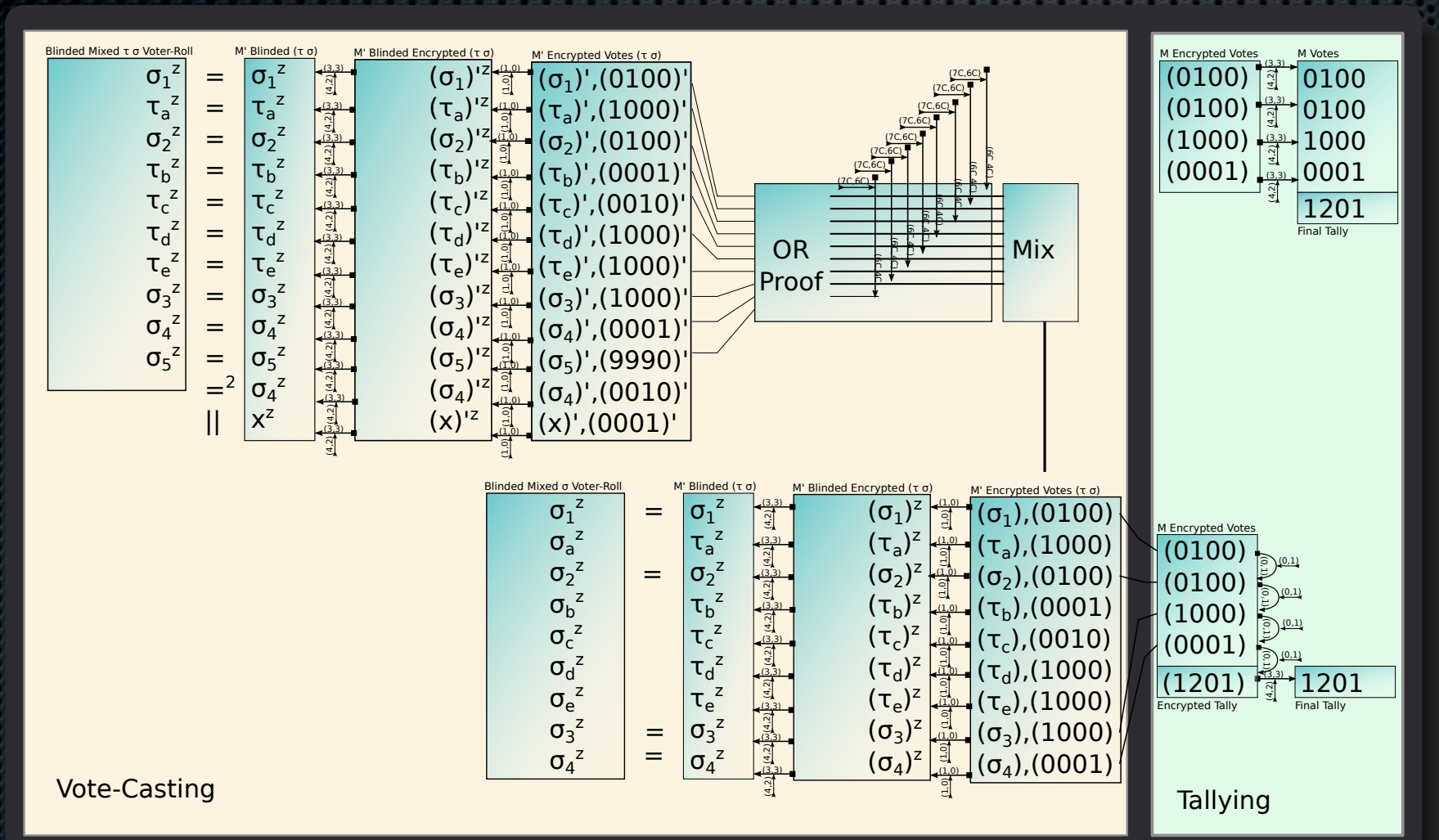
$(\#modExp, \#modMul) = opCount(operation, args...)$

Description Counts the amount of modExps and the amount of modMuls of a certain operation.

Input Any operation

Output A tuple (e,m)
where e represents the amount of modExp
where m represents the amount of modMul

Big-Picture


Blinded Mixed τ σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
OR Proof
Mix
M Encrypted Votes
M Votes
Final Tally
Blinded Mixed σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
M Encrypted Votes
Encrypted Tally
Final Tally
Vote-Casting
Tallying

Algorithm I at the Tally-Side

For M votes

1. decrypt each ballot $\rightarrow$ vote
2. prove the correct decryption of each ballot
3. count
4. present

Costs

$M * (opCount(decrypt) + opCount(proofCorrectDecryption))$
$M * ((1,2) + (2,1)) = M * (3,3)$

Timings... concrete-large scale

Assumption

Security parameter	k=4096
Time for $modExp(k)$	$0.1sec$ (Assumption 2010)
Time for $modMul(k)$	$0.01sec$ (Assumption 2010)
Parallelisation	p
Amount of votes M	$1'000'000$
CostFunction	$M * (3,3)_{opCount}$

$Time = 1'000'000 * (0.3sec + 0.03sec) = 330'000sec$
With $p \cdot 86400sec \cdot day^{-1}$
the tally would be ready in about $\frac{4}{p}$ days.

Algorithm at the Voter-Side

For M votes

1. Verification of each proof

Costs for complete Verification

$$M * (opCount(verify(proofCorrectDecryption)) + opCount(count))$$

$$\overline{M * ((4,2) + (0,0)) = M * (4,2)}$$

Voter / Observer: Timings... concrete-large scale

Assumption

Security parameter k=4096

Time for $modExp(k)$ $0.2sec$ (Assumption 2010)

Time for $modMul(k)$ $0.02sec$ (Assumption 2010)

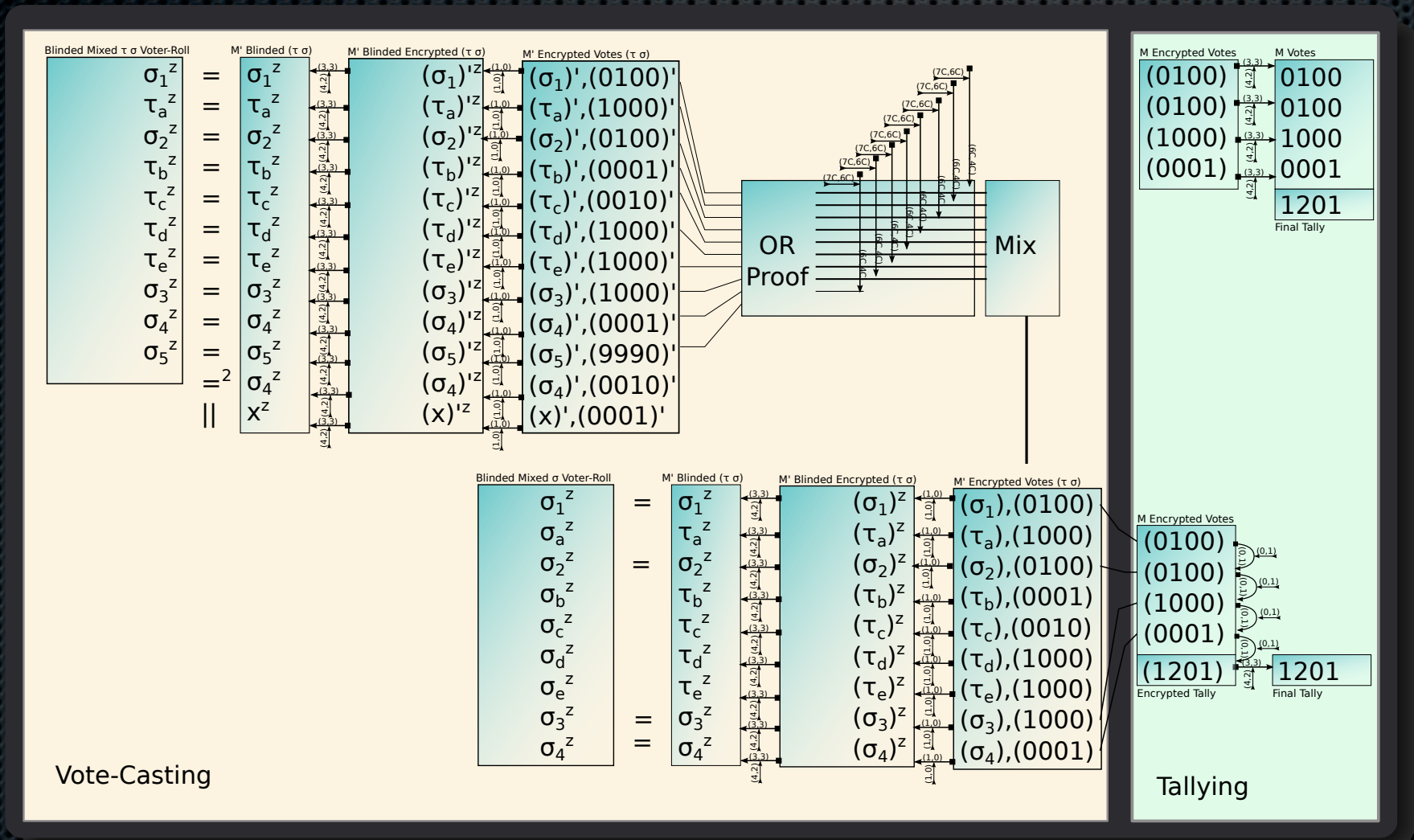
Parallelisation p usually 1 at the users side

Amount of votes M $1'000'000$

CostFunction $M * (4, 2)_{opCount}$

Time for complete verification
$= 1'000'000 * (0.8sec + 0.04sec) = 840'000sec \approx \frac{10}{p}$ days.

Big-Picture


Blinded Mixed τ σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
OR Proof
Mix
M Encrypted Votes
M Votes
Final Tally
Blinded Mixed σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
M Encrypted Votes
Encrypted Tally
Final Tally
Vote-Casting
Tallying

Algorithm II at the Tally-Side: Homomorphic Counting

For M votes

1. homomorphic 'sum' each ballot
2. decrypt sum of all ballots $\rightarrow$ final Tally
3. prove the correct decryption of the sum ballot

Costs

$$\frac{opCount(\otimes(M)) + opCount(decrypt) + opCount(proofCorrectDecryption) + opCount(\sum(M,c))}{}$$

ElGamal[a]:(0,M)+(1,2)+(3,3)+$(0,\sqrt{M^{c-1}})$= $(4, 5 + M + \sqrt{M^{c-1}})$

$Paillier : (0, M) + (1, 2) + (3, 3) + (0, 0) = (4, 5 + M)$

[a] c: Amount of choices within the vote (1-out-of-n)

Tallyer: Timings... concrete-large scale

Assumption

Security parameter	k=4096
Time for $modExp(k)$	$0.1sec$ (Assumption 2010)
Time for $modMul(k)$	$0.01sec$ (Assumption 2010)
Parallelisation	p
Amount of votes M	$1'000'000$
Amount of choices	2
CostFunction	ElGamal: $(4, M + \sqrt{M^{c-1}} + 5)$ Paillier: $(4, M + 5)$

ElGamal Time for
tally$= 0.4 + 10'000 + 1000 + 0.05 = 11'000.45 \approx \frac{3}{p}$ h. Paillier
Time for tally$= 0.4 + 10'000 + 0.05 = 10'000.45 \approx \frac{3}{p}$ h.

Algorithm at the Voter-Side

For M votes

1. homomorphic 'sum' each ballot
2. Verification of the proof
3. Verification of the correct decryption

Costs for complete verification

$$opCount(\otimes(M)) + opCount(verify(proofCorrectDecryption)) + opCount(verify(\sum(M, c)))$$

ElGamal: $(0, M) + (3, 3) + (0, 1) = (3, M + 4)$

Paillier: $(0, M) + (3, 3) = (3, M + 3)$

Voter / Observer: Timings... concrete-large scale

Assumption

Security parameter k=4096

Time for $modExp(k)$ $0.2sec$ (Assumption 2010)

Time for $modMul(k)$ $0.02sec$ (Assumption 2010)

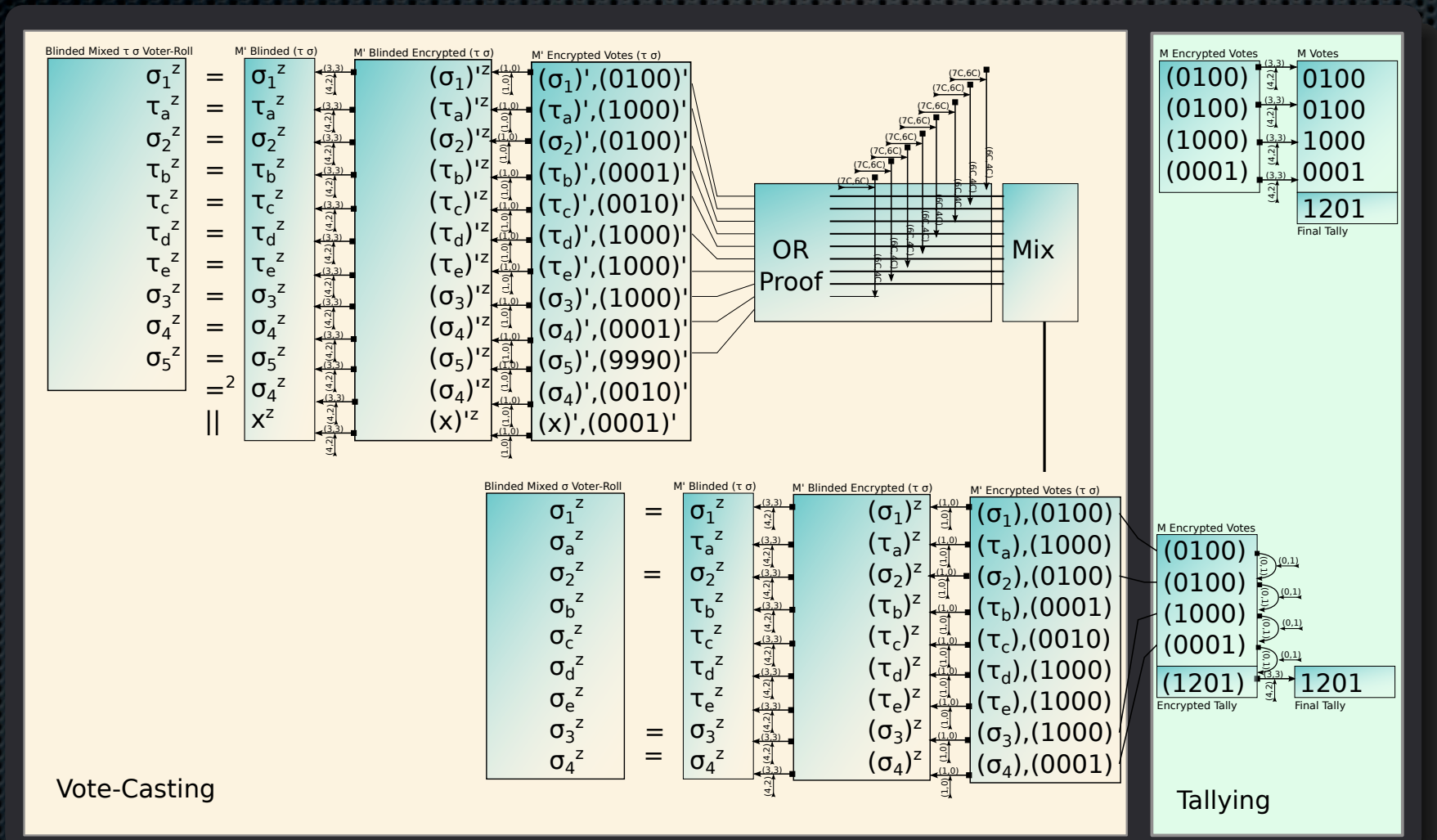
Parallelisation p usually 1 at the users side

Amount of votes M $1'000'000$

CostFunction ElGamal: $(3, M+4)$
Paillier: $(3, M+3)$

Time for complete
verification$= 0.6 + 20'000.02 + 0.06 = 20'000.68sec \approx \frac{5.5}{p}$ h.

Big-Picture


Blinded Mixed τ σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
OR Proof
Mix
M Encrypted Votes
M Votes
Final Tally
Blinded Mixed σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
M Encrypted Votes
Encrypted Tally
Final Tally
Vote-Casting
Tallying

From 1-out-of-2 $\rightarrow$ 1-out-of-n

Well... usually voting is somewhat more complicated: Think about choosing from 50 candidates:

The Vote

Choice 1	$\cdots$	Choice 49	Choice 50
000000	000000	000000	000000

The Vote Count

$Vote_1$	000000	$\cdots$	000001	000000
$Vote_2$	000000	$\cdots$	000000	000001
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$Vote_{1Mio}$	000000	$\cdots$	000000	000001
Sum	000001	$\cdots$	000001	000002

Tallyier: Timings... concrete-large scale

Assumption

Security parameter	k=4096
Time for $modExp(k)$	$0.1sec$ (Assumption 2010)
Time for $modMul(k)$	$0.01sec$ (Assumption 2010)
Parallelisation	p
Amount of votes M	$1'000'000$
Amount of choices	50
CostFunction	ElGamal: $(4, M + \sqrt{M^{c-1}} + 5)$ Paillier: $(4, M + 5)$

ElGamal Time for tally$= 0.4 + 10'000 + 10^{146} >>$ *Googol* seconds.
Paillier Time for tally$= 0.4 + 10'000 + 0.05 = 10'000.45 \approx \frac{3}{p}$ h.

1-out-of-n Elections

Split-Vote

- Each choice is within a separate vote $vote_0, \ldots, vote_c$
- Each $vote_i$ must be an encryption of: 1 or 0
- The homomorphic sum of all $vote_0 + \ldots + vote_c$ must be an encryption of: 1 or 0

Algorithm III at the Tally-Side: Split / Homomorphic Counting

For M votes

1. homomorphic 'sum' each ballot per choice c
2. decrypt sum of all ballots $\rightarrow$ final Tally per choice c
3. prove the correct decryption of the sum ballot per choice c

Costs

$$\frac{opCount(\otimes(M), C) + opCount(decrypt, C) + opCount(proofCorrectDecryption, C) + opCount(\sum(M, 1), N)}{\text{ElGamal:}(0, M \cdot C) + (1 \cdot C, 2 \cdot C) + (3 \cdot C, 3 \cdot C) + (0, \sqrt{M} \cdot C)}$$
$$= (4 \cdot C, (M + \sqrt{M} + 5) \cdot C)$$

Timings... concrete-large scale

Assumption

Security parameter	k=4096
Time for $modExp(k)$	$0.1sec$ (Assumption 2010)
Time for $modMul(k)$	$0.01sec$ (Assumption 2010)
Parallelisation	p
Amount of votes M	$1'000'000$
Amount of choices C	50
CostFunction	ElGamal: $(4 \cdot C, (M + \sqrt{M^{C-1}} + 5) \cdot C)$

ElGamal Time for tally$= 2 + 500'502.5 = 500'504.5 \approx \frac{6}{p}$ days.

Algorithm at the Voter-Side

For M votes

1. homomorphic 'sum' each ballot
2. Verification of the proof
3. Verification of the correct decryption

Costs for complete verification

$$\frac{\begin{array}{l} opCount(\otimes(M), C) + \\ opCount(verify(proofCorrectDecryption), C) + \\ opCount(verify(\sum(M, C)), N) \end{array}}{\begin{array}{l} \text{ElGamal: } (0, M \cdot C) + (3 \cdot C, 3 \cdot C) + (0, 1 \cdot C) \\ = (3 \cdot C, (M + 4) \cdot C) \end{array}}$$

Voter / Observer: Timings... concrete-large scale

Assumption

Security parameter k=4096

Time for $modExp(k)$ $0.2sec$ (Assumption 2010)

Time for $modMul(k)$ $0.02sec$ (Assumption 2010)

Parallelisation p usually 1 at the users side

Amount of votes M $1'000'000$

CostFunction ElGamal: $(3 \cdot N, (M+4) \cdot N)$

Time for complete verification$= 30 + 1'000'004 = 1'000'034sec \approx \frac{12}{p}$ days.

Tally-Conclusion

homomorphic vs. open

Security Privacy is top if the tally is done homomorph and if the private key is not unveiled at the end of the tally.

Usability For the voter / Observer homomorph tally can be completely verified by every-one

Crypto An additive homomorphic crypto-system is highly preferable for all players.

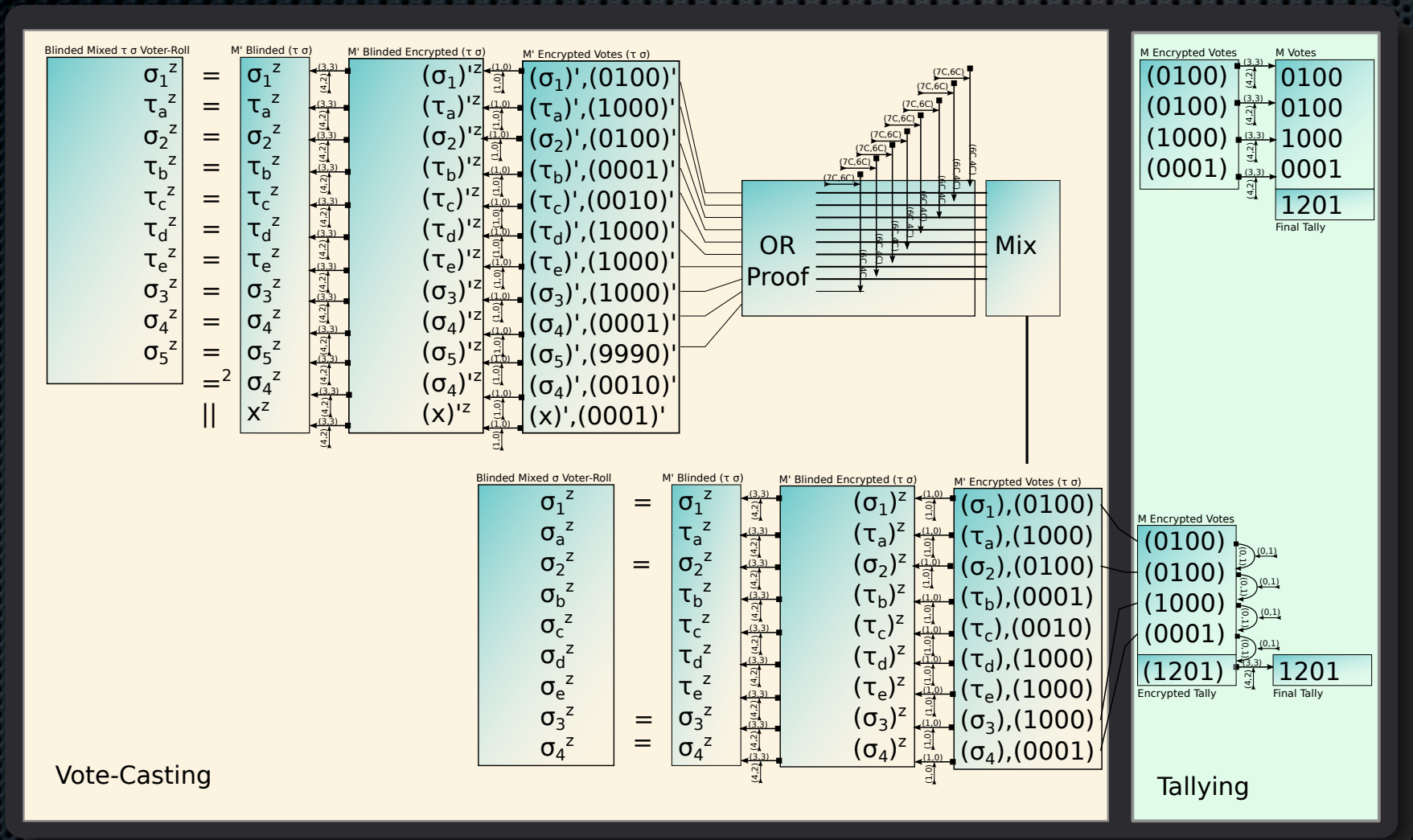
1 Mio-Tally of 1-out-of-50	ElGamal: 6 days
	Paillier: 3 h
Verification	ElGamal: 12 days (10 days)
	Paillier: 5.5 h

A Case-Study in respect to Large Scale Voting

Voting Phase

Setup	Casting	Tallying
	*	

Big-Picture


Blinded Mixed τ σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
OR Proof
Mix
M Encrypted Votes
M Votes
Final Tally
Blinded Mixed σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
M Encrypted Votes
Encrypted Tally
Final Tally
Vote-Casting
Tallying

PET with Voter-Hint

For each vote out of M'

1. blind the Credential and the hinted Credential
2. homomorphic $\oslash$ each posted Credential with the hinted Credential
3. decrypt the result
4. Verification of the correct decryption

Costs

$$\frac{opCount(modexp, 2, M') + opCount(\oslash, M') + opCount(decrypt, M') + opCount(proofCorrectDecryption, M')}{(2 \cdot M', 0) + (0, M') + (1 \cdot M', 2 \cdot M') + (2 \cdot M', 1 \cdot M')}$$
$$= (5 \cdot M', 4 \cdot M')$$

Timings... concrete-large scale

Assumption

Security parameter	k=4096
Time for $modExp(k)$	$0.1sec$ (Assumption 2010)
Time for $modMul(k)$	$0.01sec$ (Assumption 2010)
Parallelisation	p
Amount of votes M'	$3'000'000$
CostFunction	$(5 \cdot M', 4 \cdot M')$

Time for dummy-elimination:
$1'500'000 + 120'000 = 1'620'000 \approx \frac{19}{p}$ days.

Voter / Observer: Verification of PET with Voter-Hint

For each vote out of M'

1. homomorphic $\oslash$ each posted Credential with the hinted Credential
2. Verification of Proof of correct decryption

Costs

$$\frac{opCount(\oslash, M') + opCount(verificationCorrectDecryption, M')}{(0,\text{M'})+(4 \cdot M', 2 \cdot M') = (4 \cdot M', 3 \cdot M')}$$

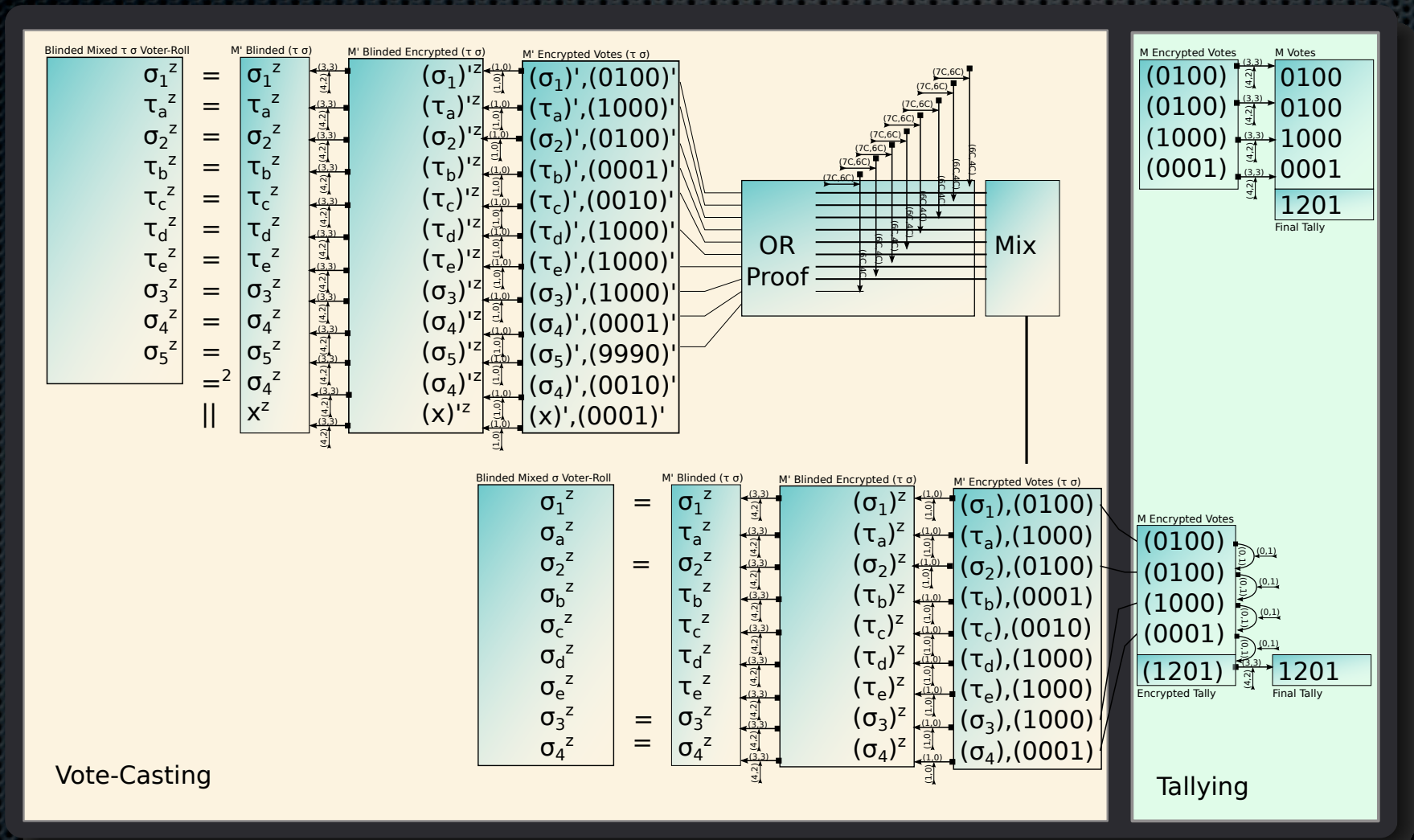
Timings... concrete-large scale

Assumption

Security parameter	k=4096
Time for $modExp(k)$	$0.2sec$ (Assumption 2010)
Time for $modMul(k)$	$0.02sec$ (Assumption 2010)
Parallelisation	p
Amount of votes M'	$3'000'000$
CostFunction	$(4 \cdot M', 3 \cdot M')$

Time for verification of dummy-elimination:
$2'400'000 + 60'000 = 2'460'000 \approx \frac{29}{p}$ days.

Big-Picture


Blinded Mixed τ σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
OR Proof
Mix
M Encrypted Votes
M Votes
Final Tally
Blinded Mixed σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
M Encrypted Votes
Encrypted Tally
Final Tally
Vote-Casting
Tallying

PET with Smith / Weber

For each vote out of M"

1. blind the credential
2. decrypt the credential
3. Proof of the correct decryption
4. (find match)

Costs

$$\frac{opCount(modexp, M'') + opCount(decrypt, M'') + opCount(proofCorrectDecryption, M'')}{(M",0)+(M",2\cdot M'') + (2 \cdot M'', 1 \cdot M'')}$$
$$= (4 \cdot M'', 3 \cdot M'')$$

Timings... concrete-large scale

Assumption

Security parameter	k=4096
Time for $modExp(k)$	$0.1sec$ (Assumption 2010)
Time for $modMul(k)$	$0.01sec$ (Assumption 2010)
Parallelisation	p
Amount of votes M''	$4'000'000$
CostFunction	$(4 \cdot M'', 3 \cdot M'')$

Time for dummy-elimination:
$1'6000'000 + 120'000 = 1'720'000 \approx \frac{20}{p}$ days.

Voter / Observer: Verification of PET with Smith / Weber

For each vote out of M”

1. blind the credential
2. Verification of correct proof of decryption

Costs

$$\frac{opCount(modexp, M'') + opCount(verificationCorrectDecryption, M'')}{(5 \cdot M'', 2 \cdot M'')}$$

Timings... concrete-large scale

Assumption

Security parameter	k=4096
Time for $modExp(k)$	$0.2sec$ (Assumption 2010)
Time for $modMul(k)$	$0.02sec$ (Assumption 2010)
Parallelisation	p
Amount of votes M''	$4'000'000$
CostFunction	$(5 \cdot M'', 2 \cdot M'')$

Time for verification of duplicate / fake / fake-elimination:
$4'000'000 + 160'000 = 4'160'000 \approx \frac{48}{p} days$.

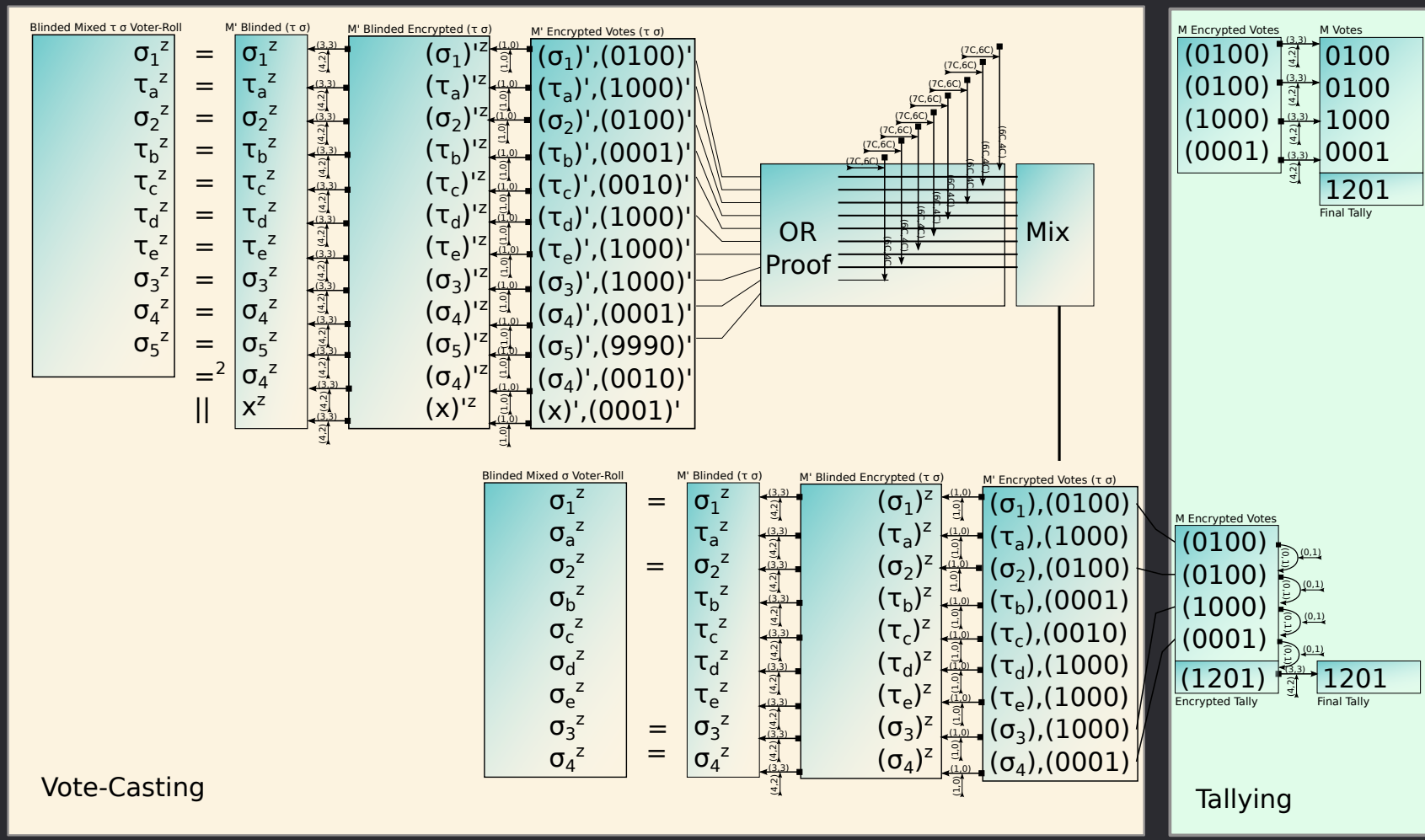
Elimination-Conclusion

Hinted vs. Smith / Weber

Hinted	System: $(5 \cdot M', 4 \cdot M')$
	Voter / Observer: $(4 \cdot M', 3 \cdot M')$
Smith / Weber	System: $(4 \cdot M', 3 \cdot M')$
	Voter / Obeserver: $(5 \cdot M', 2 \cdot M')$

Smith / Weber shifts some load to the setup phase (offline)

Big-Picture


Blinded Mixed τ σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
OR Proof
Mix
M Encrypted Votes
M Votes
Final Tally
Blinded Mixed σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
M Encrypted Votes
Encrypted Tally
Final Tally
Vote-Casting
Tallying

The voters proof

Why

In a coercion resistant system, the vote has to be valid 'Write-In Attack' $\rightarrow$ Vote abstain. The validity has to be proven by the voter

Per vote

1. proof that it is in the set—of size C—of allowed possibilities

Costs

$$\frac{opCount(ORProof, C)}{(6 \cdot C, 4 \cdot C)}$$

Timings... concrete-large scale

Assumption

Security parameter	k=4096
Time for $modExp(k)$	$0.2sec$ (Assumption 2010)
Time for $modMul(k)$	$0.02sec$ (Assumption 2010)
Parallelisation	p
Amount of choices C	50
CostFunction	$(6 \cdot C, 4 \cdot C)$

Time for calculating the proof:
$60 + 4 = 64 \approx \frac{1}{p} minute$.

The system verification

For all casted Votes M'

The system does not have to verify fake / duplicate votes[a]

1. verify OR-proof out of C-choices

[a] If first vote counts

Costs

$$\frac{opCount(VerificationORProof, C, M')}{(7 \cdot C \cdot M', 6 \cdot C \cdot M')}$$

Timings... concrete-large scale

Assumption

Security parameter	k=4096
Time for $modExp(k)$	$0.1sec$ (Assumption 2010)
Time for $modMul(k)$	$0.01sec$ (Assumption 2010)
Parallelisation	p
Amount of votes M'	$3'000'000$
Amount of choices C	50
CostFunction	$(7 \cdot C \cdot M', 6 \cdot C \cdot M')$

Time for verification of proof:
$105'000'000 + 9'000'000 = 114'000'000 secs \approx \frac{1'319}{p} days$
$\approx \frac{3.6}{p} years.$

Timings... concrete-large scale

In order to be verifiable (universal)

Everyone (Observer / Voter) should be able to rectify the systems promise:
So a single voter would calculate more than 7 years in order to get a 'deterministic' result that only valid votes have been accepted.
?Universal verifiable?

Summary

System load

Filter I Fake / Duplicate Elimination: $(4 \cdot M'', 3 \cdot M'')$
OR-Verification: $(7 \cdot C \cdot M', 6 \cdot C \cdot M')$

Filter II Dummy Elimination: $(4 \cdot M', 3 \cdot M')$

$\approx \frac{4}{p}$ *years*

Voter load

Filter I Verify Fake / Duplicate Elimination: $(5 \cdot M'', 2 \cdot M'')$
OR-Proof: $(6 \cdot C, 4 \cdot C)$
OR-Verification: $(7 \cdot C \cdot M', 6 \cdot C \cdot M')$

Filter II Verify Dummy Elimination: $(4 \cdot M', 2 \cdot M')$

$\approx \frac{1}{p}$ *minute* $+ \frac{8}{p}$ *years* ($\frac{49}{p}$ *days* without OR-Verification)

Time-Line

Voting Phase

Setup	Casting	Tallying
1 Month	1 Month	6 hours

feasibility

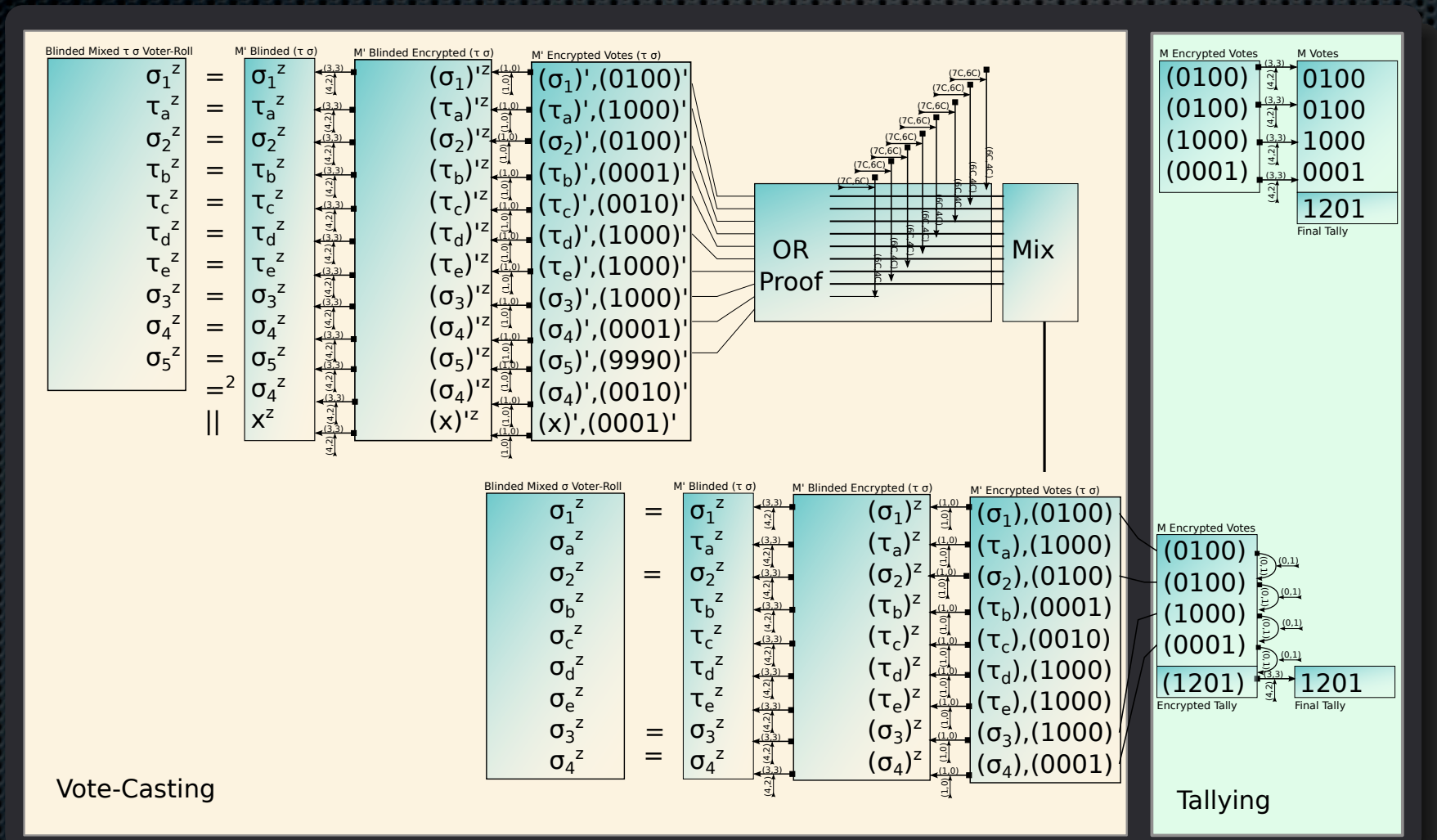
The universal verifiable system...

... is realistic using a parallelization factor of 100 on the server side.

... is unrealistic on the user-side with an acceptable usability.

... is realistic if the user accepts a certain level of trust.

Big-Picture


Blinded Mixed τ σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
OR Proof
Mix
M Encrypted Votes
M Votes
Final Tally
Blinded Mixed σ Voter-Roll
M' Blinded (τ σ)
M' Blinded Encrypted (τ σ)
M' Encrypted Votes (τ σ)
M Encrypted Votes
Encrypted Tally
Final Tally
Vote-Casting
Tallying

Things

System Properties

eginThe system does have a voter rolleginThe system protects itself from fake-votes (no DDOS)eginThe system is linear in respect to voters and voteseginSplitting the voters into smaller groups *augments* overall computing timeeginIf the voter is not able to verify the complete voting process, trust is required. $\rightarrow$ GenèveeginExample: Trip to the loo while observing real voting process.

Things

Alternative to OR-Proof $\rightarrow$ Blinded Rainbow-Table

eginCould be used like Weber just before tallying counting. This would be as expensive as the τ, σ-FiltereginRequires no further proof from the votereginWould probably leak with too much information (Pfitzmann)