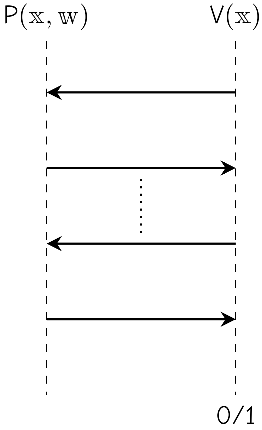
Publicly Verifiable Zero-Knowledge and Post-Quantum Signatures from VOLE-in-the-Head

Carsten Baum[1,2], Lennart Braun[1], Cyprien Delpech de Saint Guilhem[3], Michael Klooß[4], Emmanuela Orsini[5], Lawrence Roy[1], Peter Scholl[1]

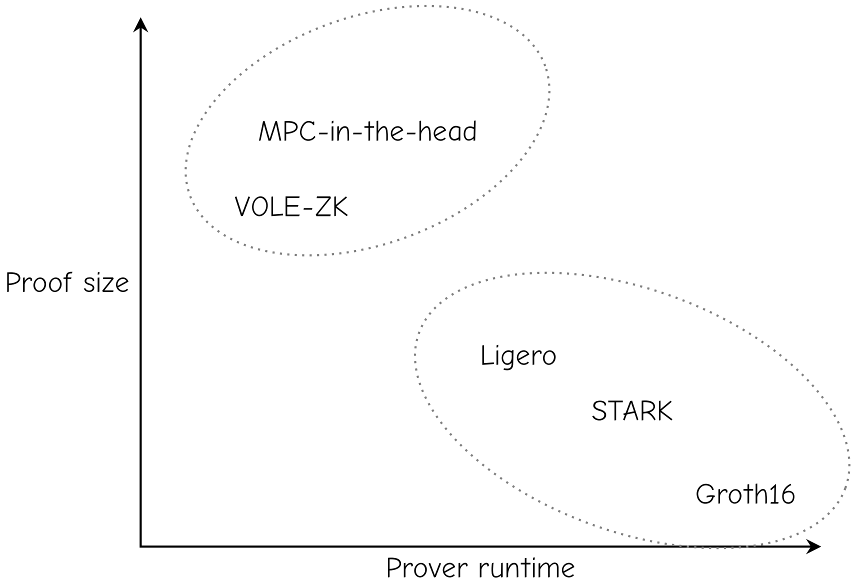
2023-08-23

[1]Aarhus University
[2]Technical University of Denmark
[3]imec-COSIC, KU Leuven
[4]Aalto University
[5]Bocconi University

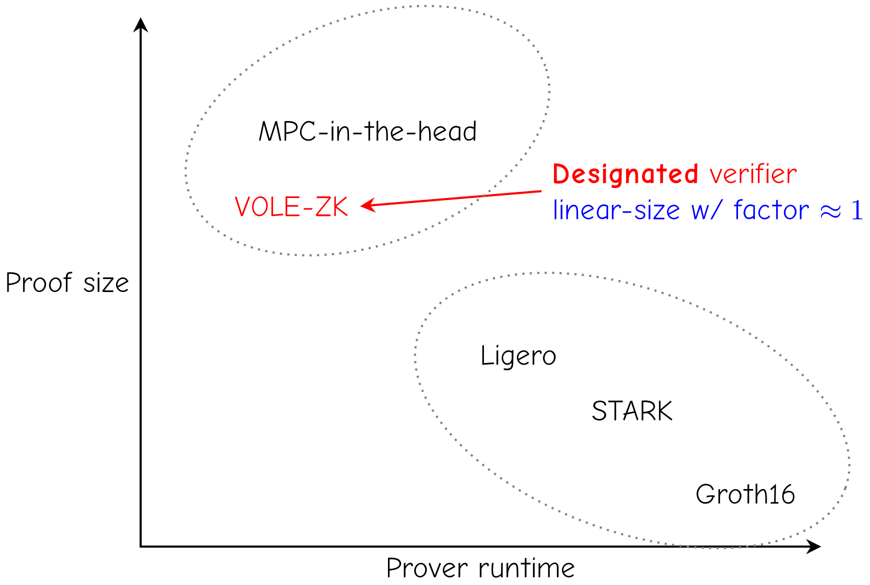
Zero-knowledge (ZK) proofs of knowledge (PoK)



- Prover P convinces verifier V that $(\mathbb{x}, \mathbb{w}) \in \mathcal{R}$.
  - *Zero-knowledge:* Verifier learns nothing else
  - *Knowledge soundness:* Prover *knows* $\mathbb{w}$.



MPC-in-the-head
VOLE-ZK
Proof size
Ligero
STARK
Groth16
Prover runtime


MPC-in-the-head
VOLE-ZK
Designated verifier
linear-size w/ factor ≈ 1
Proof size
Ligero
STARK
Groth16
Prover runtime

VOLE-in-the-Head

Publicly verifiable VOLE-based ZK

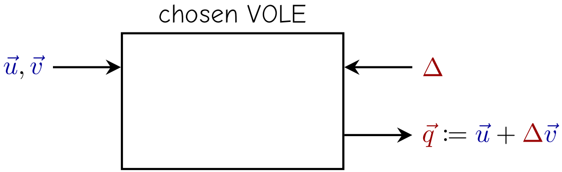
Characteristics:

- Fast: Only symmetric-key crypto
- Simple and flexible: Builds on VOLE ZK proofs
- Secure in the ROM
- Linear-size

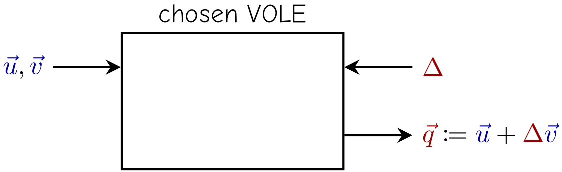
VOLE-ZK

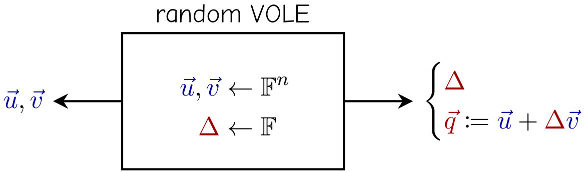
Efficient ZK in the VOLE hybrid model

Vector Oblivious Linear Evaluation (VOLE)

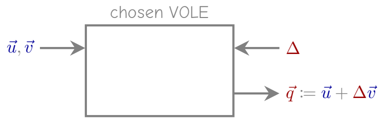


Vector Oblivious Linear Evaluation (VOLE)





Linear commitments from VOLE



- **Commitment:** (q_i, Δ) commits to *message* v_i with *randomness* u_i.

- **Hiding:** $u_i \leftarrow \mathbb{F}$ masks Δv_i perfectly.

- **Binding:** (q_i, Δ) binds to (u_i, v_i)
  - $q_i = u_i + \Delta v_i$ is **linear** in Δ
  - Opening q_i to both $(u_i, v_i) \neq (u_i', v_i') \implies$ must guess Δ.

- **Linear:** VOLE relation is linear in Δ:

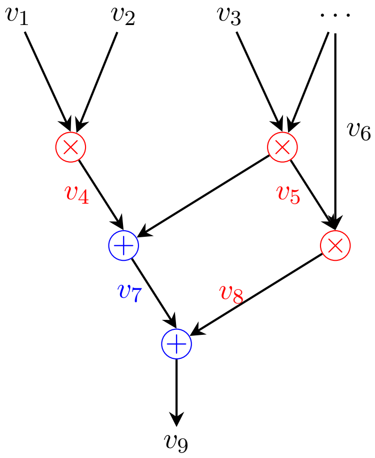
$$q_i + q_i' = (u_i + u_i') + \Delta(v_i + v_i')$$

ZK from VOLE: Commit-then-prove

- Commit to witness variables
- Linear equation checks: ✓
- Quadratic equation checks: Linearization ✓

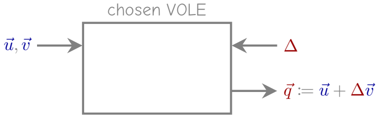
ZK from VOLE: Commit-then-prove

- Commit to witness variables
- Linear equation checks: ✓
- Quadratic equation checks: Linearization ✓
- Circuit-SAT: "Evaluate" gate-by-gate:
  - ADD gates / Linear gates free.
  - MUL gates: Witness contains outputs.



Quadratic equation check from VOLE

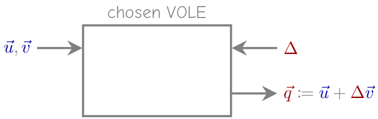
Efficient linearization (Quicksilver [Yan+21])



- P holds: $(u_i, v_i)_i$ over $\mathbb{F}$
- V holds: $(q_i)_i$, Δ over $\mathbb{F}$
- VOLE relation: $u_i + \Delta v_i = q_i$ for all i.
- P's Claim: Quadratic equation $f(v_1, \ldots, v_n) = 0$ holds, $f \in \mathbb{F}[X_1, \ldots, X_n]$

Quadratic equation check from VOLE

Efficient linearization (Quicksilver [Yan+21])



- P holds: $(u_i, v_i)_i$ over $\mathbb{F}$
- V holds: $(q_i)_i$, Δ over $\mathbb{F}$
- VOLE relation: $u_i + \Delta v_i = q_i$ for all i.
- P's Claim: Quadratic equation $f(v_1, \ldots, v_n) = 0$ holds, $f \in \mathbb{F}[X_1, \ldots, X_n]$

$$\begin{aligned} f(q_1, \ldots, q_n) &= f(u_1 + \Delta v_1, \ldots, u_n + \Delta v_n) \\ &= \underbrace{f(u_1, \ldots, u_n)}_{=: a_0 \in \mathbb{F}} + \Delta \cdot \underbrace{\vec{u}^\top F \vec{v}}_{=: a_1 \in \mathbb{F}} + \Delta^2 \underbrace{f(v_1, \ldots, v_n)}_{\overset{!}{=} 0} \end{aligned} \quad (1)$$

- $f(q_1, \ldots, q_n)$ is linear in Δ **if and only if** $f(v_1, \ldots, v_n) = 0$
- $a_0, a_1 \in \mathbb{F}$ are computable from $(u_i, v_i)_i$ alone.

Cost analysis: Arithmetic circuit SAT

- **ZK:** Add random mask to mask a_0, a_1
- **Amortize** N checks $f_i \in \mathbb{F}[X_1, \ldots, X_n]$ via random linear combination.
- **Generalization:** d field elements $a_0, \ldots, a_{d-1}$ for degree d check.

Circuit-SAT

- n_{input} VOLEs for circuit input
- 1 VOLE per multiplication gate
- 2 VOLEs + Openings for masked quadratic check (amortized)

VOLE-in-the-Head

From private coin to public coin

High level idea: (V)COM as (V)OLE

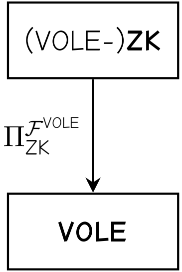
- Δ can be revealed at the end of the protocol. (V has no secrets.)
- Instead of real VOLE, **commit** to VOLE inputs.
  - VOLE is inconvenient, use OT.
  - Use OT-to-VOLE conversion from SoftSpoken OT [Roy22][6]
- In [Cas+19][7] this idea is used for $\binom{2}{1}$-OT, we use all-but-one-OT.
- NB: All VOLEs/VCOMs are "random" and derandomized in the protocol.

---

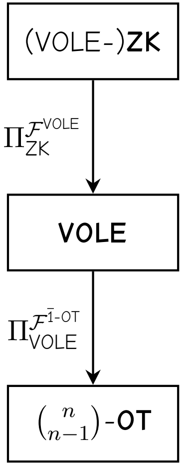
[6] Roy (Crypto'22)

[7] Cascudo, Damgård, David, Döttling, Dowsley, and Giacomelli (Asiacrypt'19)

Pipeline

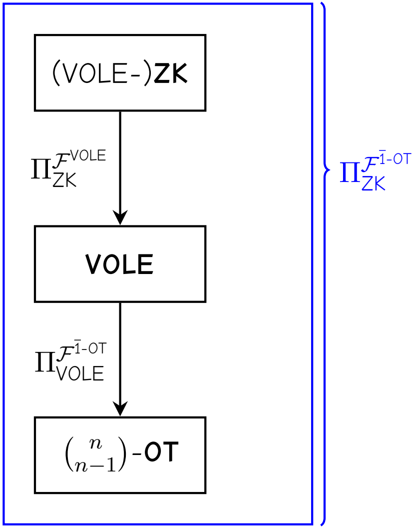


Pipeline



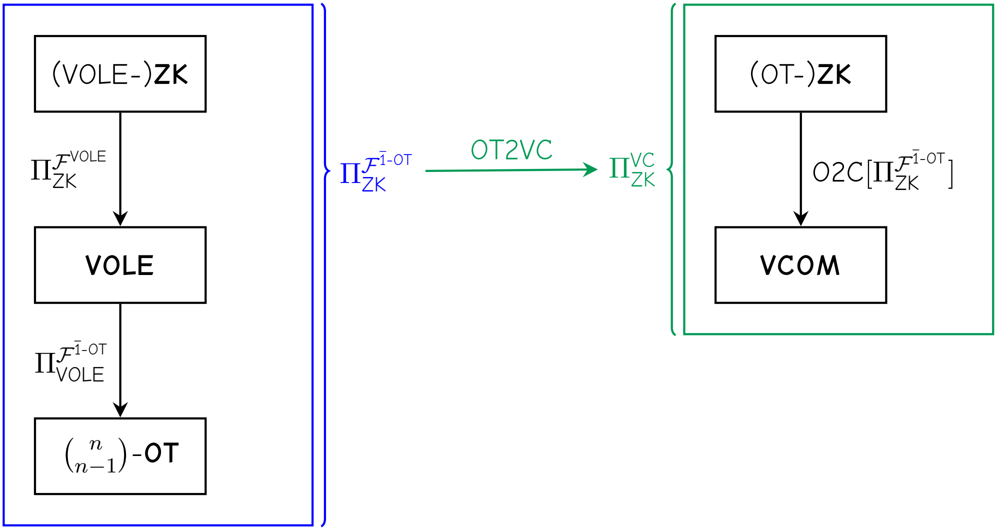
(VOLE-)ZK
$\Pi_{\text{ZK}}^{\mathcal{F}^{\text{VOLE}}}$
VOLE
$\Pi_{\text{VOLE}}^{\mathcal{F}^{\bar{1}\text{-OT}}}$
$\binom{n}{n-1}$-OT

Pipeline



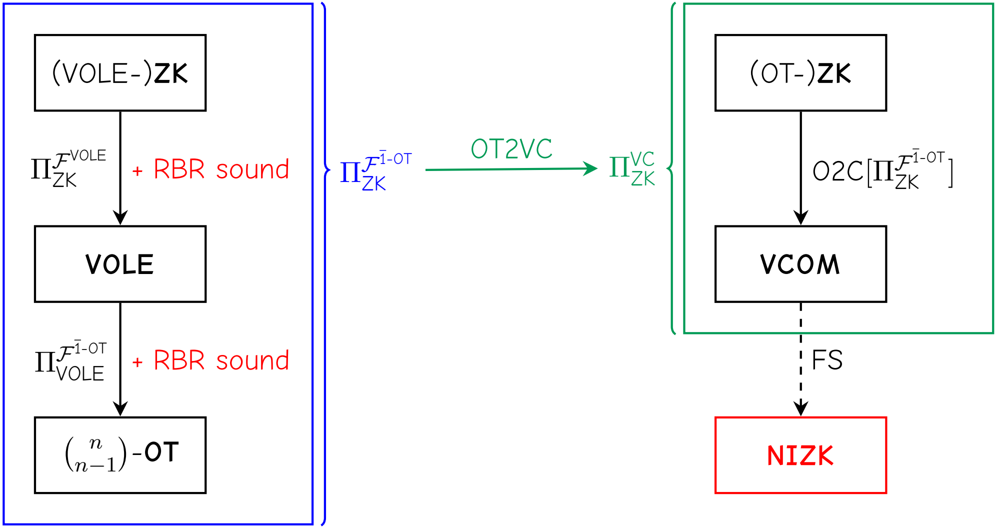
(VOLE-)ZK
$\Pi_{\mathsf{ZK}}^{\mathcal{F}^{\mathsf{VOLE}}}$
VOLE
$\Pi_{\mathsf{VOLE}}^{\mathcal{F}^{\bar{1}\text{-OT}}}$
$\binom{n}{n-1}$-OT
$\Pi_{\mathsf{ZK}}^{\mathcal{F}^{\bar{1}\text{-OT}}}$

Pipeline

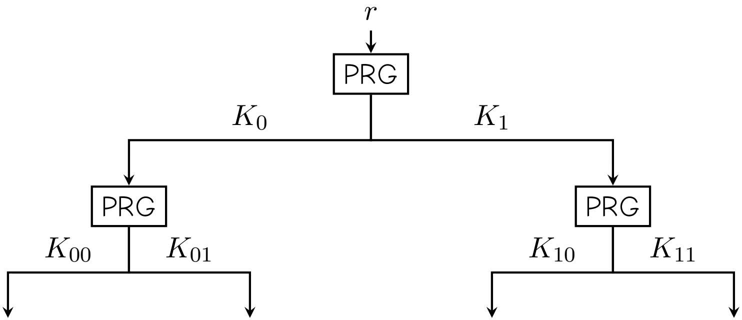


(VOLE-)ZK
$\Pi_{\mathsf{ZK}}^{\mathcal{F}^{\mathsf{VOLE}}}$
VOLE
$\Pi_{\mathsf{VOLE}}^{\mathcal{F}^{\bar{1}\text{-}\mathsf{OT}}}$
$\binom{n}{n-1}$-OT
$\Pi_{\mathsf{ZK}}^{\mathcal{F}^{\bar{1}\text{-}\mathsf{OT}}}$
OT2VC
$\Pi_{\mathsf{ZK}}^{\mathsf{VC}}$
(OT-)ZK
$\mathsf{O2C}[\Pi_{\mathsf{ZK}}^{\mathcal{F}^{\bar{1}\text{-}\mathsf{OT}}}]$
VCOM

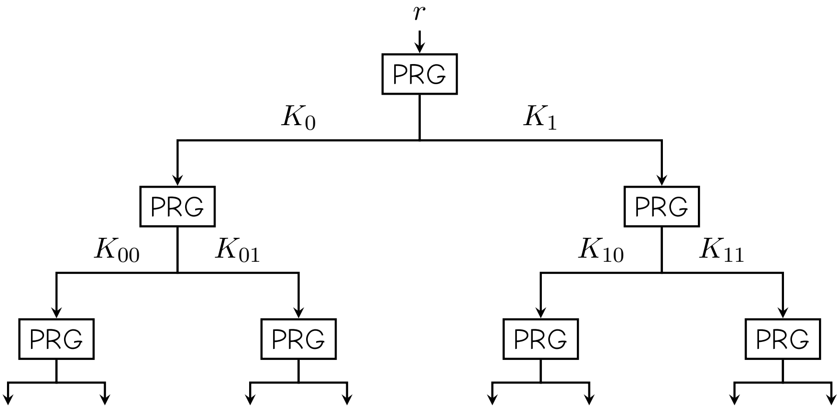
Pipeline


(VOLE-)ZK
$\Pi_{\mathsf{ZK}}^{\mathcal{F}^{\mathsf{VOLE}}}$
+ RBR sound
VOLE
$\Pi_{\mathsf{VOLE}}^{\mathcal{F}^{\bar{1}\text{-OT}}}$
+ RBR sound
$\binom{n}{n-1}$-OT
$\Pi_{\mathsf{ZK}}^{\mathcal{F}^{\bar{1}\text{-OT}}}$
OT2VC
$\Pi_{\mathsf{ZK}}^{\mathsf{VC}}$
(OT-)ZK
$\mathsf{O2C}[\Pi_{\mathsf{ZK}}^{\mathcal{F}^{\bar{1}\text{-OT}}}]$
VCOM
FS
NIZK

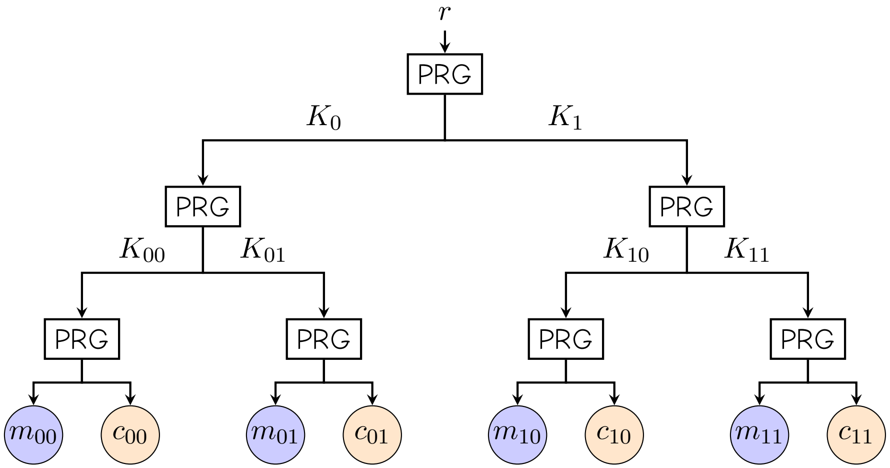
Random all-but-one VCOM as all-but-one OT



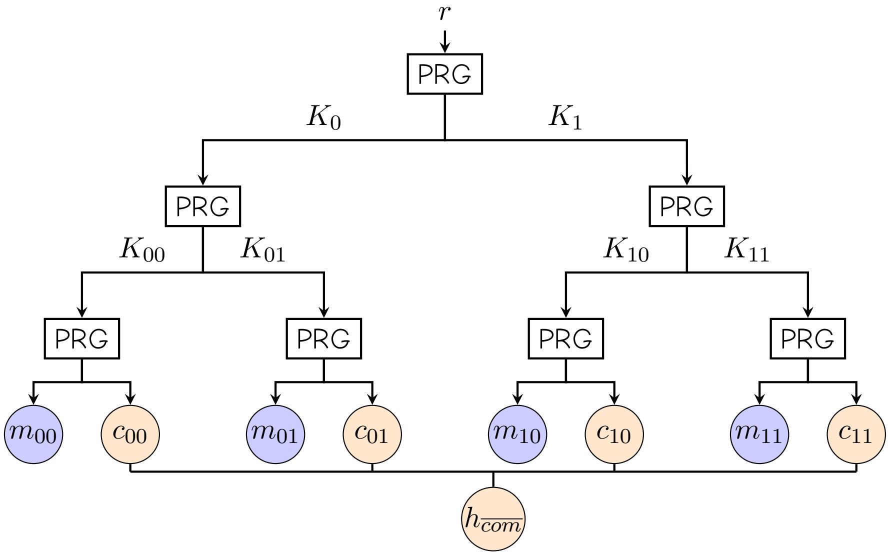
Random all-but-one VCOM as all-but-one OT



Random all-but-one VCOM as all-but-one OT

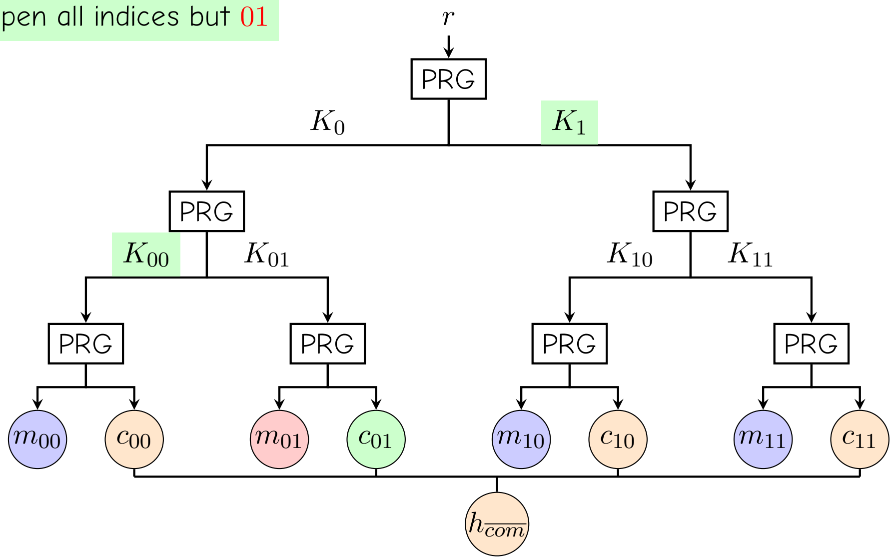


Random all-but-one VCOM as all-but-one OT



Random all-but-one VCOM as all-but-one OT

Open all indices but 01



From $\overline{1}$-OT to VOLE

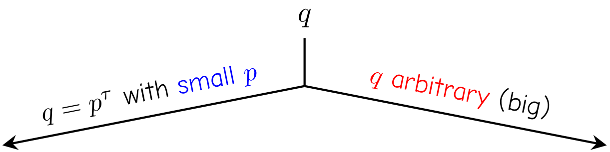
- Conversion from SoftSpoken OT [Roy22]:

  $\rightsquigarrow$ $\binom{n}{n-1}$-OT gives VOLE over $\mathbb{F}_n$.

- Δ easily guessable ($1/n$ chance):

  $\rightsquigarrow$ τ parallel VOLEs with independent Δ_i
  - Easy: Repetition code. (Same input/witness in each instance.)
  - General: Use linear code.
  - Both require consistency check over the parallel instances.

Zero-Knowledge Proofs



τ-repetition code
$v \leftarrow \mathbb{F}_p^\tau$, $u \leftarrow \mathbb{F}_q^\tau$, $\Delta \leftarrow \mathbb{F}_q$
QuickSilver-based ZK over
extension field $\mathbb{F}_q$

General linear code
$\Delta_i \leftarrow \{0, \ldots, n-1\} \subseteq \mathbb{F}_q$ for $i \in [1, \tau]$
subspace VOLE compatible ZK protocol

FAEST

Post-quantum signatures from VOLEitH

[7] Based on submission to NIST PQC Standardization process with Christian Majenz, Shibam Mukherjee, Sebastian Ramacher, Christian Rechberger as additional co-authors.

FAEST — Construction

Fiat–Shamir-based signature scheme

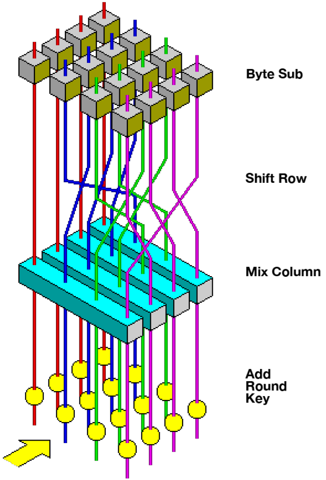
- Keypair sk, vk with $vk = (x, y) = (x, \mathsf{AES}(sk, x))$.
- Use VOLEitH to prove knowledge of

$$\{sk \mid \mathsf{AES}(sk, x) = y\}$$

- Fiat–Shamir transformation $\rightsquigarrow$ signature scheme

FAEST: Handling AES-128

- AES rounds are $\mathbb{F}_2$-linear, except **byte sub**. Substitute $x \in \mathbb{F}_{2^8}$ by
  - $S(0) = 0$,
  - $S(x) = x^{-1}$ in $\mathbb{F}_{2^8}$.
- For ZK:[a] $x \cdot y = 1 \iff y = x^{-1}$
- Overall:
  - 1600 bit witness (for key + AES circuit)
  - 200 quadratic constraints over $\mathbb{F}_{2^8}$



[a]For better efficiency, restrict to keys where $S(0)$ is never used.

FAEST performance

Specification	Sign/Verify	Size
FAEST-128s	≈ 8 ms	≈ 5.0 kB
FAEST-128f	≈ 1 ms	≈ 6.3 kB
FAEST-256s	≈ 27 ms	≈ 22.1 kB
FAEST-256f	≈ 3 ms	≈ 28.4 kB

Optimized implementation on notebook (Ryzen 7 5800H, 3.2 GHz)

Conclusion

- VOLE-ZK: Lightweight, fast, linear-size
- VOLEitH: public-coin, NIZK via Fiat–Shamir transformation
- FAEST signature: Conservative security, reasonable performance

Conclusion

Thank you!

- VOLE-ZK: Lightweight, fast, linear-size
- VOLEitH: public-coin, NIZK via Fiat–Shamir transformation
- FAEST signature: Conservative security, reasonable performance

References I

[Cas+19] Ignacio Cascudo, Ivan Damgård, Bernardo David, Nico Döttling, Rafael Dowsley, and Irene Giacomelli. "Efficient UC Commitment Extension with Homomorphism for Free (and Applications)". In: **ASIACRYPT 2019, Part II**. Vol. 11922. LNCS. Dec. 2019.

[Roy22] Lawrence Roy. "SoftSpokenOT: Quieter OT Extension from Small-Field Silent VOLE in the Minicrypt Model". In: **CRYPTO 2022, Part I**. Vol. 13507. LNCS. Aug. 2022.

[Yan+21] Kang Yang, Pratik Sarkar, Chenkai Weng, and Xiao Wang. "QuickSilver: Efficient and Affordable Zero-Knowledge Proofs for Circuits and Polynomials over Any Field". In: **ACM CCS 2021**. Nov. 2021.