Restoring Soundness of the Orion Proof System & More

Thomas den Hollander, Daniel Slamanig
Research Institute CODE
Universität der Bundeswehr München

Orion: Zero Knowledge Proof with Linear Prover Time (CRYPTO'22)

- Proof system with
  - $O(N)$ prover time
  - $O(\log(N))$ verifier time* & proof size
- Two main innovations
  - Algorithm for linear-time encodable linear code
    - Previously inverse polynomial or impractical
  - Proof composition with code-switching
    - Based on tensor code PCS ([BCG+17, BCG20, GLS+ (Brakedown)])
    - Take $O(\sqrt{N})$ verifier time & proof size, add outer proof
    - Not limited to same linear code, no proving hash functions

Our work

- Orion is unsound, both with and without zk
  - Demonstrate using practical attack
- Propose a solution
  - Preserve linear prover time complexity
  - No hash functions inside outer SNARK circuit
  - No new commitments/rounds to protocol
- For zero-knowledge
  - Propose a linear-time encodable zero-knowledge linear code
  - Increased prover time
  - Significantly smaller verifier time & proof size

PCS: Commitment phase

- **Commit(pp, φ; r) → C**
    1. Matrix of coefficients W
    2. Encode each row, add random vectors
        - $D_i = E_C(W_i) + r_i \,||\, r_i$
    3. Encode each column
        - $E_j = E_C(D_j)$
    4. Merkle Commitment
        - $C = Commit_M(E)$

$$\psi(x) \qquad \begin{bmatrix} — & W_1 & — \\ — & W_2 & — \\ & \vdots & \\ — & W_k & — \end{bmatrix} \qquad 1$$

$$\psi(x) = \begin{bmatrix} 1 \\ x^k \\ \cdots \\ x^{(k-1)k} \end{bmatrix}^T \left[\psi \quad D = \begin{bmatrix} E_C(W_1) + \vec{r_1} & \vec{r_1} \\ E_C(W_2) + \vec{r_2} & \vec{r_2} \\ \vdots & \vdots \\ E_C(W_k) + \vec{r_k} & \vec{r_k} \end{bmatrix} \right] \begin{bmatrix} 1 \\ x \\ \cdots \\ x^{k-1} \end{bmatrix}$$

Evaluation phase

- Tensor code PCS
  - P sends linear combination of encoded rows
    - Row: $D_i = E_C(W_i) + r_i \,||\, r_i$
    - $c_\gamma = \langle \gamma, D \rangle$
  - V checks that result is a codeword
    - $c_\gamma = E_C(W_\gamma) + r_\gamma \,||\, r_\gamma$
  - V checks linear combination at random column set J
    - $c_\gamma = \langle \gamma, D \rangle$ for $j \in J$
  - Evaluation same, but using x_0 instead of γ
- Orion adds outer SNARK
  - Commit to c_γ, build inside CP-SNARK and compare only at $j \in J$
  - Also sample row set I
  - Encode columns $D_{\cdot j}$ inside CP-SNARK, compare with E at $(i, j) \in I \times J$

Evaluation phase

- **Eval(pp, C, $X = x_0 \otimes x_1$, $y = x_0^T W x_1$, φ)**
    1. V sends challenge vector γ
    2. P computes
        a. $c_\gamma = \langle \gamma, D \rangle$
        b. $W_\gamma = \langle \gamma, W \rangle$
        c. $r_\gamma = \langle \gamma, R \rangle$
        d. And sends $C_{c\gamma} = \mathrm{Commit}(c_\gamma)$
    3. V sends column set J, making sure $j \in J \Rightarrow j+n \notin J$
    4. P commits to CP-SNARK witness: W_γ, r_γ, columns $D_{\bullet j}$ for $j \in J$
    5. V sends row set I
    6. P computes CP-SNARK proof π
        a. Check $c_\gamma = E_C(W_\gamma) + r_\gamma \,||\, r_\gamma$, compare to $C_{c\gamma}$ at $j \in J$
        b. Check $c_\gamma = \langle \gamma, D_{\bullet j} \rangle$ at columns $j \in J$
        c. Compare $E_C(D_{\bullet j})$ to *C* at $(i,j) \in I \times J$
    7. V checks π and openings

Issue due to zero-knowledge…

…

2.d. P sends $C_{c\gamma}$ = Commit(c_γ)

3. V sends column set J, *making sure* $j \in J \Rightarrow j+n \notin J$

4. P commits to CP-SNARK witness: W_γ, r_γ, columns $D_{\bullet j}$ for $j \in J$

…

6.a. Check $c_\gamma = E_C(W_\gamma) + r_\gamma \,||\, r_\gamma$, compare to $C_{c\gamma}$ at $j \in J$

…

- Prover can choose r_γ, <u>after</u> J was sampled
- $E_C(W_\gamma) + r_\gamma$ and r_γ are never opened at the same offset
- Simply choose suitable r_γ!
- Evaluate to any point

…but the issue persists without zk

	…
2.d.	P sends $C_{c\gamma} = \text{Commit}(c_\gamma)$
3.	V sends column set J, ~~making sure $j \in J \Rightarrow j+n \notin J$~~
4.	P commits to CP-SNARK witness: W_γ, $\cancel{r_\gamma}$, columns $D_{\bullet j}$ for $j \in J$
	…
6.a.	Check $c_\gamma = E_C(W_\gamma) \cancel{+ r_\gamma} \cancel{\| r_\gamma}$, compare to $C_{c\gamma}$ at $j \in J$
	…

- J is known before commitment to W_γ
- Find W_γ such that $E_C(W_\gamma)=c_\gamma$ at J
- Solve linear system
- Evaluate to any point, with overwhelming probability

How to fix?

…

2.d. P sends $C_{c\gamma}$ = Commit(c_γ)

Commit to W_γ, r_γ before knowing J

3. V sends column set J, making sure $j \in J \Rightarrow j+n \notin J$

4. P commits to CP-SNARK witness: W_γ, r_γ, columns $D_{\bullet j}$ for $j \in J$

…

6.a. Check $c_\gamma = E_C(W_\gamma) + r_\gamma \,||\, r_\gamma$, compare to $C_{c\gamma}$ at $j \in J$

…

J must be known when committing to $D_{\bullet j}$, otherwise not succinct!

Commit twice?

- We could simply add another round of commitments
- Open commitment inside outer SNARK?
  - Outer SNARK circuit grows
  - Increased proof size from additional commitment
- Another round of CP-SNARK commitments?
  - Two (succinct) commitments, increasing verifier time & proof size
    - Verifier time potentially mitigated using batching

Our solution

- J has two purposes, which can be separated!
- Use J to check linear combinations of rows
- Use J' to compare with commitment

	…
2.d.	P sends $C_{c\gamma}$ = Commit(c_γ)
3.	V sends column set J
4.	P commits to CP-SNARK witness: W_γ, r_γ, columns $D_{\bullet j}$ for $j \in J$
	…
5.	V sends row set I *and column set J'*
6.a.	Check $c_\gamma = E_C(W_\gamma) + r_\gamma \mid\mid r_\gamma$, *compare to $C_{c\gamma}$ at $j \in J'$*
	…

How to deal with zero-knowledge?

…

2.d. P sends $C_{c\gamma}$ = Commit(c_γ)

3. V sends column set J

If this has j…

4. P commits to CP-SNARK witness: W_γ, r_γ, columns $D_{\bullet j}$ for $j \in J$

…

… this should not have j + n

5. V sends row set I *and column set J'*

6.a. Check $c_\gamma = E_C(W_\gamma) + r_\gamma \,||\, r_\gamma$, *compare to* $C_{c\gamma}$ *at* $j \in J'$

…

- Still unsound: P knows V won't query c_γ at $j \pm n$ for $j \in J$

New zero-knowledge code

- No restrictions on J, J': uniformly random
- Use polynomial to hide any |J| + |J'| evaluations
  - Fixed degree, O(1)
  - No constant term
- Retains minimum relative distance
- General transformation

$$E_{C,ZK}(y; r)_i = \left(E_C(y) \mid\mid E_C(y)\right)_i + \sum_{j>0} r_i\, i^j$$

& More…

- New knowledge soundness & zero-knowledge proof
  - Simulator needed to know X before committing to polynomial
- Challenge space now logarithmic
  - *Sampling γ actually requires O(√N) work from verifier
  - [DP23]: Use $(1\ \gamma_1)\otimes(1\ \gamma_2)\otimes\ldots\otimes(1\ \gamma_{\log(k)})$ instead
- Multi-point opening
- Explicit consideration of Fiat-Shamir

New zk-SNARK: Scorpius

- Proof system with
  - O(N) prover time
  - O(log(N)) verifier time & proof size
- Compared to Orion
  - Increased prover time
  - Faster verifier & smaller proof size
- Rigorous knowledge soundness & zero-knowledge proofs

Conclusion

- Orion is unsound, both with & without ZK
  - Attack efficient and perfect/negligible failure probability
- We provide a new zero-knowledge code
  - General transformation that retains minimum relative distance
  - Linear time encodable
- We propose Scorpius, with
  - Knowledge soundness fix without any overhead
    - Retaining linear prover
  - ZK code with increased prover, smaller verifier time & proof size

Thanks for listening!

Any questions?

ePrint: https://eprint.iacr.org/2024/1164.pdf