Australian National University

Flashproofs

Efficient Zero-Knowledge Arguments of Range & Polynomial Evaluation with Transparent Setup

Nan Wang and Sid Chi-Kin Chau

Asiacrypt 2022

Our contributions

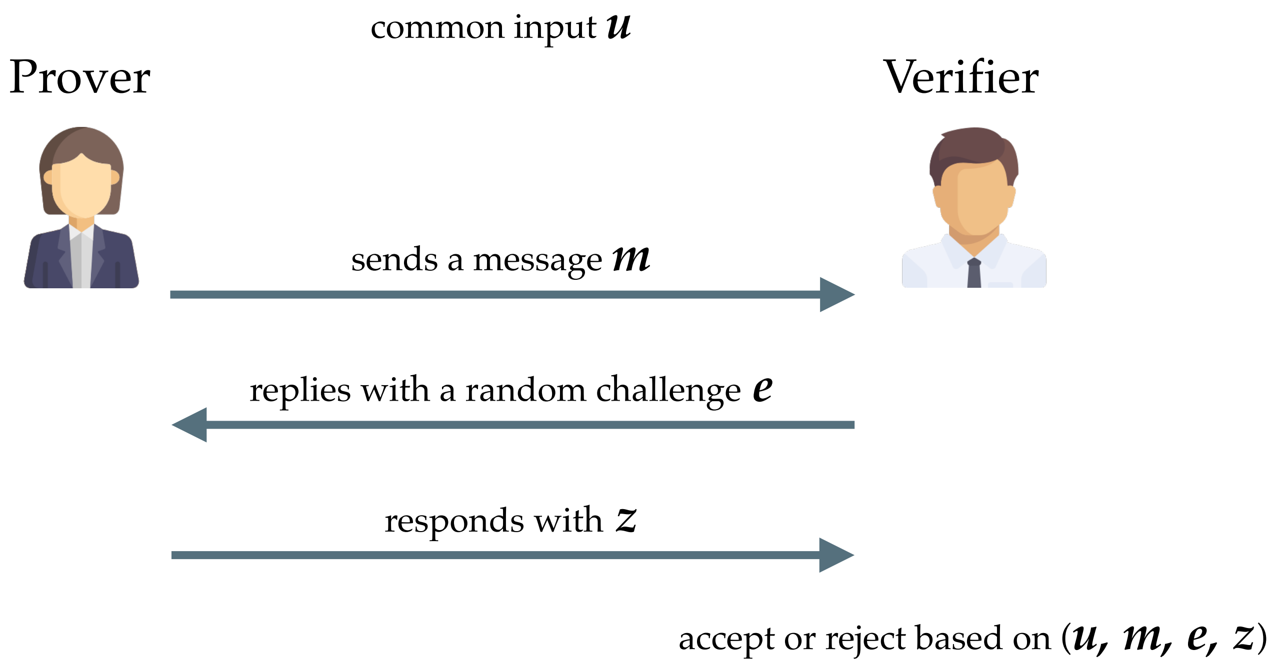
- Discrete logarithm (DL) assumption
- Transparent (non-trusted) setup
- Σ-protocol
- No pairing operations
- Zero-Knowledge Argument of Range
    - proves a committed value x lies in a specific range
    - $x \in [0,\ 2^{32} - 1], x \in [0,\ 2^{64} - 1]$
- Zero-Knowledge Argument of Polynomial Evaluation
    - proves two committed values x and y satisfy a public polynomial relation
    - $y = x^{64} + 1, y = \prod_{i=1}^{64}(x - i) = 0$

Zero-Knowledge Argument of Knowledge

- A zero-knowledge proof allows a prover to convince a verifier of the truth of a statement without revealing any secret information.
- An argument is computationally sound proof that no probabilistic polynomial-time provers are able to deceive a verifier into falsely accepting it.
- More formally, given an NP-language L, a prover aims to convince a verifier of knowing a witness ω for a statement $u \in L$.
    - Completeness: A prover can convince a verifier of $u \in L$, if $u \in L$
    - Soundness: A prover cannot convince a verifier of $u \in L$, if $u \notin L$.
    - Zero-knowledge: The proof reveals nothing except the truth that $u \in L$.

Σ-Protocol

- An efficient approach for constructing honest verifier zero-knowledge proofs
- 3-round interactive protocol
- Non-interactive protocol via Fiat-Shamir transform



Zero-Knowledge Arguments of Range

- Prove a committed value is in the range $[0,\ 2^N - 1]$
- A new variant of the bit-decomposition approach
- Tailored to confidential transactions on blockchain platforms
- Prove the transfer amount is non-negative and the balance has sufficient funds for the transfer amount
- $O(N^{\frac{2}{3}})$ communication and verification efficiency
- Achieve comparable verification gas costs to that of the most efficient zkSNARK (Groth16)
- Support the aggregation of multiple arguments for further efficiency improvement

Bit-Decomposition

- $y = \sum_{i=0}^{N-1} 2^i b_i,\ b_i \in \{0,1\} \Longrightarrow y \in [0, 2^N - 1]$

- Bulletproof is a popular generic-purpose zero-knowledge argument, which can instantiate a zero-knowledge range argument.

- It uses a variant of the bit-decomposition approach to achieve O(log N) communication cost and O(N) proving and verification.

Our Technique

Prover:

$$y = \sum_{i=0}^{N-1} 2^i b_i \Longrightarrow \begin{pmatrix} 2^0 b_0 & \cdots & 2^{K-1} b_{K-1} \\ 2^K b_K & \cdots & 2^{K+K-1} b_{K+K-1} \\ \vdots & \ddots & \vdots \\ 2^{(L-1)K} b_{(L-1)K} & \cdots & 2^{(L-1)K+K-1} b_{(L-1)K+K-1} \end{pmatrix} = \begin{pmatrix} w_0 & \cdots & w_{K-1} \\ w_K & \cdots & w_{K+K-1} \\ \vdots & \ddots & \vdots \\ w_{(L-1)K} & \cdots & w_{(L-1)K+K-1} \end{pmatrix}$$

$$\begin{pmatrix} w_0 & \cdots & w_{K-1} \\ w_K & \cdots & w_{K+K-1} \\ \vdots & \ddots & \vdots \\ w_{(L-1)K} & \cdots & w_{(L-1)K+K-1} \end{pmatrix} \cdot \begin{pmatrix} e_0 \\ \vdots \\ e_{K-1} \end{pmatrix} + \begin{pmatrix} r_0 \\ \vdots \\ r_{L-1} \end{pmatrix} = \begin{pmatrix} v_0 \\ \vdots \\ v_{L-1} \end{pmatrix} \qquad v_l = \sum_{k=0}^{K-1} w_{lK+k} e_k + r_l$$

Verifier: $$f_l = \sum_{k=0}^{K-1} 2^{lK+k} e_k - v_l = \sum_{k=0}^{K-1} (2^{lK+k} - w_{lK+k}) e_k - r_l$$

Our Technique

$$(1)\quad f_l \cdot v_l \stackrel{?}{=} \underbrace{\sum_{k=0}^{K-1} w_{lK+k}(2^{lK+k} - w_{lK+k})e_k^2}_{=0} + \sum_{k=0,j=1}^{k=K-2,j=K-1} t_{k,j}e_{k,j} + \sum_{k=0}^{K-1} q_k e_k + q_K \qquad e_{k,j} = e_k \cdot e_j,\ k \neq j$$

$$w_{lK+k}(2^{lK+k} - w_{lK+k}) = 0 \Longrightarrow w_{lK+k} \in \{0, 2^{lK+k}\}$$

$$(2)\quad \sum_{l=0}^{L-1} v_l \stackrel{?}{=} \sum_{k=0}^{K-1} s_k e_k + s_K, \quad s_k = \sum_{l=0}^{L-1} w_{lK+k}, \quad s_K = \sum_{l=0}^{L-1} r_l \qquad (3)\quad y \stackrel{?}{=} \sum_{k=0}^{K-1} s_k$$

When $K \approx \lceil N^{\frac{1}{3}} \rfloor$, both communication and verification costs achieve the minimum.

$$|\Pi| = L + 2K + \frac{K(K-1)}{2} + 4 = \lceil \frac{N}{K} \rceil + \frac{K^2}{2} + \frac{3K}{2} + 4$$

Our Optimisation Technique

64-bit: $(e_k)_{k=0}^{K-1} = (e^{-1}, e, e^4, e^5)$

=0

$$f_l \cdot v_l = w_{10}e^{10} + \xcancel{w_8e^8 + w_2e^2} + w_{-2}e^{-2}$$

$$+w_9e^9 + w_6e^6 + \underbrace{w_5e^5}_{e^4e} + \underbrace{w_4e^4}_{e^5e^{-1}} + w_3e^3 + w_1e + w_{-1}e^{-1} + \underbrace{w_0}_{ee^{-1}}$$

(a) Values of (L, K)

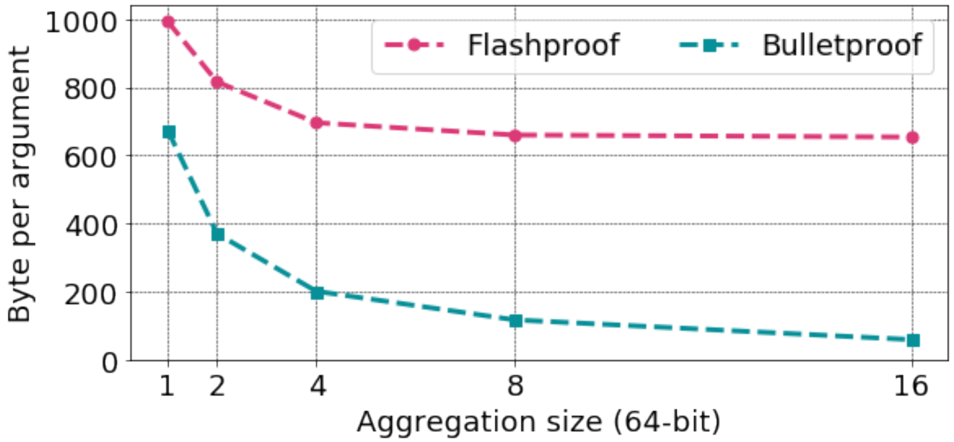
N	8-bit	16-bit	32-bit	64-bit
L	4	8	11	16
K	2	2	3	4

(b) Comparison of range arguments for 64-bit

Type	Prover No. of Exp ($\mathbb{G}$)	Verifier No. of Exp ($\mathbb{G}$)	Proof Size (Byte)
Original Work	197	33	1090
Optimised Work	146	30	994
Saving	51 (25.9%)	3 (9.1%)	96 (8.8%)

Efficiency Comparison with Bulletproof

	N	8	10	12	14	16	18	20	22	32	52	64
Prover No. of Exp ($\mathbb{G}$)	Bulletproof	116	238	238	238	238	480	480	480	480	962	962
	This work	21	24	27	30	33	36	39	42	80	122	146
Verifier No. of Exp ($\mathbb{G}$)	Bulletproof	56	98	98	98	98	180	180	180	180	342	342
	This work	11	12	13	14	15	16	17	18	22	27	30
Proof Size (Byte)	Bulletproof	482	546	546	546	546	610	610	610	610	674	674
	This work	**385**	**417**	**449**	**481**	**513**	**545**	**577**	**609**	738	898	994



Gas Cost Comparison

Type	Transparent Setup	Gas Cost	Ether	USD	Proof Size (Byte)
zkSNARK (Groth16)	✗	220,100	0.0033	$5.8	192
This work (32-bit)	✓	233,250	0.0035	$6.1	738
This work (64-bit)	✓	314,140	0.00471	$8.2	994
CKLR21 (32-bit)*	✓	330,868	0.00496	$8.7	827
CKLR21 (64-bit)*	✓	454,301	0.00681	$11.9	964
zkSNARK (SONIC, Helped)*	✗	492,000	0.00738	$12.9	385
zkSNARK (SONIC, Unhelped)*	✗	655,000	0.00983	$17.2	1155
zkSNARK (BCTV14)	✗	773,124	0.0116	$20.2	288
Bulletproofs (32-bit)	✓	2,046,252	0.03069	$53.6	610
Bulletproofs (64-bit)	✓	3,703,549	0.05555	$96.9	674

- Gas price:15 gwei (Etherscan), Ether price: $1745 USD (Coindesk), UTC 11:15 am 12/09/2022
- * indicates the gas costs are estimated

Zero-Knowledge Argument of Polynomial Evaluation

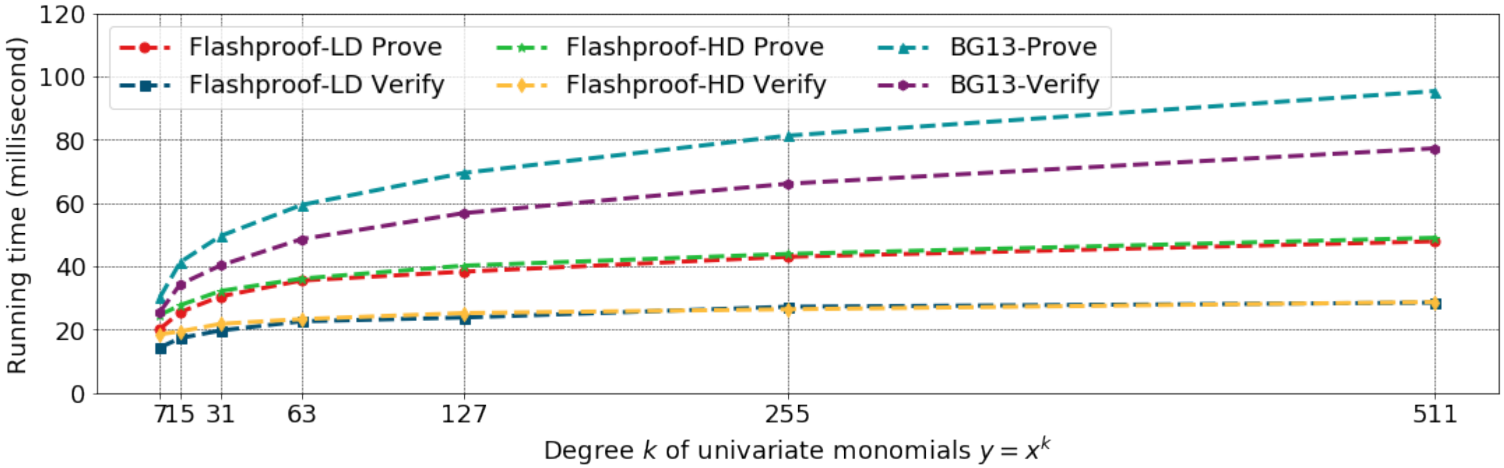
- Builds on the work of Bayer & Groth (BG13, Eurocrypt 13)
- Two zero-knowledge protocols optimised for the polynomials of lower-degree $N \in [3, 2^9]$ and higher-degree $N > 2^9$.
- Instantiates the membership argument by constructing a polynomial function $y = \prod_{i=1}^{I}(x - i) = 0$ for a public list $i \in [1, \ldots, I]$.
- Combined with the range argument, it can be used to satisfy complex mathematical relations by leveraging the Maclaurin series

$$y = \sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} \ldots \qquad y = e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} \ldots$$

- Achieves $O(\sqrt{\log N})$ efficiency in communication and verification (group exponentiation) for the aggregation of multiple arguments satisfying different polynomial relations and sharing the same inputs

Efficiency Comparison

Type	Bulletproofs	BG13	This Work (4.2) Lower-Deg $N \in [3, 2^9]$	This Work (4.3) Higher-Deg $N > 2^9$
Prover No. of Exp ($\mathbb{G}$)	$15N + 2\log N - 10$	$8\log N - 4$	$4\log N + 2$	$3\log N + 3\sqrt{\log N} + 2$
Verifier No. of Exp ($\mathbb{G}$)	$5N + 2\log N + 10$	$7\log N - 1$	$2\log N + 7$	$\log N + 3\sqrt{\log N} + 6$
Proof Size No. of Elements	$2\log N + 8$ ($\mathbb{G}$) 5 ($\mathbb{Z}_p$)	$4\log N - 2$ ($\mathbb{G}$) $3\log N$ ($\mathbb{Z}_p$)	$\log N + 3$ ($\mathbb{G}$) $\log N + 3$ ($\mathbb{Z}_p$)	$2\sqrt{\log N} + 3$ ($\mathbb{G}$) $\boxed{\log N} + \sqrt{\log N} + 4$ ($\mathbb{Z}_p$)



Thanks!