One Tree to Rule Them All: Optimizing GGM Trees and OWFs for Post-Quantum Signatures

Carsten Baum, Ward Beullens, **Shibam Mukherjee**, Emmanuela Orsini,
Sebastian Ramacher, Christian Rechberger, Lawrence Roy, Peter Scholl

10th December 2024

So Far…

NIST post-quantum digital signature algorithm finalists (2022)

- Dilithium (Lattice)
- Falcon (Lattice)
- SPHINCS+ (Stateless-hash)

2-out-of-3 based on Lattice hardness assumption!

Symmetric primitives use well studied structures

- ➢ Allows quicker long-term confidence!

SPHINCS+ [9] Signature Size

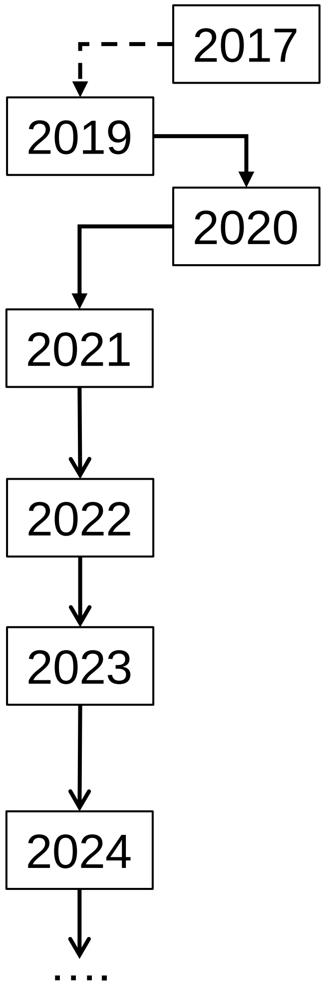
- L1 – 7.8 kB
- L3 – 16.2 kB
- L5 – 29.7 kB

New NIST post-quantum signature additional around (2023)

- ➢ Primarily focus on non-lattice hardness assumption!

So Far… (Cont.)

Post Quantum Signatures based on symmetric primitives based on MPCitH and VOLEitH paradigm (L1)



Picnic [1]
~32 kB (LowMC)

BBQ [2]
~51 kB (AES)

Picnic-3 [12]
~12.5 kB (LowMC)

Banquet [3]
~19 kB (AES)

Helium [5]
~10 kB (AES)

Rainier [4]
~5 kB ($Rain_3$)

FAEST [6][10]
5 kB (AES)
4.5 kB (AES-EM)

AIMER [11]
~4.1 kB (Aim)

FAESTER **(This work)**
4.5 kB (AES)
4.1 kB (AES-EM)

MandaRain **(This work)**
2.8 kB ($Rain_3$)

MPC-in-the-Head paradigm
VOLE-in-the-Head paradigm

So Far… (FAEST at a high level)

- Signer knows a sk used in signing the message m
- Signer proves to the verifier in ZK

  "I know $\mathsf{sk} \in \{0,1\}^\lambda$ such that $\mathrm{OWF}_{\mathsf{sk}}(x) = y$, where x and y are pk"

- Verifier verifies the signature (ZK proof) with the corresponding pk
- Zero-Knowledge proof in VOLE-in-the-Head (VOLEitH) paradigm
  Quicksilver proof [8]
- Non-interactive proof with Fiat-Shamir transformation
- Currently in the Round-2 of NIST additional PQ digital signature process

A Vole (Wikipedia)

Our Contributions

- Faster and smaller FAESTER(-EM) signature

- Optimized one tree Batched all-but-one Vector Commitment (BAVC)
  Signature size reduction for the same signing runtime

- Uniform keys
  Zero input S-Box now possible

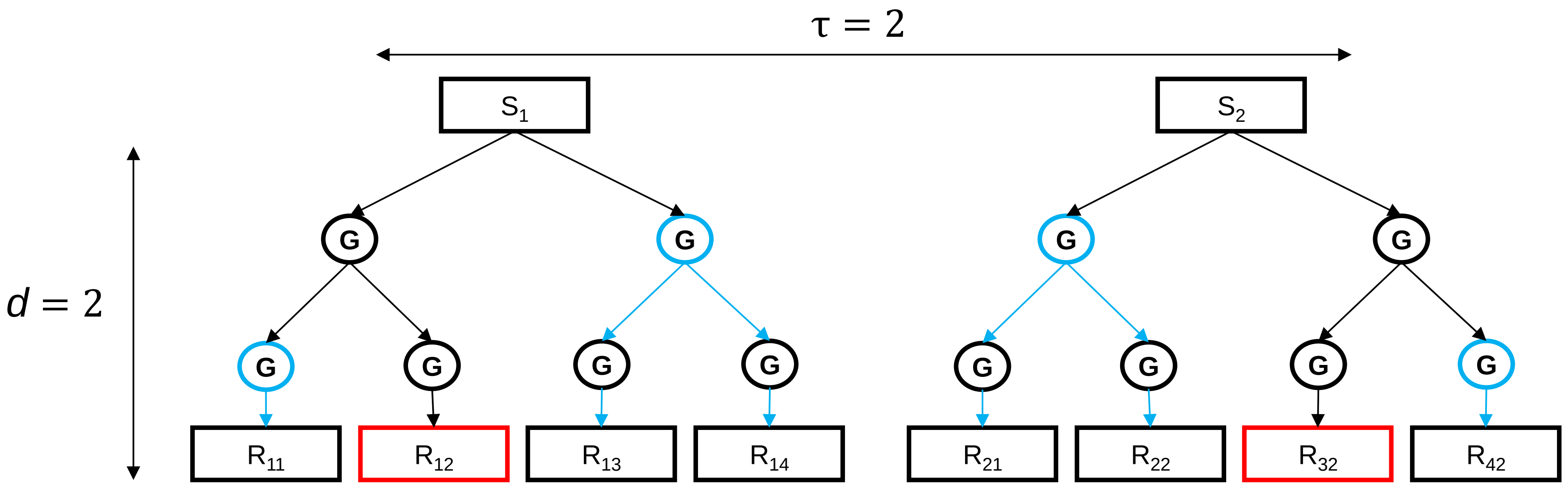
- Degree-7 constraints in proof system
  Smaller signature with AES as OWF

- Use of optimized OWFs like Rain [4] and MQ [7]
  - Even smaller and faster VOLEitH signatures

- Extensive parameter exploration for future improvement directions

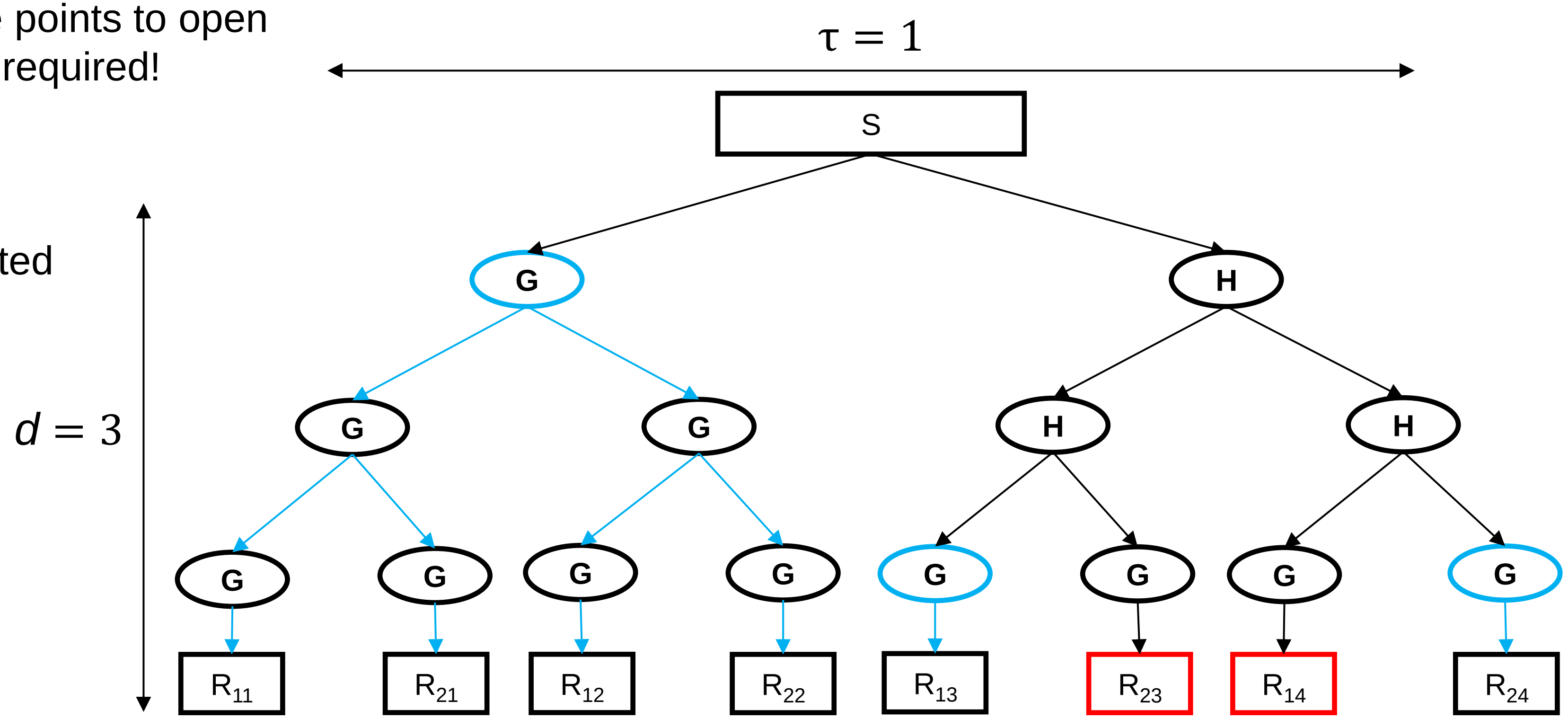
Batched all-but-one Vector Commitment (BAVC)

- AVCs use Merkle Trees to generate the “in-the-head OWF computation shares”
- τ tree repetitions required to reach λ bit security soundness
- Each repetition requires $log(2^d)$ communication, where d is the depth of the tree
- Example: 4 field elements communicated

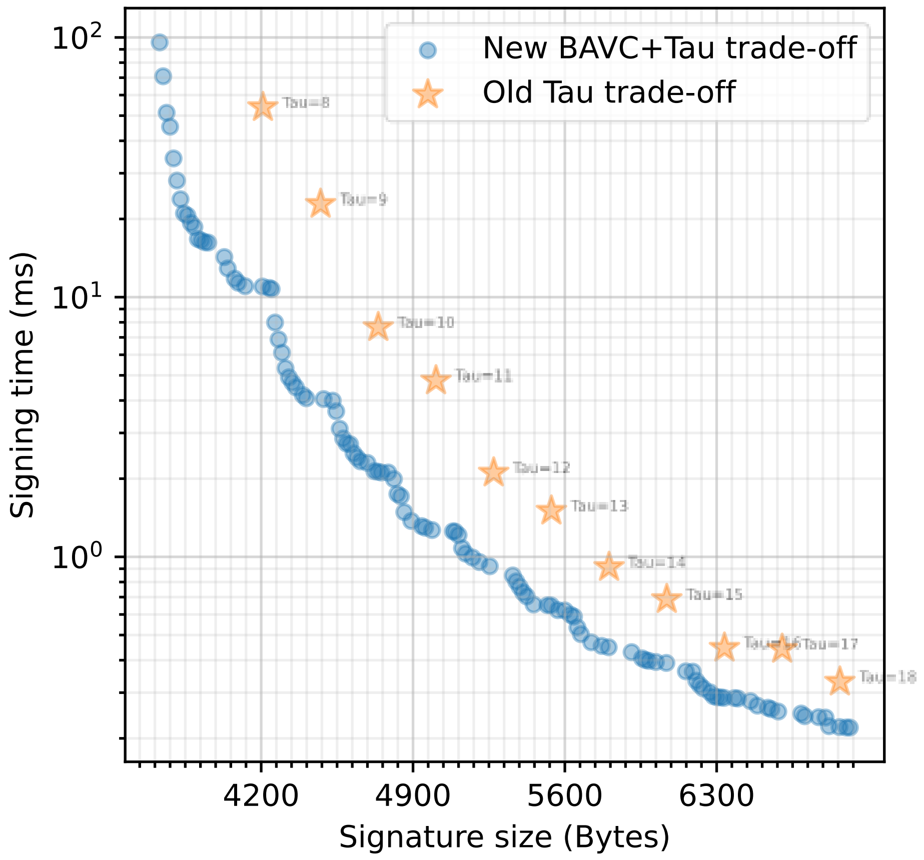


Optimizing BAVC

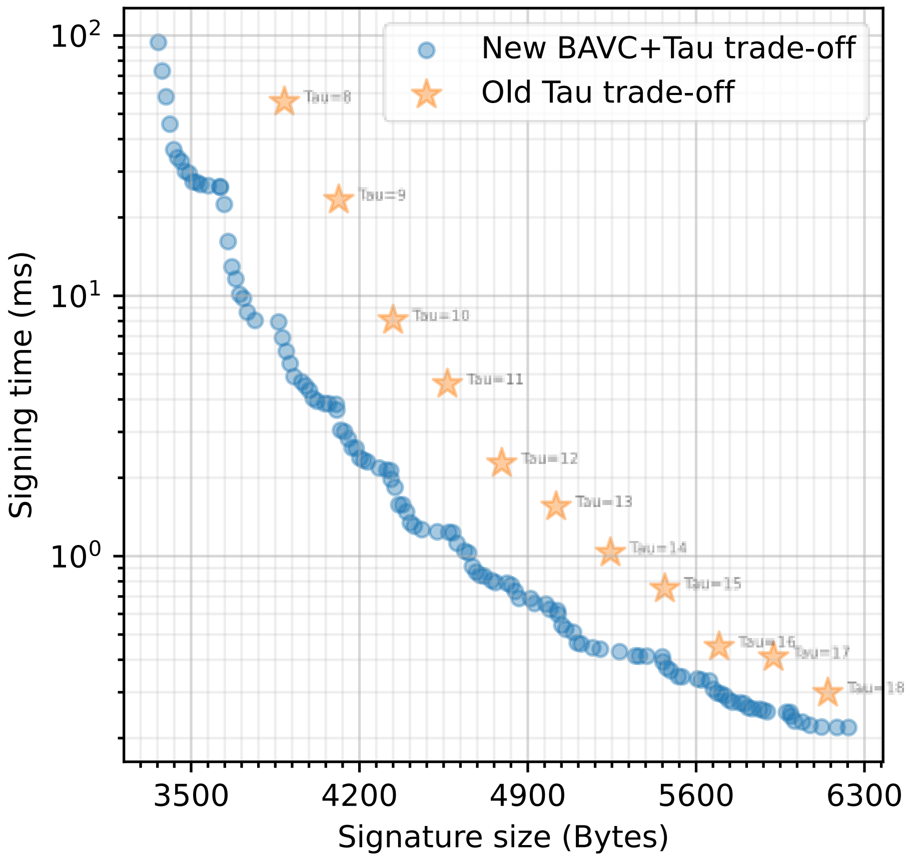
- Use one big tree
- Interleave the random seeds from the τ trees
- Rejection sampling when selecting the points to open
  Security preserved as proof-of-work required!
- Less than $\tau \times log(2^d)$ communication
- Example: 3 field elements communicated



Optimized BAVC



FAEST-128 vs FAESTER-128



FAEST-EM-128 vs FAESTER-EM-128

Uniform AES keys in FAESTER(-EM)

- AES S-box is the non-linear function

  $S : x \mapsto x^{254} \in F_{2^8}$

- Prover proves $y = S(x)$

- Degree-2 constraint check $xy = 1$

- Problems

  - x and y must be non-zero!

  - Key restriction such that S-box has non-zero inputs only
    - Rejection sampling
    - 1-2 bits loss in *sk* entropy

Uniform AES keys in FAESTER(-EM) (Cont.)

Key Observation! (Solution)

- $xy^2 = y \wedge x^2y = x$

- Checks
  - $x = 0 \wedge y \neq 0$
  - $y = 0 \wedge x \neq 0$

- Degree-3 constraint? More Communication?

- Squaring is linear in F_2

- Proof size stays the same!

FAEST-d7

- Prove AES via Degree-7 constrains, variant of the Quicksilver proof system

- Express AES S-Box as Degree-7 circuits over F_2
  - $S : x \mapsto x^{254} \in F_{2^8}$
  - 254 has a hamming weight of 7!

- Combine with meet-in-the-middle approach

- Prover only commits to every other AES round instead of every round
  - Reduction in non-linear communication
  - **5% reduction** in signature size

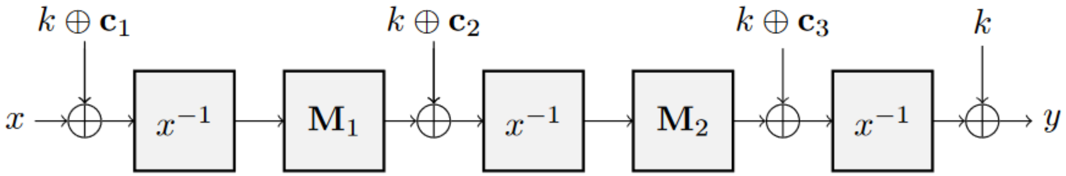
- Improved Signature Sizes (L1)

  - FAEST-d7 – 4.7 kB (FAEST – 5 kB)
  - FAESTER-d7 – 4.3 kB (FAESTER – 4.5 kB)

Signatures with Optimized OWFs (MandaRain)

- MPCitH and VOLEitH signatures use OWFs to proof knowledge of the sk
- Small number of non-linear operations in OWFs is “ideal”
  - Reduces the signature size
- MPCitH/VOLEitH friendly Rain [4] OWF
  - Block cipher
  - $S : x \mapsto x^{-1} \in F_{2^\lambda}$
- Rain_3 with 3 rounds (2 non-linear const.)
- Rain_4 with 4 rounds (3 non-linear const.)
  More conservative!
- One of the smallest signature size
  2.8 kB (Rain_3 L1)
- Very fast signing and verification time
  0.34 ms (Rain_3 L1)

Mandarin (Wikipedia)



The Rain [4] encryption function with 3 rounds.

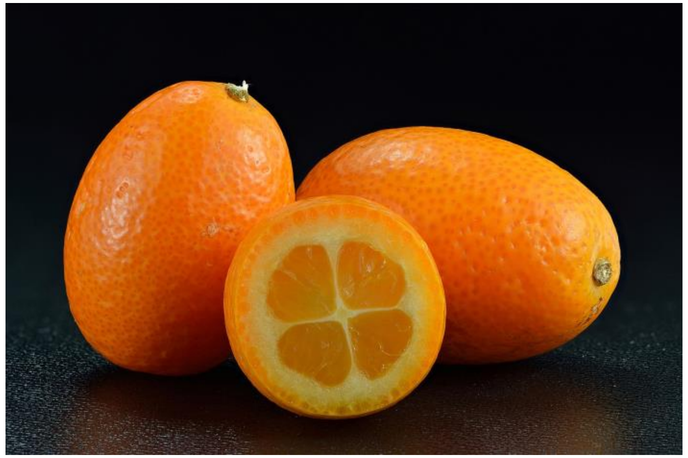
Signatures with Optimized OWFs (KuMQuat)

- MQ problem as OWF for VOLEitH signature
- $F \in \mathrm{MQ}_{n,m,q}$ is a multivariate map over F_q with n variables and m equations

$$\left(y_i := x^{\mathbf{T}} \cdot \mathbf{A}_i \cdot x + b_i^{\mathbf{T}} \cdot x\right)_{i \in [m]}$$

Non-Linear operation!

- $\mathbf{A}_i \in F_q^{n \times n}$ (randomly sampled upper triangular matrix)
- $b_i \in F_q^n$ (uniformly sampled vectors)
- $(\mathbf{A}_1, \ldots, \mathbf{A}_m, b_1, \ldots, b_m) \leftarrow$ Generator(seed)
- Given $F \in \mathrm{MQ}_{n,m,q}$ and $\boldsymbol{y} = (y_1, \ldots, y_m)$, find x, such that $F(x) = \boldsymbol{y}$



Kumquat (Wikipedia)

Signatures with Optimized OWFs (KuMQuat) (Cont.)

- Signature scheme construction

  $\mathsf{sk} \leftarrow (x, \mathsf{seed})$
  $\mathsf{pk} \leftarrow (y, \mathsf{seed})$

  Chosen MQ versions
  - MQ-2^1 with $q = 2^1$
  - MQ-2^8 with $q = 2^8$
  - Direct field extension to 2^λ

- Smallest signature size among all NIST Round-1 VOLEitH and MPCitH signature schemes
  - **2.5 kB** (MQ-2^1 L1)

- Fast signing and verification time
  - **0.53 ms** (MQ-2^1 L1)

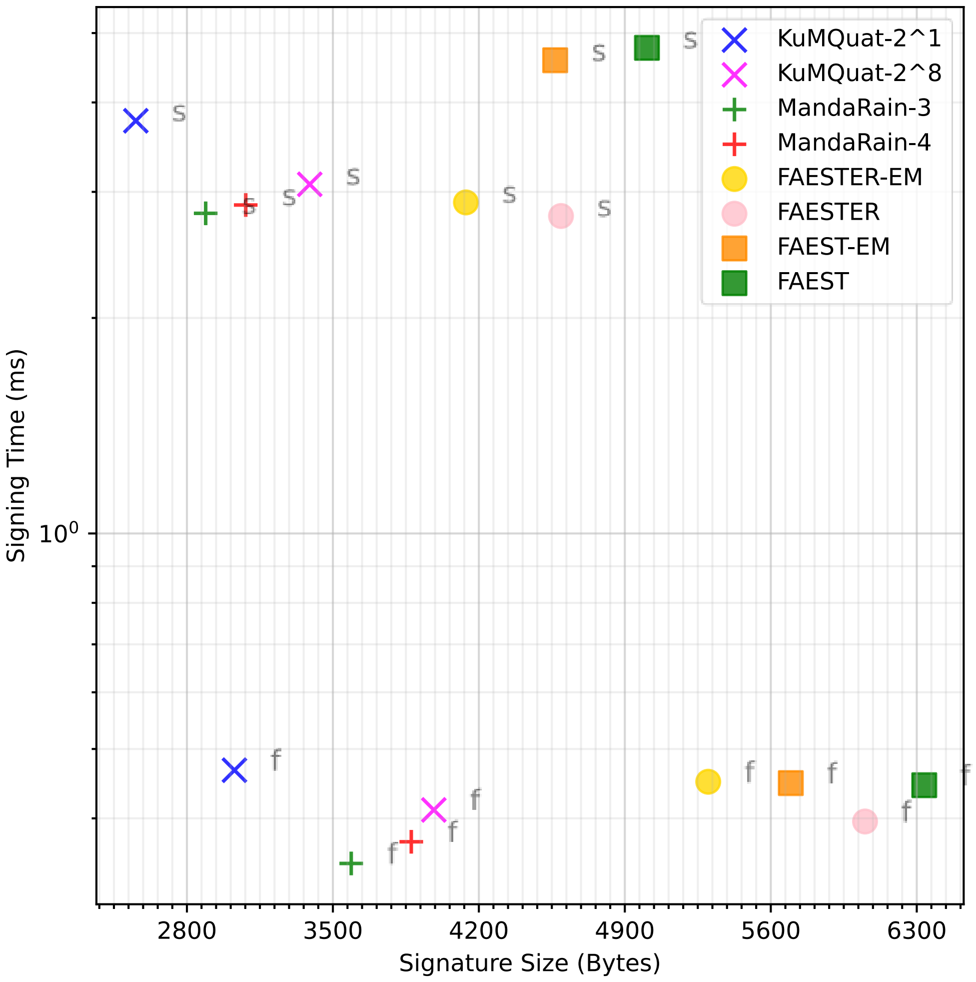
- More conservative MQ parameters possible without affecting the signature size considerably!

Benchmark (Highlights)

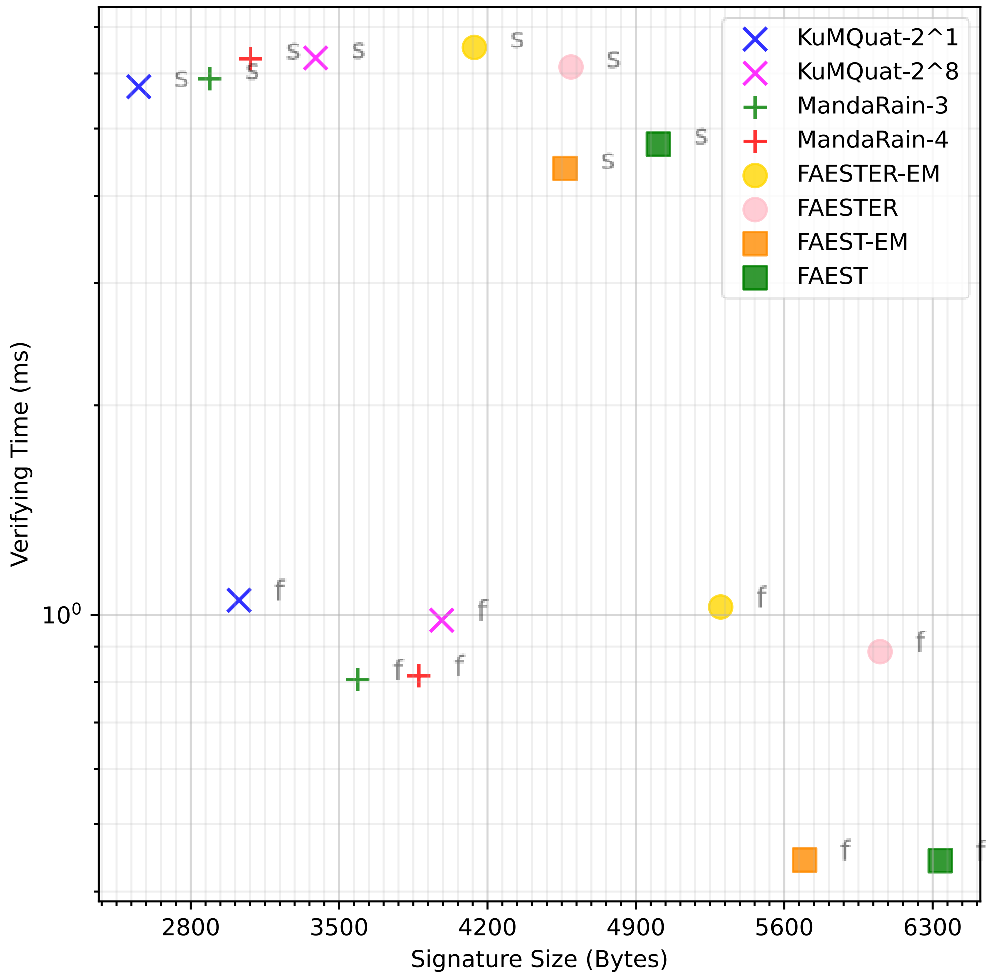
Scheme L1	Keygen (ms)	Sign (S/F) (ms)	Verify* (S/F) (ms)	Signature Size (S/F) (kB)
FAEST	0.0006	4.381 / 0.404	4.102 / 0.395	5006 / 6336
FAESTER	0.0006	3.282 / 0.433	4.467 / 0.610	**4594** / 6052
FAEST-EM	0.0005	4.151 / 0.446	4.415 / 0.474	4566 / 5696
FAESTER-EM	0.0005	3.005 / 0.422	4.386 / 0.609	**4170** / 5444
FAEST-d7	-	-	-	4790 / 6020
FAESTER-d7	-	-	-	**4374** / 5732
MandaRain-3	0.0018	**2.8 / 0.346**	5.895 / 0.807	**2890** / 3588
MandaRain-4	0.0026	**2.876 / 0.371**	6.298 / 0.817	3052 / 3876
KuMQuat-2^1	0.173	4.305 / 0.539	4.107 / 0.736	**2555** / 3028
KuMQuat-2^8	0.174	3.599 / 0.4	4.053 / 0.623	2890 / 3588

S and F are the slow and the fast versions, respectively.
* When not using one big tree optimization, sign/verify times are same!

Benchmark (in-house comparison)

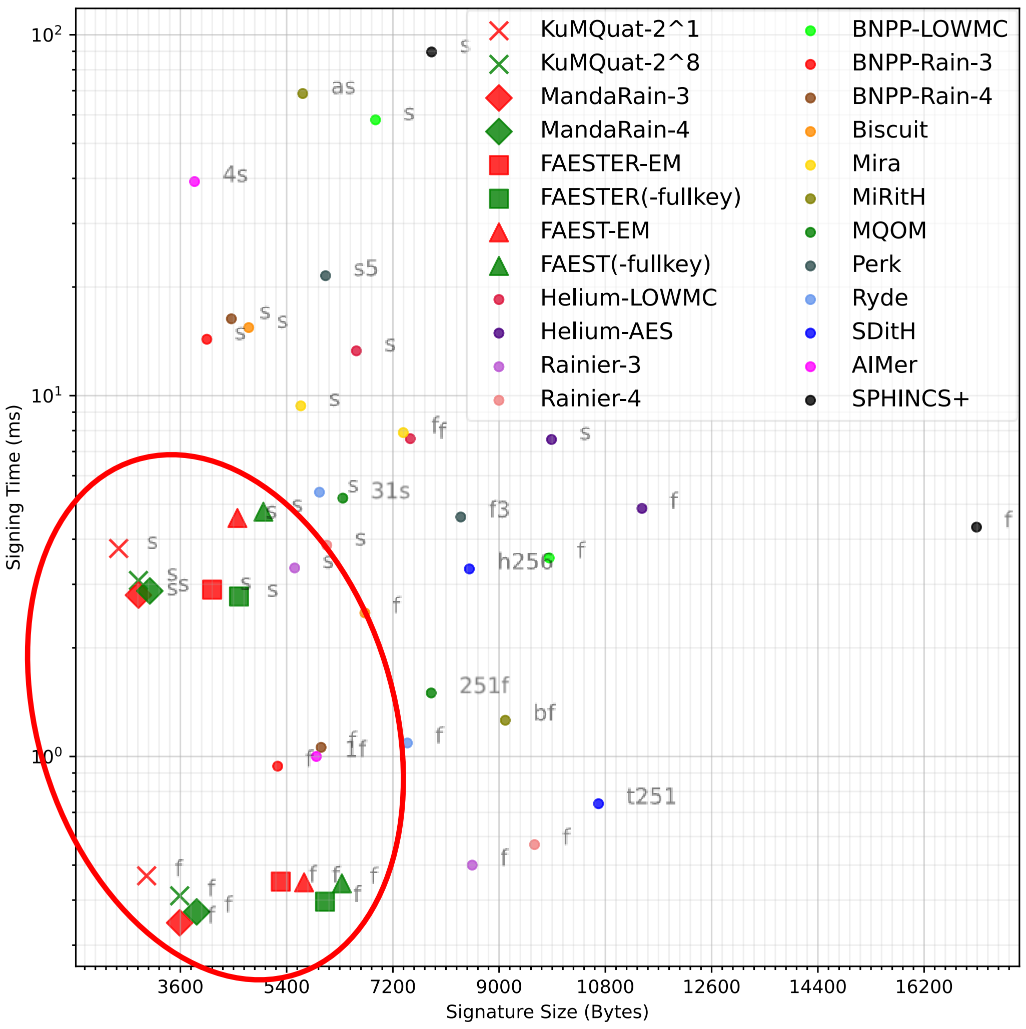


L1 Signing Overview

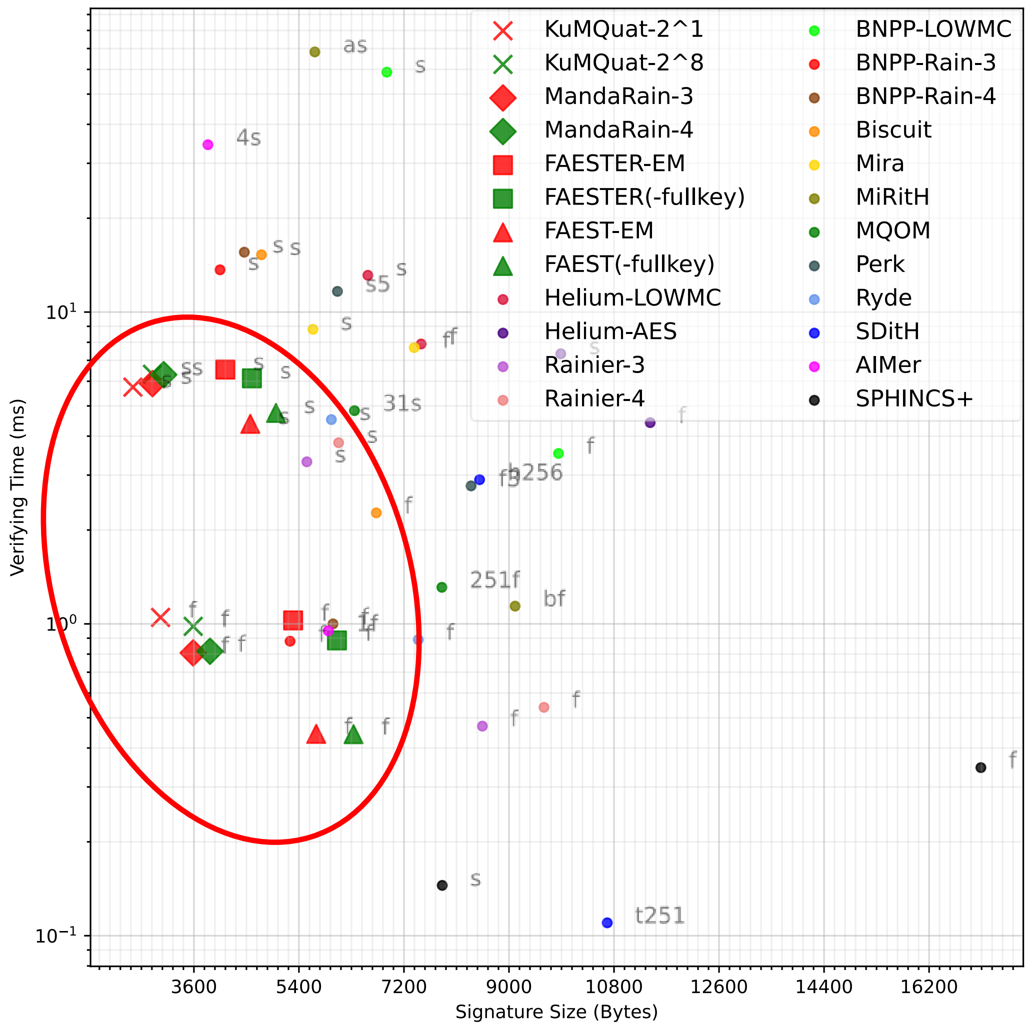


L1 Verify Overview

Benchmark (NIST Round-1 comparison)



L1 Signing Overview



L1 Verify overview

Signature names are according to the NIST Additional Signature Round-1 submissions

Find the full paper here

https://eprint.iacr.org/2024/490

Questions?

Bibliography

[1] Chase, M., Derler, D., Goldfeder, S., Orlandi, C., Ramacher, S., Rechberger, C., Slamanig, D. and Zaverucha, G., 2017, October. Post-quantum zero-knowledge and signatures from symmetric-key primitives. In *Proceedings of the 2017 acm sigsac conference on computer and communications security* (pp. 1825-1842).

[2] de Saint Guilhem, C.D., De Meyer, L., Orsini, E. and Smart, N.P., 2019, August. BBQ: using AES in picnic signatures. In *International Conference on Selected Areas in Cryptography* (pp. 669-692). Cham: Springer International Publishing.

[3] Baum, C., de Saint Guilhem, C.D., Kales, D., Orsini, E., Scholl, P. and Zaverucha, G., 2021, May. Banquet: short and fast signatures from AES. In *IACR International Conference on Public-Key Cryptography* (pp. 266-297). Cham: Springer International Publishing.

[4] Dobraunig, C., Kales, D., Rechberger, C., Schofnegger, M. and Zaverucha, G., 2022, November. Shorter signatures based on tailor-made minimalist symmetric-key crypto. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security* (pp. 843-857).

[5] Kales, D. and Zaverucha, G., 2022. Efficient lifting for shorter zero-knowledge proofs and post-quantum signatures. *Cryptology ePrint Archive*.

Bibliography

[6] Baum, C., Braun, L., de Saint Guilhem, C.D., Klooß, M., Orsini, E., Roy, L. and Scholl, P., 2023, August. Publicly verifiable zero-knowledge and post-quantum signatures from vole-in-the-head. In *Annual International Cryptology Conference* (pp. 581-615). Cham: Springer Nature Switzerland.

[7] Benadjila, R., Feneuil, T. and Rivain, M., 2024, July. MQ on my mind: Post-quantum signatures from the non-structured multivariate quadratic problem. In *2024 IEEE 9th European Symposium on Security and Privacy (EuroS&P)* (pp. 468-485). IEEE.

[8] Yang, K., Sarkar, P., Weng, C. and Wang, X., 2021, November. Quicksilver: Efficient and affordable zero-knowledge proofs for circuits and polynomials over any field. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security* (pp. 2986-3001).

[9] Bernstein, D.J., Hülsing, A., Kölbl, S., Niederhagen, R., Rijneveld, J. and Schwabe, P., 2019, November. The SPHINCS+ signature framework. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security* (pp. 2129-2146).

[10] Baum, C., Braun, L., de Saint Guilhem, C.D., Klooß, M., Majenz, C., Mukherjee, S., Orsini, E., Ramacher, S., Rechberger, C., Roy, L. and Scholl, P., 2023. *FAEST: algorithm specifications*. Technical report, National Institute of Standards and Technology.

Bibliography

[11] Kim, S., Ha, J., Son, M., Lee, B., Moon, D., Lee, J., Lee, S., Kwon, J., Cho, J., Yoon, H. and Lee, J., 2023. *The aimer signature scheme*. Technical report. NIST.

[12] Kales, D. and Zaverucha, G., *Improving the performance of the Picnic signature scheme. IACR TCHES, 2020 (4): 154–188, 2020* [online]

Additional Slides

Table 1: Non-linear complexity of VOLEitH signature schemes using different OWFs.

Description	FAEST			FAEST-EM		
λ	AES-128	AES-192	AES-256	AES-EM-128	AES-EM-192	AES-EM-256
No. of S-Boxes in key expansion	40	32	52	0	0	0
No. of S-Boxes in encryption	160	192	224	160	288	448
Total no. of $\mathbb{F}_{2^8}$ constraints	200	416	500	160	288	448
	FAESTER			FAESTER-EM		
λ	AES-128	AES-192	AES-256	AES-EM-128	AES-EM-192	AES-EM-256
No. of S-Boxes in key expansion	40	32	52	0	0	0
No. of S-Boxes in encryption	160	192	224	160	288	448
Total no. of $\mathbb{F}_{2^8}$ constraints	200	416	500	160	288	488
	MandaRain-3			MandaRain-4		
λ	Rain-3-128	Rain-3-192	Rain-3-256	Rain-4-128	Rain-4-192	Rain-4-256
No. of S-Boxes in encryption	3	3	3	4	4	4
Total no. of $\mathbb{F}_{2^\lambda}$ constraints	3	3	3	4	4	4
	KuMQuat-2^1			KuMQuat-2^8		
λ	MQ-$\mathbb{F}_{2^1}$-L1	MQ-$\mathbb{F}_{2^1}$-L3	MQ-$\mathbb{F}_{2^1}$-L5	MQ-$\mathbb{F}_{2^8}$-L1	MQ-$\mathbb{F}_{2^8}$-L3	MQ-$\mathbb{F}_{2^8}$-L5
Total no. of $\mathbb{F}_{2^n}$ constraints	152	224	320	48	72	96

Table 2: VOLEitH signature schemes and their parameters. We denote the signature schemes as SCHEME-$\lambda_{\mathsf{s/f}}$. l is the number of VOLE correlations required for the NIZK proof. w and $\mathsf{T_{open}}$ are the values for the optimized BAVC as described in Section 3.1. τ is the number of VOLE repetitions determining the choice between s (slow) and f (fast) versions. k_0 and k_1 are bit lengths of small VOLEs. B is the padding parameter affecting the security of the VOLE check. Secret key (sk), public key (pk) and signature sizes are in bytes.

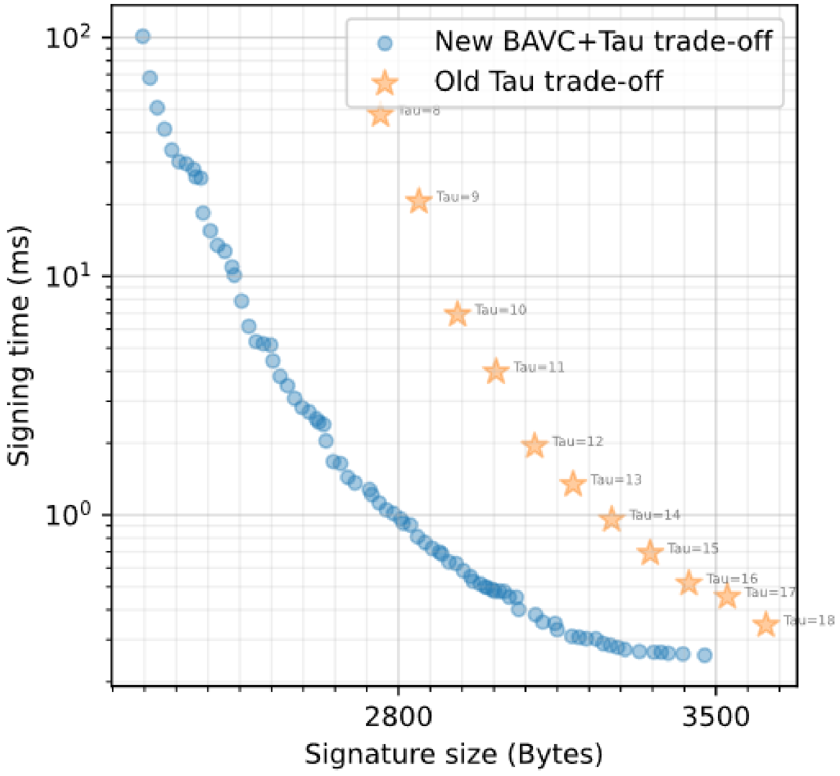
Signature Scheme	OWF $E_{sk}(x)$	l	w	$\mathsf{T_{open}}$	τ	τ_0	τ_1	k_0	k_1	sk size	pk size	sig. size
FAEST-128$_{\mathsf{s}}$	AES128$_{sk}(x)$	1600	–	–	11	7	4	12	11	16	32	5006
FAEST-128$_{\mathsf{f}}$	AES128$_{sk}(x)$	1600	–	–	16	0	16	8	8	16	32	6336
FAEST-EM-128$_{\mathsf{s}}$	AES128$_x(sk) \oplus sk$	1280	–	–	11	7	4	12	11	16	32	4566
FAEST-EM-128$_{\mathsf{f}}$	AES128$_x(sk) \oplus sk$	1280	–	–	16	0	16	8	8	16	32	5696
FAEST-d7-128$_s$	AES128$_{sk}(x)$	800	–	–	11	7	4	12	11	16	32	4790
FAEST-d7-128$_f$	AES128$_{sk}(x)$	800	–	–	16	0	16	8	8	16	32	6020
FAESTER-128$_{\mathsf{s}}$	AES128$_{sk}(x)$	1600	7	102	11	0	11	11	11	16	32	4594
FAESTER-128$_{\mathsf{f}}$	AES128$_{sk}(x)$	1600	8	110	16	8	8	8	7	16	32	6052
FAESTER-EM-128$_{\mathsf{s}}$	AES128$_x(sk) \oplus sk$	1280	7	103	11	0	11	11	11	16	32	4170
FAESTER-EM-128$_{\mathsf{f}}$	AES128$_x(sk) \oplus sk$	1280	8	112	16	8	8	8	7	16	32	5444
FAESTER-d7-128$_s$	AES128$_{sk}(x)$	800	5	102	11	0	11	11	11	16	32	4374
FAESTER-d7-128$_f$	AES128$_{sk}(x)$	800	6	110	16	8	8	8	7	16	32	5732
MandaRain-3-128$_{\mathsf{s}}$	Rain-3-128$_{\mathsf{sk}}(x)$	384	7	100	11	7	4	12	11	16	32	2890
MandaRain-3-128$_{\mathsf{f}}$	Rain-3-128$_{\mathsf{sk}}(x)$	384	8	108	16	0	16	8	8	16	32	3588
MandaRain-4-128$_{\mathsf{s}}$	Rain-4-128$_{\mathsf{sk}}(x)$	512	7	101	11	7	4	12	11	16	32	3082
MandaRain-4-128$_{\mathsf{f}}$	Rain-4-128$_{\mathsf{sk}}(x)$	512	8	110	16	0	16	8	8	16	32	3876
KuMQuat-2^1-L1$_{\mathsf{s}}$	MQ-2^1-L1$_{\mathsf{sk}}(x)$	152	7	99	11	7	4	12	11	19	35	2555
KuMQuat-2^1-L1$_{\mathsf{f}}$	MQ-2^1-L1$_{\mathsf{sk}}(x)$	152	4	102	16	0	16	8	8	19	35	3028
KuMQuat-2^8-L1$_{\mathsf{s}}$	MQ-2^8-L1$_{\mathsf{sk}}(x)$	384	7	100	11	7	4	12	11	48	64	2890
KuMQuat-2^8-L1$_{\mathsf{f}}$	MQ-2^8-L1$_{\mathsf{sk}}(x)$	384	4	108	16	0	16	8	8	48	64	3588

Table 3: Rain Parameters

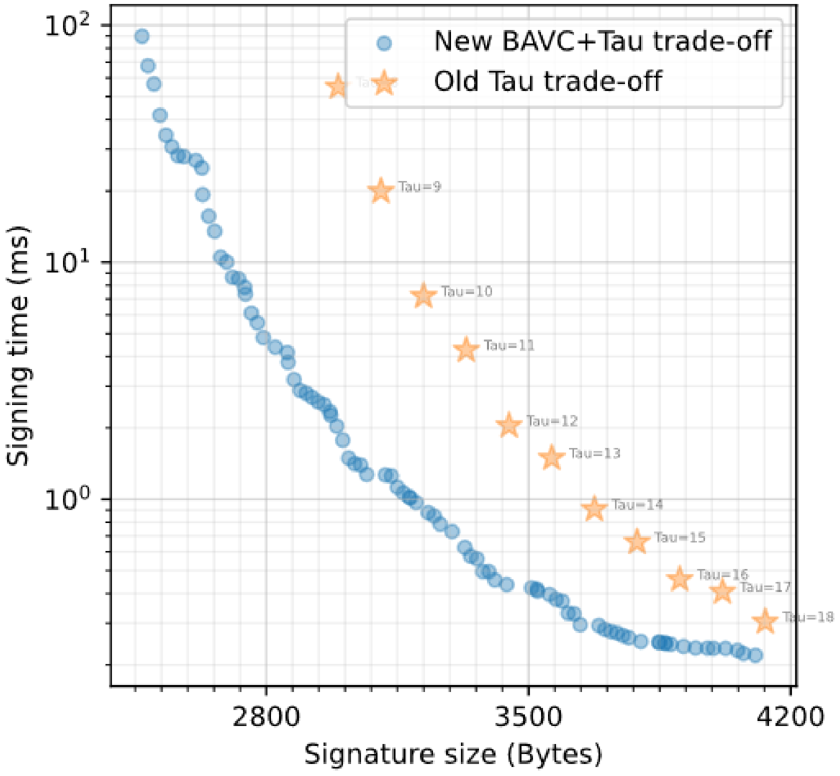
Instance	Seclvl	State	Rounds
Rain-3-128	L1	$\mathbb{F}_2^{128}$	3
Rain-3-192	L3	$\mathbb{F}_2^{192}$	3
Rain-3-256	L5	$\mathbb{F}_2^{256}$	3
Rain-4-128	L1	$\mathbb{F}_2^{128}$	4
Rain-4-192	L3	$\mathbb{F}_2^{192}$	4
Rain-4-256	L5	$\mathbb{F}_2^{256}$	4

Table 4: MQ Parameters

Instance	Seclvl	Field	$m = n$
MQ-2^1-L1	L1	$\mathbb{F}_{2^1}$	152
MQ-2^8-L1	L1	$\mathbb{F}_{2^8}$	48
MQ-2^1-L3	L3	$\mathbb{F}_{2^1}$	224
MQ-2^8-L3	L3	$\mathbb{F}_{2^8}$	72
MQ-2^1-L5	L5	$\mathbb{F}_{2^1}$	320
MQ-2^8-L5	L5	$\mathbb{F}_{2^8}$	96



(a) KuMQuat-2^1-L1.



(b) KuMQuat-2^8-L1.

Figure 7: KuMQuat τ-signature size and runtime trade-off.

Table 5: Signing Time (ms), Verification Time (ms), and Signature Size (bytes) of different VOLEitH-based signature schemes (optimized implementations). Slow and fast versions are denoted with s and f respectively.

Scheme	Runtime in ms			Size in bytes		
	Keygen	Sign	Verify	sk	pk	Signature
FAEST-128_s	0.0006	4.381	4.102	16	32	5006
FAEST-128_f	0.0005	0.404	0.395	16	32	6336
FAEST-EM-128_s	0.0005	4.151	4.415	16	32	4566
FAEST-EM-128_f	0.0005	0.446	0.474	16	32	5696
FAESTER-128_s	0.0006	3.282	4.467	16	32	4594
FAESTER-128_f	0.0005	0.433	0.610	16	32	6052
FAESTER-EM-128_s	0.0005	3.005	4.386	16	32	4170
FAESTER-EM-128_f	0.0005	0.422	0.609	16	32	5444
MandaRain-3-128_s	0.0018	2.800	5.895	16	32	2890
MandaRain-3-128_f	0.0018	0.346	0.807	16	32	3588
MandaRain-4-128_s	0.0026	2.876	6.298	16	32	3052
MandaRain-4-128_f	0.0026	0.371	0.817	16	32	3876
KuMQuat-2^1-$L1_s$	0.173	4.305	4.107	19	35	2555
KuMQuat-2^1-$L1_f$	0.172	0.539	0.736	19	35	3028
KuMQuat-2^8-$L1_s$	0.174	3.599	4.053	48	64	2890
KuMQuat-2^8-$L1_f$	0.172	0.400	0.623	48	64	3588