OPRFs from Isogenies

Design and Analysis

Lena Heimberger[1], Tobias Hennerbichler[1], Fredrik Meisingseth[1], Sebastian Ramacher[2], Christian Rechberger[1]

[1]Graz University of Technology [2]Austrian Institute of Technology

3rd of July, 2024

CSIDH + Naor-Reingold (O)PRF

CSIDH [8] in one slide

- node: curve with Montgomery coefficient
- edge: ℓ-isogeny
- random walk on commutative graph
- CSIDH 512 key: $k_i \in \{-5, 5\}$

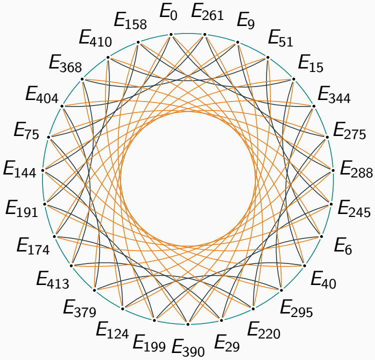


Figure 1: CSIDH graph over $\mathbb{F}_{419}$. Graphic by Chloe Martindale.

CSI-FiSh [6]

- Use Fiat-Shamir Transformation to get signature scheme:
  - add/subtract private keys
  - compute *lattice reduction*
    modulo class group number
- key with *small* key coefficients
- relational lattice available up to 1024 bits

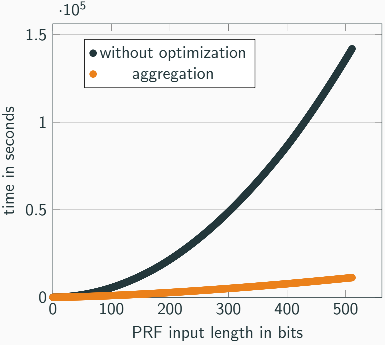
Naor-Reingold Construction + CSIDH

$$\mathcal{F}_{NR}\Big((k_0, \cdots, k_n), (x_1, \cdots, x_n)\Big) = k_0 \cdot k_1^{x_1} \cdot k_2^{x_2} \cdots k_n^{x_n}$$

Naor-Reingold Construction + CSIDH

$$F_{NR-CSIDH}\left((\mathbf{k_0}, \mathbf{k_1}, \ldots, \mathbf{k_n}, E_0), (x_1, \ldots, x_n)\right) :=$$

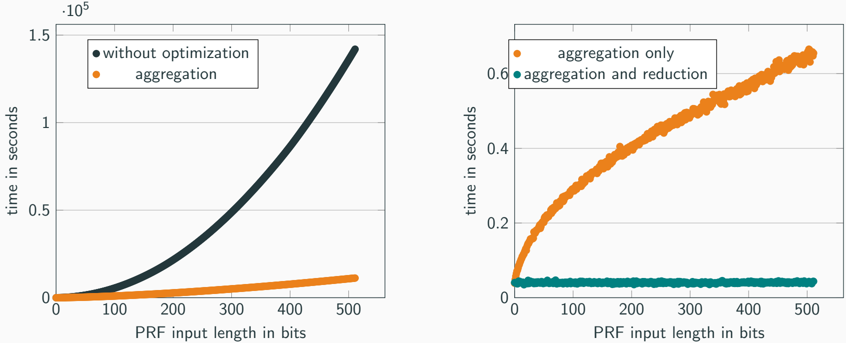
$$(\mathbf{k_0} + \sum_{i=1}^{n} \mathbf{k_i} x_i) * E_0$$



Naor-Reingold Construction + CSIDH

$$F_{NR-CSIFiSh-OPT}\left((\mathbf{k_0}, \mathbf{k_1}, \ldots, \mathbf{k_n}, E_0), (x_1, \ldots, x_n)\right) :=$$

$$\text{reduce\_mod}\left((\mathbf{k_0} + \sum_{i=1}^{n} \mathbf{k_i} x_i), cn\right) * E_0$$



Result: near constant runtime at 43 ms for PRF computation

Two methods for Oblivious Evaluation

Oblivious Pseudorandom Functions

server

key $k \in \mathcal{K}$

client

input $x \in \mathcal{X}$

Oblivious Pseudorandom Functions

server

$$\mathrm{F}(\mathrm{k}, \mathrm{x})$$

client

key $k \in \mathcal{K}$

input $x \in \mathcal{X}$

Naor-Reingold OPRF

Server		**Client**
keys $K = \mathbf{k_0}, \dots, \mathbf{k_n}$		input $X = [x_1, \dots, x_n]$
$\mathbf{r} \leftarrow [0]$		$\mathbf{k} \leftarrow [0]$
for $i \in [1, n]$:		**for** $i \in [1, N]$:
$\quad \mathbf{r_i} \xleftarrow{\$} \text{keygen}()$		
$\quad \mathbf{r} \leftarrow \mathbf{r} + \mathbf{r_i}$	$\mathbf{r_i}$, $\mathbf{k_i} + \mathbf{r_i}$ $\rightarrow$ $\boxed{\binom{2}{1}\text{-OT}}$ $\leftarrow$ x_i; $\rightarrow$ $x_i\mathbf{k_i} + \mathbf{r_i}$	$\quad \mathbf{k} \leftarrow \mathbf{k} + x_i\mathbf{k_i} + \mathbf{r_i}$
$E_{fin} \leftarrow (\mathbf{k_0} - \mathbf{r}) * E_0$	$\xrightarrow{E_{fin}}$	$E \leftarrow \mathbf{k} * E_{fin}$
		return E

OPUS: Removing the OT

Server		**Client**
$\{\mathbf{k_0}, \mathbf{k_1}, \cdots, \mathbf{k_n}\} \xleftarrow{\$} \text{keygen}()$		input $X \leftarrow \{x_1, \cdots, x_n\}$,
$\mathbf{r_s} \leftarrow [0]$		$\mathbf{r_c} \leftarrow [0], E_{client} \leftarrow E_0$
foreach $\mathrm{i} \in \{1, \ldots, n\}$:		**foreach** $\mathrm{i} \in \{1, \ldots, n\}$:
		$\mathbf{r_{c,i}} \xleftarrow{\$} \text{keygen}()$
$\mathbf{r_{s,i}} \xleftarrow{\$} \text{keygen}()$	$\xleftarrow{E_{blinded}}$	$E_{blinded} \leftarrow \mathbf{r_{c,i}} * E_{client}$
$E_{s,i,0} \leftarrow \mathbf{r_{s,i}} * E_{blinded}$		
$E_{s,i,1} \leftarrow \mathbf{k_i} * E_{s,i,0}$		
$\mathbf{r_s} \leftarrow \mathbf{r_s} - \mathbf{r_{s,i}}$	$\xrightarrow{E_{s,i,0},\ E_{s,i,1}}$	$E_{client} \leftarrow E_{s,i,x_i}$
		$\mathbf{r_c} \leftarrow \mathbf{r_c} - \mathbf{r_{c,i}}$

. . . . Finalize and Unblind . . . .

	$\xleftarrow{\mathbf{r_{c,0}} * E_{client}}$	$\mathbf{r_{c,0}} \xleftarrow{\$} \text{keygen}()$
$E_s \leftarrow (\mathbf{k_0} + \mathbf{r_s}) * (\mathbf{r_{c,0}} * E_{client})$	$\xrightarrow{E_s}$	$E_{client} \leftarrow (\mathbf{r_c} - \mathbf{r_{c,0}}) * E_s$
		return E_{client}

Computation and Communication Cost

protocol	rounds	comm. cost	isog. comp.	model (C-S)
NR-OT	2	$2\sigma \cdot \gamma + 2\gamma^2 + \sigma$	$5\gamma + 2$	◐-◐
NR-OT	4	$5\sigma \cdot \gamma + 5\gamma^2 + \sigma$	$11\gamma + 2$	● -◐
OPUS	$2\gamma + 2$	$3\sigma \cdot \gamma + 2\sigma$	$3\gamma + 3$	◐-◐

γ input bits, ρ CSIDH modulus

And in Comparison?

The Table

work	assumption	rounds	comm. cost	model (C-S)	no preproc.	no trusted setup	verifiable	full impl. available
[13]	3-Hash SDHI	2	766 bits	●-●	✓	✓	✓	✓
[2]	R(LWE)+SIS	2	2MB	◐-◐	✓	✓	✗	✓
[2]	R(LWE)+SIS	2	$>$ 128 GB	●-●	✓	✓	✓	✗
[1]	mod(2,3)+lattices	2	2.5 MB+10 KB	●-◐	✓	✓	✗	✗
[1]	mod(2,3)+lattices	2	2.5 MB+160 KB	●-◐	✓	✓	✓	✗
[12]	Legendre PRF	3	$\gamma \cdot 13$ kB	◐-◐	✗	✓	✓	✗
[5]	Legendre PRF	9	911 KB	●-●	✗	✓	✗	
[11]	Legendre PRF	2	?	◐-◐	✗	✓	✓	✓
[10]	AES+GC	2	6.79MB	◐-◐	✓	✓	✗	✓
[9]	mod(2,3)	2	1836 bits	◐-◐	✗	✗	✗	✗
[3]	Isogenies $\mathbb{F}_{p^2}$	2	3.0 MB	●-●	✓	✗	✗	✗
[3]	Isogenies $\mathbb{F}_{p^2}$	2	8.7 MB	●-●	✓	✗	✓	✗
[4]	higher-dimensional Isogenies $\mathbb{F}_{p^2}$	2	28.9 kB	●-●	✓	✓	✓	✓
[7]	Isogenies $\mathbb{F}_p$ + lattices	2	20.54 kB	◐-◐	✓	✗	✗	✗
[7]	Isogenies $\mathbb{F}_p$ + lattices	2	20.54 kB	◐-◐	✓	✗	✗	✗
[7]	Isogenies $\mathbb{F}_p$ + lattices	4	34.88 kB	●-◐	✓	✗	✗	✗
this work	Isogenies $\mathbb{F}_p$ + lattices + HE OT	2	640 kB	◐-◐	✓	✓	✗	✓
this work	CSIDH	258	24.7 kB	◐-◐	✓	✓	✗	✓

hosted at `heimberger.xyz/oprfs`

OPRFs from Isogenies

Design and Analysis

Lena Heimberger[1], Tobias Hennerbichler[1], Fredrik Meisingseth[1], Sebastian Ramacher[2], Christian Rechberger[1]

[1]Graz University of Technology [2]Austrian Institute of Technology

3rd of July, 2024

References i

M. R. Albrecht, A. Davidson, A. Deo, and D. Gardham.
Crypto dark matter on the torus: Oblivious PRFs from shallow PRFs and FHE.
Cryptology ePrint Archive, Report 2023/232, 2023.
`https://eprint.iacr.org/2023/232`.

M. R. Albrecht, A. Davidson, A. Deo, and N. P. Smart.
Round-optimal verifiable oblivious pseudorandom functions from ideal lattices.
pages 261–289, 2021.

A. Basso.
A post-quantum round-optimal oblivious PRF from isogenies.
SAC Selected Areas in Cryptography, 2023.

References ii

A. Basso.
POKE: A framework for efficient PKEs, split KEMs, and OPRFs from higher-dimensional isogenies.
Cryptology ePrint Archive, Paper 2024/624, 2024.
`https://eprint.iacr.org/2024/624`.

W. Beullens, L. Dodgson, S. Faller, and J. Hesse.
The 2Hash OPRF framework and efficient post-quantum instantiations.
Cryptology ePrint Archive, Paper 2024/450, 2024.
`https://eprint.iacr.org/2024/450`.

W. Beullens, T. Kleinjung, and F. Vercauteren.
CSI-FiSh: Efficient isogeny based signatures through class group computations.
pages 227–247, 2019.

References iii

D. Boneh, D. Kogan, and K. Woo.
Oblivious pseudorandom functions from isogenies.
pages 520–550, 2020.

W. Castryck, T. Lange, C. Martindale, L. Panny, and J. Renes.
CSIDH: An efficient post-quantum commutative group action.
pages 395–427, 2018.

I. Dinur, S. Goldfeder, T. Halevi, Y. Ishai, M. Kelkar, V. Sharma, and G. Zaverucha.
MPC-friendly symmetric cryptography from alternating moduli: Candidates, protocols, and applications.
pages 517–547, 2021.

S. H. Faller, A. Ottenhues, and J. Ottenhues.
Composable oblivious pseudo-random functions via garbled circuits.
pages 249–270, 2021.

References iv

N. Kaluderovic, N. Cheng, and K. Mitrokotsa.
A post-quantum distributed OPRF from the legendre PRF.
Cryptology ePrint Archive, Paper 2024/544, 2024.
`https://eprint.iacr.org/2024/544`.

I. A. Seres, M. Horváth, and P. Burcs.
The legendre pseudorandom function as a multivariate quadratic cryptosystem: security and applications.
In *AAECC*. Springer, 01 2023.

N. Tyagi, S. Celi, T. Ristenpart, N. Sullivan, S. Tessaro, and C. A. Wood.
A fast and simple partially oblivious PRF, with applications.
pages 674–705, 2022.

Concrete Communication Cost

parameters		protocol	rounds	comm. cost	isog. comp.	model (C-S)
$\gamma = 128$	$\sigma = 512$	NR-OT	2	21 kB	624	◐-◐
		NR-OT	4	51 kB	1410	● -◐
		OPUS	258	25 kB	386	◐-◐
$\gamma = 256$	$\sigma = 2048$	NR-OT	2	148 kB	1282	◐-◐
		NR-OT	4	369 kB	2818	● -◐
		OPUS	514	197 kB	770	◐-◐
$\gamma = 256$	$\sigma = 5280$	NR-OT	2	355 kB	1282	◐-◐
		NR-OT	4	886 kB	2818	● -◐
		OPUS	514	508 kB	770	◐-◐

Isogenies and Private Set Intersection

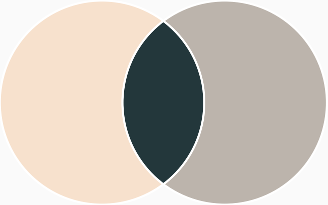
Alice $(x_1, \ldots, x_m)$

Bob $(y_1, \ldots, y_n), k$

$\{F_k(x_i)\}_{i\in[m]}$

$$\xleftrightarrow{\quad OPRF \quad}$$

$$\xleftarrow{\quad \{z_j = F_k(y_i)\}_{j\in[n]} \quad}$$



- If $x_i = y_j$ then $F_k(x_i) = z_j$
- Otherwise $F_k(y_j)$ is pseudorandom.

NR-OT Performance

Input-length	Keygen PRF	Comp. PRF	Client	Server	OT keygen
128	204ms	43ms	90ms 128 kiB	91ms 256 kiB	429ms 256 kiB
256	378ms	43ms	97ms 256 kiB	97ms 512 kiB	428ms 256 kiB
512	763ms	45ms	101ms 384 kiB	101ms 768 kiB	427ms 256 kiB

OPUS Performance

Bit-length	Keygen PRF	Comp. PRF	Client	Server	Overall
128	0.11ms	168ms	3.00s 8.06 kiB	5.73s 16.06 kiB	8.73s 24.13 kiB
256	0.26ms	234ms	5.83s 16.1 kiB	11.30s 32.1 kiB	17.13s 48.13 kiB
512	0.51ms	326ms	11.47s 32.06 kiB	22.42s 64.06 kiB	33.89s 96.13 kiB

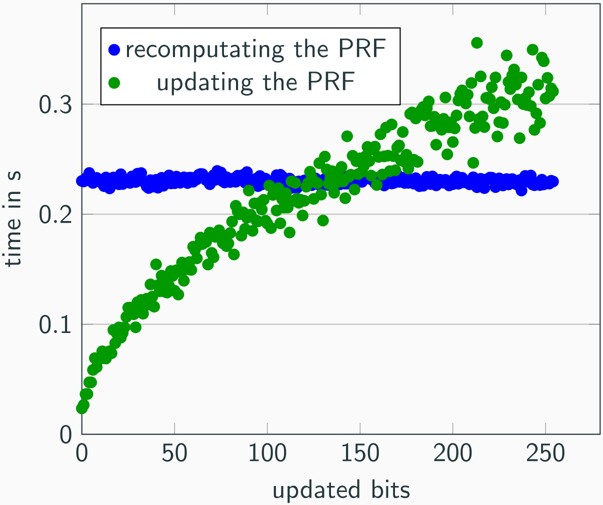
Computation Overhead: libOPAQUE vs. OPAQUE with Isogenis

Function	libopaque	PQ		PQ / libopaque	
		OPUS	NR-OT	OPUS	NR-OT
Reg. Client	119.37ms	39.82s	11.59s	$\times$ 333.62	$\times$ 97.10
Reg. Server	95.63ms	39.84s	11.61s	$\times$ 416.62	$\times$ 121.42
Auth. Client	96.54ms	31.21s	3.25s	$\times$ 323.27	$\times$ 33.69
Auth. Server	120.32ms	32.01s	2.74s	$\times$ 268.15	$\times$ 22.80

Communication Overhead: libOPAQUE vs. OPAQUE with Isogenies

Function	libopaque	PQ		PQ / libopaque	
		OPUS	NR-OT	OPUS	NR-OT
Reg. Client	224B	64kiB	817kiB	$\times$ 294.4	$\times$ 3733
Reg. Server	64B	48kiB	144kiB	$\times$ 770	$\times$ 2307.4
Auth. Client	160B	17kiB	769kiB	$\times$ 106.1	$\times$ 4920.2
Auth. Server	320B	65kiB	161kiB	$\times$ 208.2	$\times$ 515.7

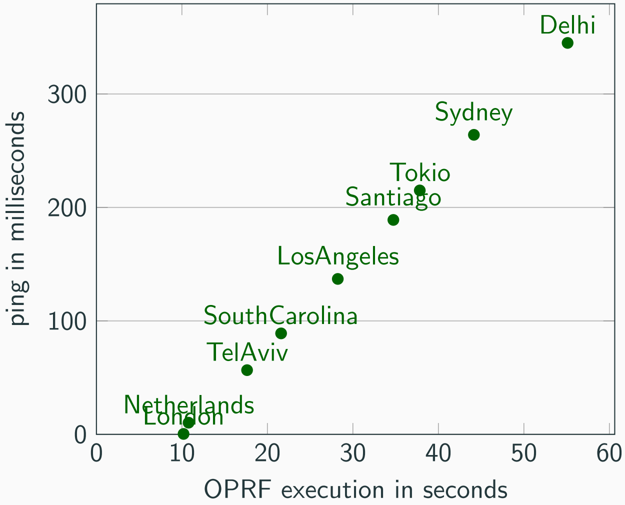
Updating the PRF values when computing many evaluations



OPUS full protocol

Server		Client
$\{k_0, k_1, \cdots, k_n\} \xleftarrow{\$} \mathrm{keygen}()$		input $X \leftarrow \{x_1, \cdots, x_n\}$
$r_s \xleftarrow{\$} \mathrm{keygen}()$		
$E_{s,0} \leftarrow (r_s) * E_0$		
	$\xrightarrow{E_{s,0}}$	$E_{client} \leftarrow E_{s,0}$
	OPRF computation	
		foreach i $\in 1, \ldots, n$:
foreach i $\in 1, \ldots, n$:	$\xleftarrow{r_i * E_{client}}$	$r_i \xleftarrow{\$} \mathrm{keygen}()$
Evaluate $k_i * (r_i * E_{c,i})$		**if** x_i:
	$\xrightarrow{E_{s,i} \leftarrow (k_i + r_i) * E_{client}}$	$E_{client} \leftarrow r_i^{-1} * E$
	Unblinding	

The impact of OPUS' rounds



	parameters		setup		online	
	$\lvert S\rvert$	$\lvert C\rvert$	$\lvert S\rvert$	$\lvert C\rvert$	$\lvert S\rvert$	$\lvert C\rvert$
NR-OT	2^0	2^0	0.26s	0.51s	0.06s	0.10s
			134 bytes	1 byte	128 kiB	0.75MiB
	2^5	2^5	1.63s	1.88s	3.11s	3.15s
			263 bytes	1 byte	4MiB	8.5 MiB
	2^{10}	2^{10}	45.04s	45.28s	99.66s	99.71s
			4.31 MiB	1 byte	128 MiB	256.6 MiB
OPUS	2^0	2^0	0.26s	0.26s	15.47s	15.91s
			133 bytes	0 bytes	17.07 kiB	9.04 kiB
	2^5	2^5	8.71s	8.71s	328.46s	329.14s
			262 bytes	0 bytes	546.25 kiB	290.26 kiB
	2^{10}	2^{10}	303.38s	303.38s	16367.12s	16367.60s
			4.31 kiB	0 bytes	34.14 MiB	18.08 MiB
ECNR	2^0	2^0	0.01s	0s	0.23s	0.05s
			133 bytes	0 bytes	12.04 kiB	16 bytes
	2^5	2^5	0.02s	0s	0.21s	0.06s
			262 bytes	0 bytes	137.05 kiB	512 bytes
	2^{10}	2^{10}	0.3s	0s	0.64s	0.57s
			4.36 kiB	0 bytes	4.04 MiB	16 kiB

Private Set Intersection with OPUS

Server		Client
$\{\mathbf{k_0}, \mathbf{k_1}, \cdots, \mathbf{k_n}\} \xleftarrow{\$} \text{keygen}()$		
l inputs $\{S_1, \cdots, S_l\}$		m inputs $\{C_1, \cdots, C_m\}$
$CF = \text{cuckoofilter}()$		
foreach i $\in \{1, \ldots, l\}$:		
CF.insert(PRF(X_l))	$\xrightarrow{\text{CF}}$	$E_{client} = []$
foreach i $\in \{1, \ldots, m\}$:		**foreach** i $\in \{1, \ldots, m\}$:
$\mathbf{r_{s,i}} \leftarrow [0]$		$\mathbf{r_{c,i}} \leftarrow [0], E_{client,i} \leftarrow E_0$
foreach j $\in \{1, \ldots, n\}$:		**foreach** j $\in \{1, \ldots, n\}$:
		$\mathbf{r_{c,i,j}} \xleftarrow{\$} \text{keygen}()$
$\mathbf{r_{s,i,j}} \xleftarrow{\$} \text{keygen}()$	$\xleftarrow{(E_{blinded}, i, j)}$	$E_{blinded} \leftarrow \mathbf{r_{c,i,j}} * E_{client}$
$E_{s,i,0} \leftarrow \mathbf{r_{s,i,j}} * E_{blinded}$		
$E_{s,i,1} \leftarrow \mathbf{k_i} * E_{s,i,0}$		
$\mathbf{r_{s,i}} \leftarrow \mathbf{r_{s,i}} - \mathbf{r_{s,i,j}}$	$\xrightarrow{(E_{s,i,0}, E_{s,i,1}, i, j)}$	$E_{client,i} \leftarrow E_{s,i,c_{i,j}}$
		$\mathbf{r_{c,i}} \leftarrow \mathbf{r_{c,i}} - \mathbf{r_{c,i,j}}$

.... Finalize ....

Server		Client
	$\xleftarrow{(\mathbf{r_{c,i,0}} * E_{client,i}, i, m)}$	$\mathbf{r_{c,i,0}} \xleftarrow{\$} \text{keygen}()$

ECNR with Table Lookups

LUT size	nr. mult.	pre-comp. in ms	Time for PRF comp.	improv. with LUT	improv. w/o LUT
2^0 -1	2^7	/	599.45s	/	/
2^2 -1	2^6	0.05ms	572.10s	4.78%	4.78%
2^4 -1	2^5	0.16ms	538.26s	11.36%	11.37%
2^8 -1	2^4	6.24ms	514.61s	16.48 %	16.49%
2^{16}-1	2^3	1.73s	482.11s	17.74 %	18.15%