SQIsignHD

Sqiing in higher dimensions

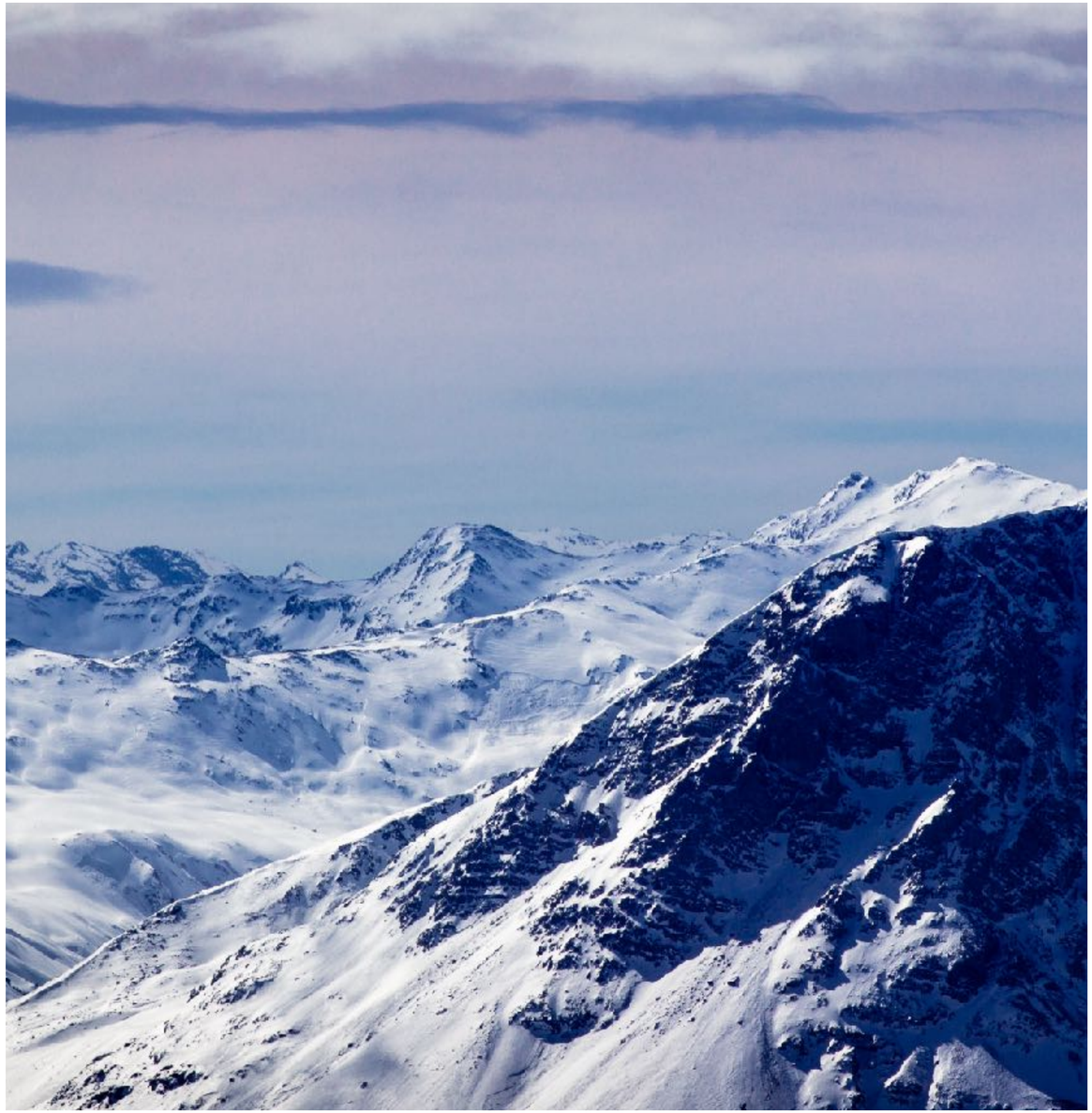
Benjamin Wesolowski, CNRS and ENS de Lyon

29 April 2024, CAIPI Symposium, Rennes, France

Based on a joint work with **Pierrick Dartois**, **Antonin Leroux**, and **Damien Robert**

SQIsign & friends

Isogeny-based signature schemes



Picture by Beppe Rijs

Isogeny-based signatures

Several isogeny-based digital signature schemes

Shared feature: **compact keys** (unless **tradeoff**)

	Based on	Iterations	Sig. size	Efficiency	Note
[Yoo+17]	SIDH assumptions	λ	$O(\lambda^2)$	8 seconds	broken
[GPS17] GPS	Endomorphism ring problem	λ	$O(\lambda^2)$	no. implem.	impractical
[DG19] SeaSign	CSIDH assumption	λ	tradeoff	7 minutes	
[BKV19] CSI-FiSh	CSIDH assumption	λ	tradeoff	400 to 2000 ms	subexp. precomp.

SQIsign

[De Feo, Kohel, Leroux, Petit, W. - Asiacrypt 2020] SQISign: compact post-quantum signatures from quaternions and isogenies

- Signature from **proof of knowledge of endomorphism ring**
- **Most compact post-quantum signature scheme**: PK + Signature combined 5× smaller than Falcon

Secret key (bytes)	Public key (bytes)	Signature (bytes)	Security
16	64	204	NIST-I

	Key gen. (MCycles)	Signing (MCycles)	Verif. (MCycles)
Original SQIsign	1242	4811	99
Optimized SQIsign	286	1354 400ms	21 6ms

Drawbacks of SQIsign

- Signing in 400ms is **too slow**
- Security proof: the ZK property is based on an ***ad hoc* assumption**
- **Bad scaling** to higher security levels (signing at NIST-V takes 40s)

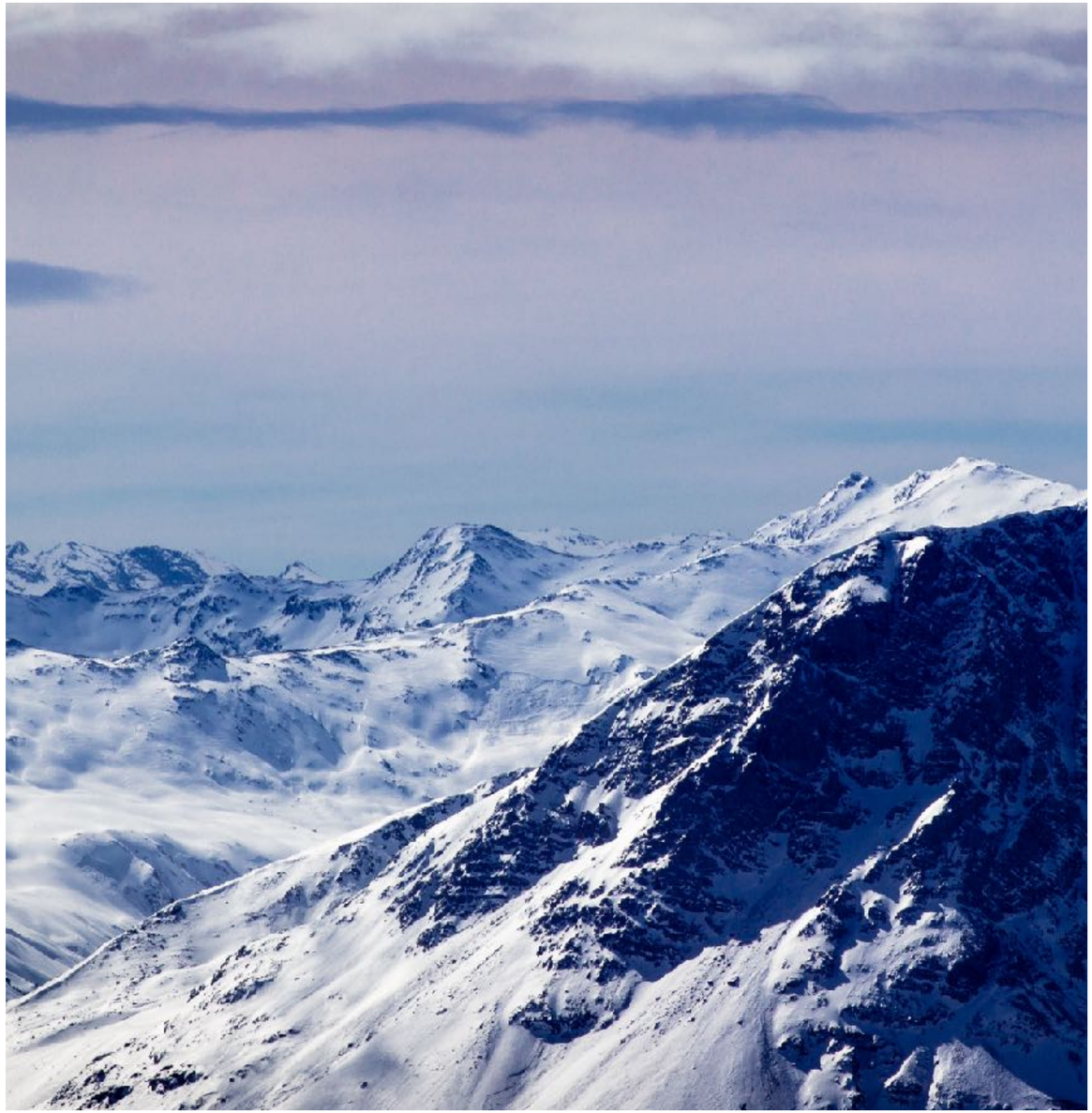
SQIsignHD solves all of these

With even more compact PK + Signature!

9x smaller than Falcon

Supersingular curves

Isogenies and endomorphisms



Picture by Beppe Rijs

Computing $\mathrm{End}(-) \Leftrightarrow$ Finding isogenies

The endomorphism ring problem

For E supersingular $\mathrm{End}(E) = \{\varphi : E \to E\}$ is a lattice of dimension 4

EndRing: Given a supersingular elliptic curve E, find a basis of $\mathrm{End}(E)$

Solution of the form $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$...

- How are α_i represented?
  - Any *efficient representation*

Efficient representation of isogenies

How to represent an isogeny?

- an **efficient representation** of φ: can evaluate $\varphi(P)$ in poly. time for any P

Examples:

- explicit polynomial fomulae — only good for small degree
- $\varphi \circ \psi$ represented by (‘compose’, φ, ψ) where φ and ψ are in efficient repr.
- $\varphi + \psi$ represented by (‘add’, φ, ψ) where φ and ψ are both $E_1 \to E_2$

Computing End(–) ⇔ Finding isogenies

End(–) is a GPS that allows to find your way in isogeny graphs:

- given End(E_1) and End(E_2), one can find ~~an isogeny $E_1 \to E_2$ in poly. time~~ a basis of Hom(E_1, E_2)

You can update the GPS coordinates as you travel through isogenies:

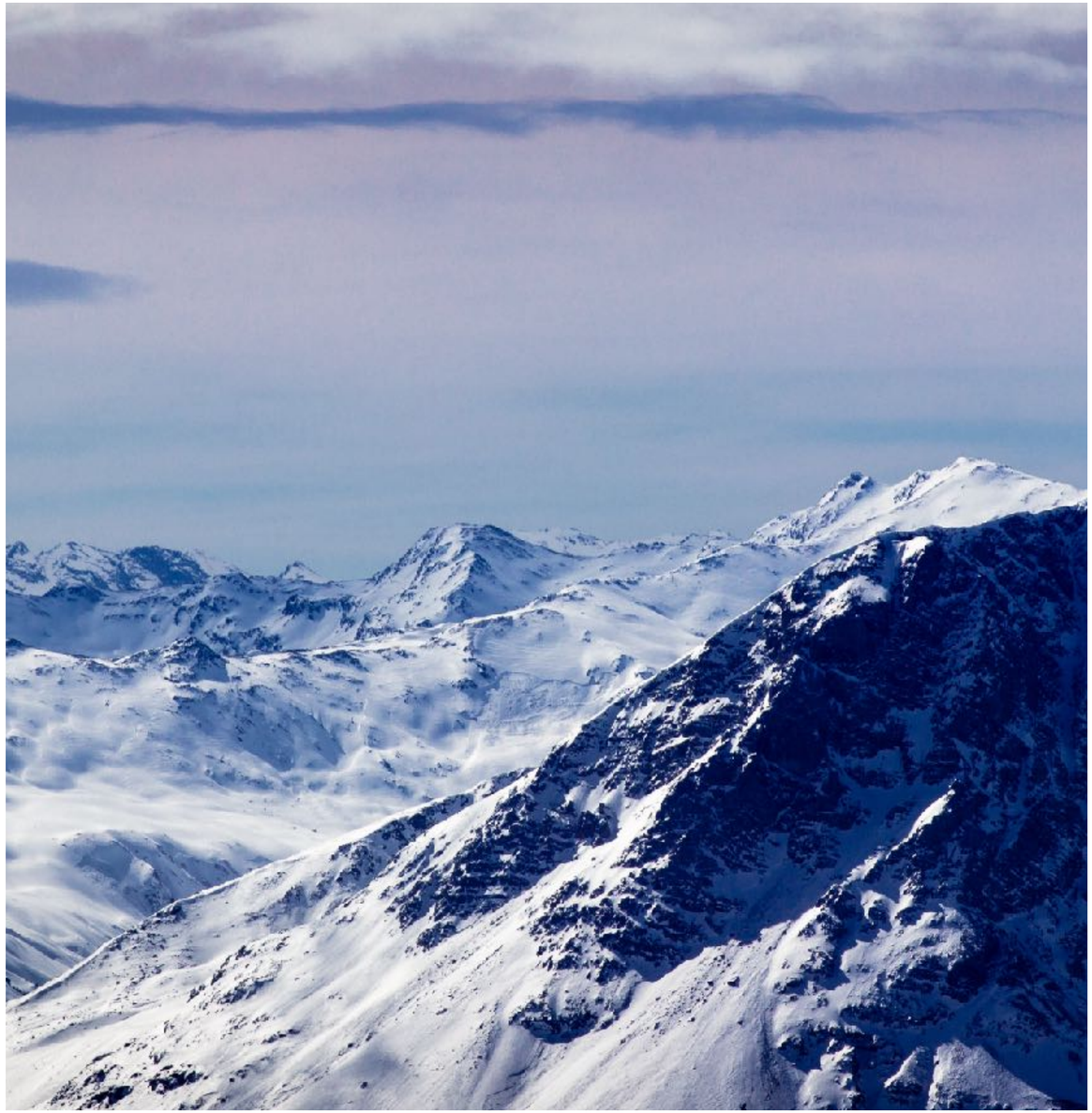
- given End(E_1), and an isogeny $E_1 \to E_2$, one can find End(E_2) in poly. time

For E_1, E_2 supersingular, Hom(E_1, E_2) is a lattice of rank 4

Computing End(–) ⇔ Computing Hom(–, –)

Key generation

Generating a curve with its endomorphism ring



Picture by Beppe Rijs

Idea of SQIsign

Basic idea of SQIsign:

- **Public key**: a supersingular curve E_{pk}
- **Secret key**: a basis of $\text{End}(E_{\text{pk}})$
- **SQIsign proof of knowledge**: a sigma protocol to prove knowledge of $\text{End}(E_{\text{pk}})$
- **SQIsign**: Fiat-Shamir transform of the proof of knowledge

Key recovery = EndRing

How to generate a random E_{pk} together with $\text{End}(E_{\text{pk}})$?

How to generate even a single supersingular curve?

A special supersingular curve

Example: $p \equiv 3 \pmod 4$, so $\mathbb{F}_{p^2} = \mathbb{F}_p(\alpha)$ where $\alpha^2 = -1$, and

Consider $E_0 : y^2 = x^3 + x$

Two non-trivial endomorphisms:

- $\pi : E_0 \to E_0 : (x, y) \mapsto (x^p, y^p)$
- $\iota : E_0 \to E_0 : (x, y) \mapsto (-x, \alpha y)$

$\pi^2 = [-p]$

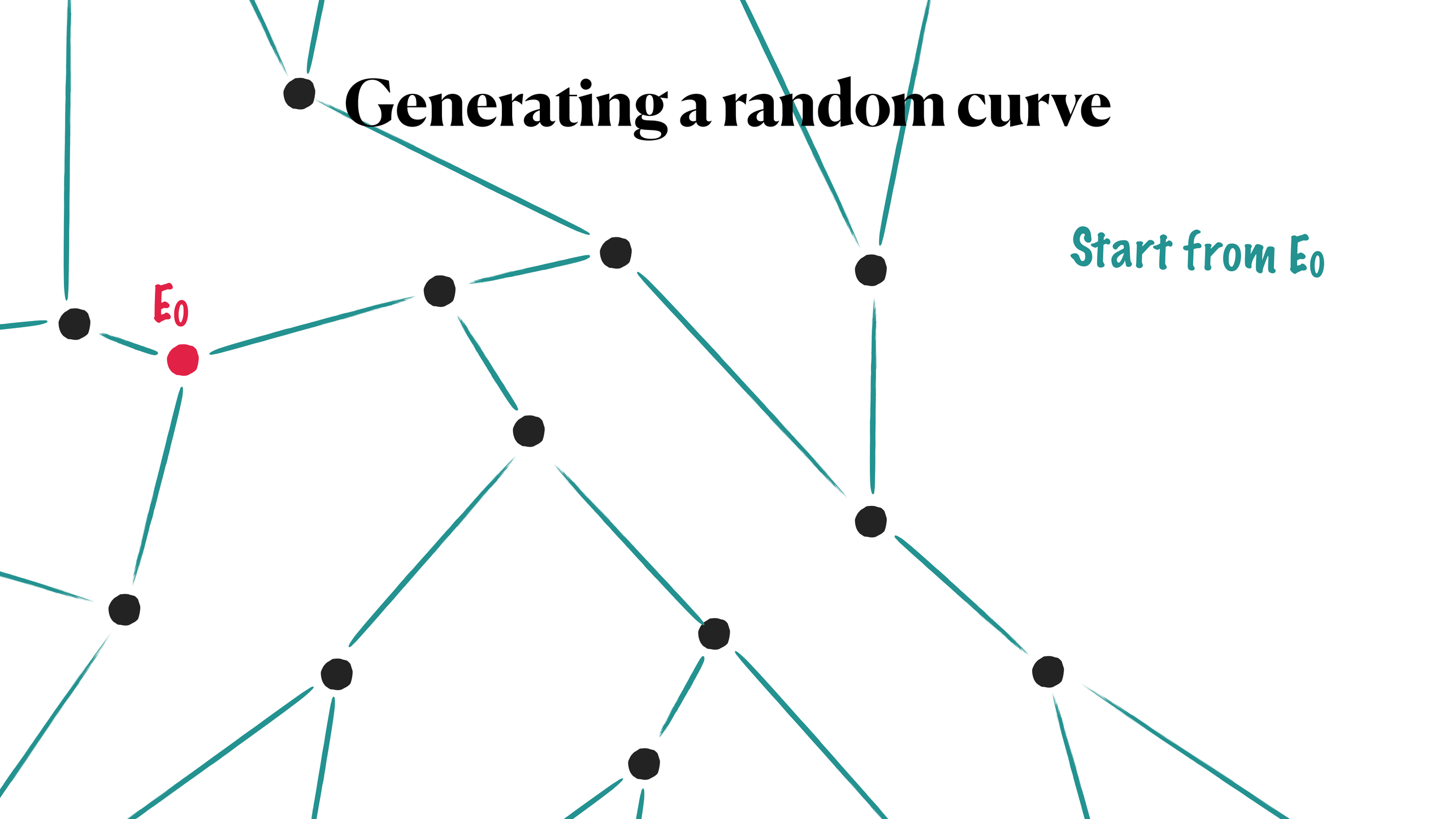
$\iota^2 = [-1]$

and $\iota\pi = -\pi\iota$

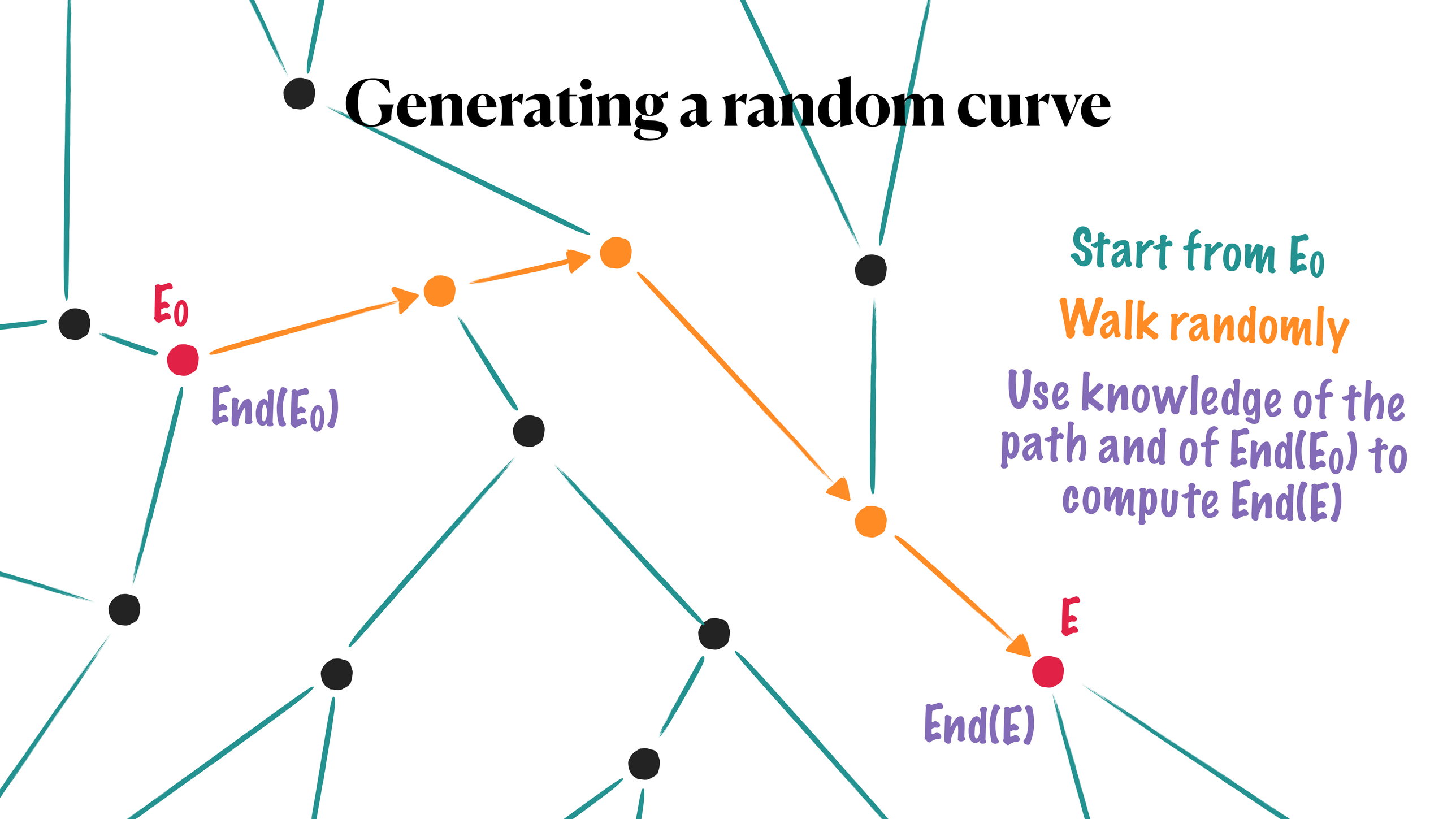
$$\textbf{End}(E_0) = \mathbb{Z} \oplus \mathbb{Z}\iota \oplus \mathbb{Z}\,\frac{\iota + \pi}{2} \oplus \mathbb{Z}\,\frac{1 + \iota\pi}{2}$$

E_0 and End(E_0) is our reference

Generating a random curve



Generating a random curve



Start from E_0

Walk randomly

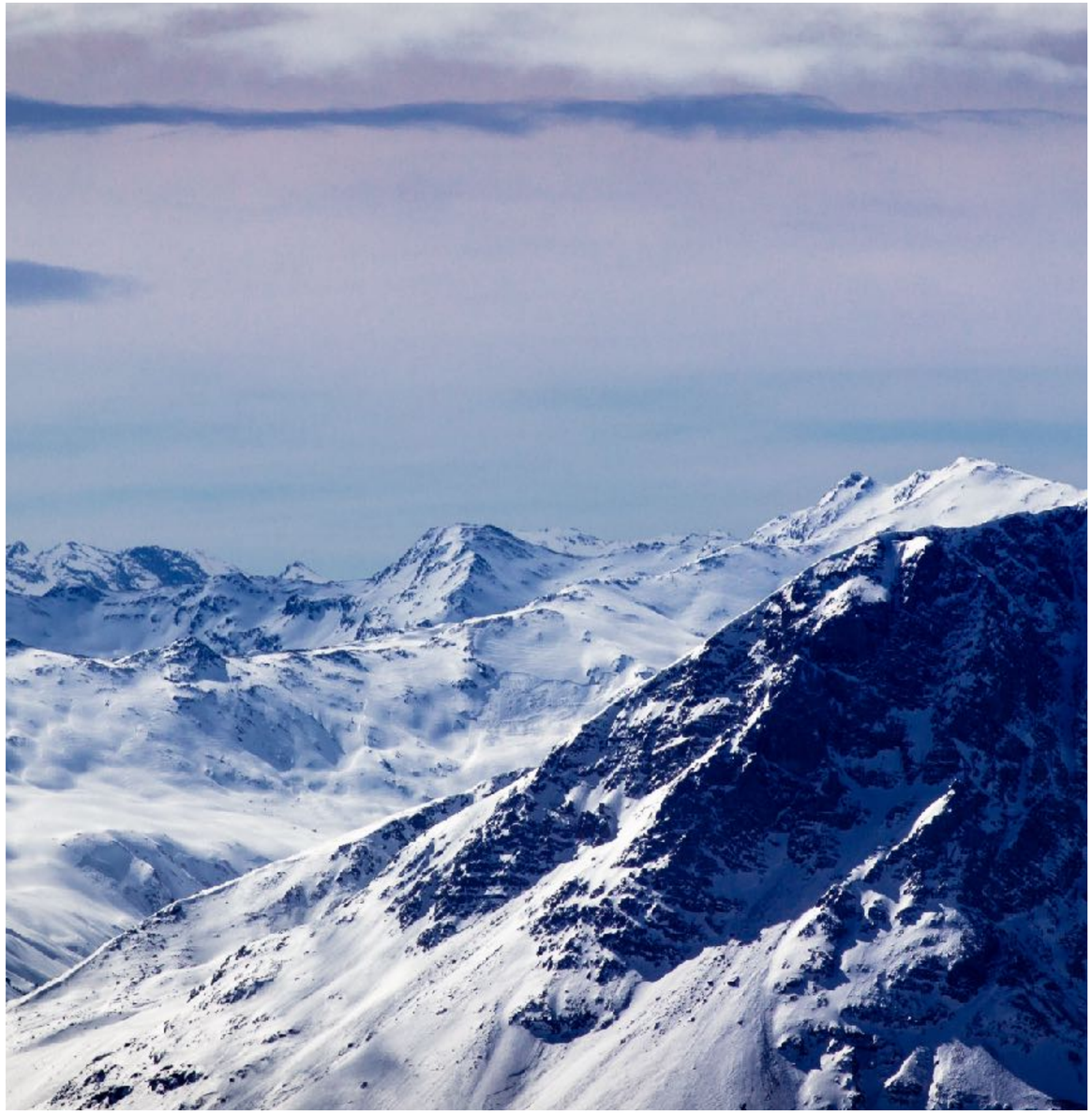
Use knowledge of the path and of $End(E_0)$ to compute $End(E)$

One can generate $(E, \mathrm{End}(E))$ with E uniform

(Trapdoor generation of uniform EndRing instances)

SQIsign

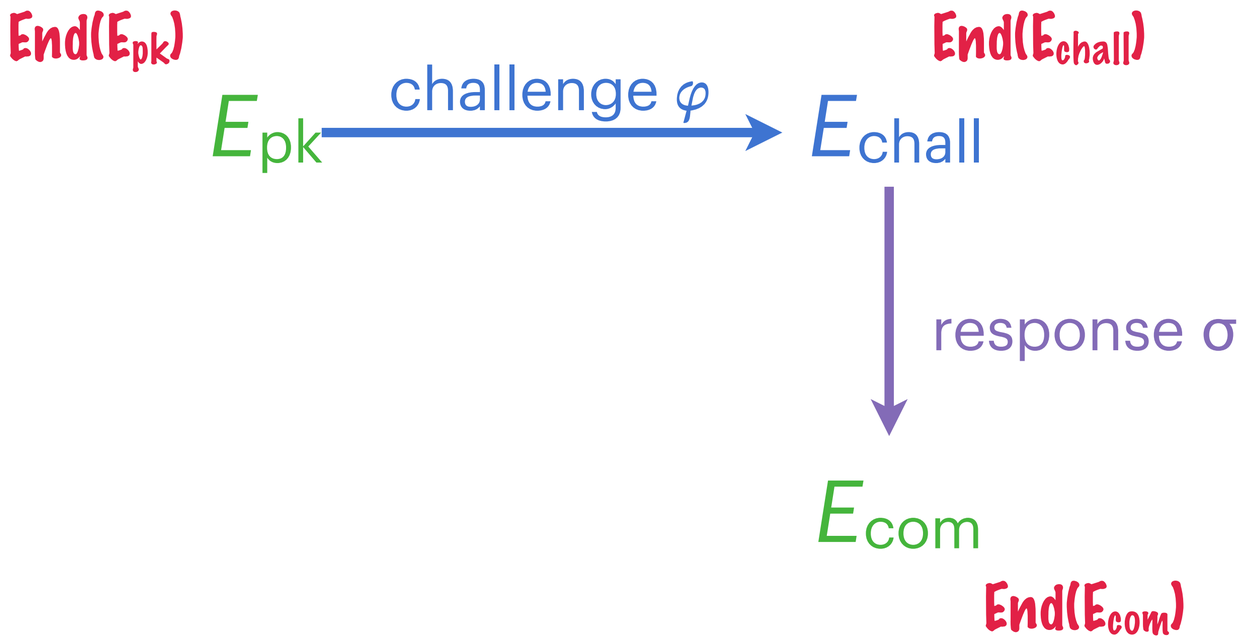
Proving knowledge of an endomorphism



Picture by Beppe Rijs

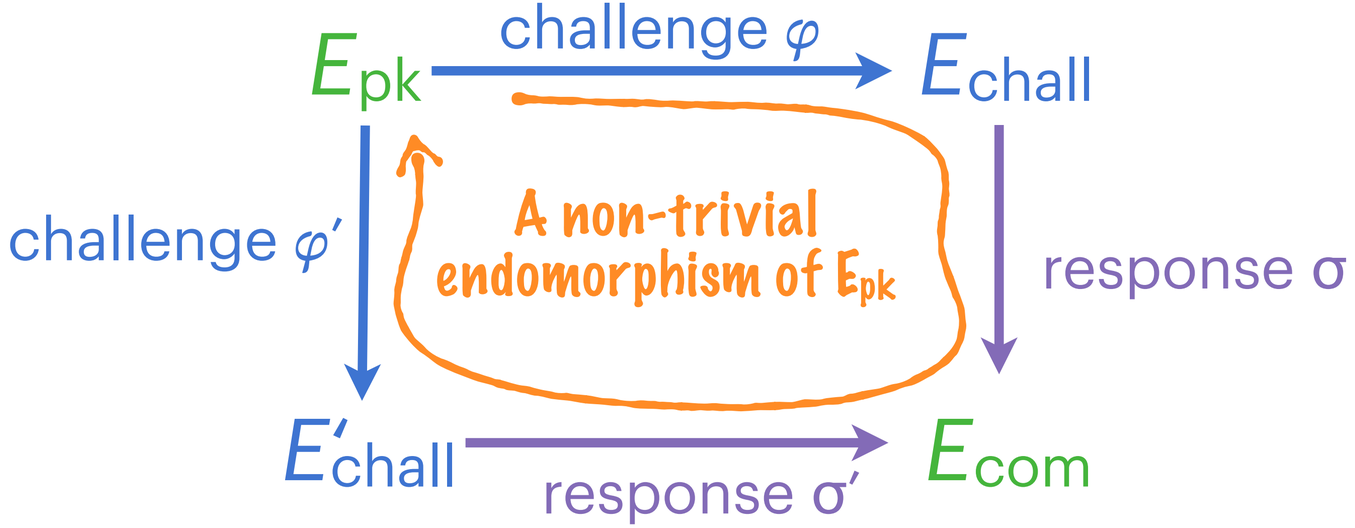
A sigma protocol following [DKLPW20]

- Generate a random pair $(E_{pk}, \mathrm{End}(E_{pk}))$
- **public key** = E_{pk}, **secret key** = $\mathrm{End}(E_{pk})$



Alice (prover)		**Bob** (verifier)
Generate random $(E_{com}, \mathrm{End}(E_{com}))$	$\xrightarrow{E_{com}}$	
	$\xleftarrow{\varphi}$	Generate random $\varphi : E_{pk} \to E_{chall}$
Compute $\sigma : E_{chall} \to E_{com}$	$\xrightarrow{\sigma}$	Check that σ is an isogeny $E_{chall} \to E_{com}$

Special soundness



OneEnd ⇔ EndRing

Can respond to 2 challenges ⇒ **Can find an endomorphism**

Computing the response isogeny

1. From End(E_{chall}) and End(E_{com}), find a basis (φ_1, φ_2, φ_3, φ_4) of Hom(E_{chall}, E_{com})
2. Return some $\sigma \in$ Hom(E_{chall}, E_{com})

Question: how to choose σ? How to represent it?

- Need σ fast to evaluate, for efficient verification
- Need σ (and its representation) not to leak the secret
  - **Warning:** typical representation of φ_i leaks End(E_{chall})

Computing the response isogeny

Original SQIsign [DKLPW20]**:**

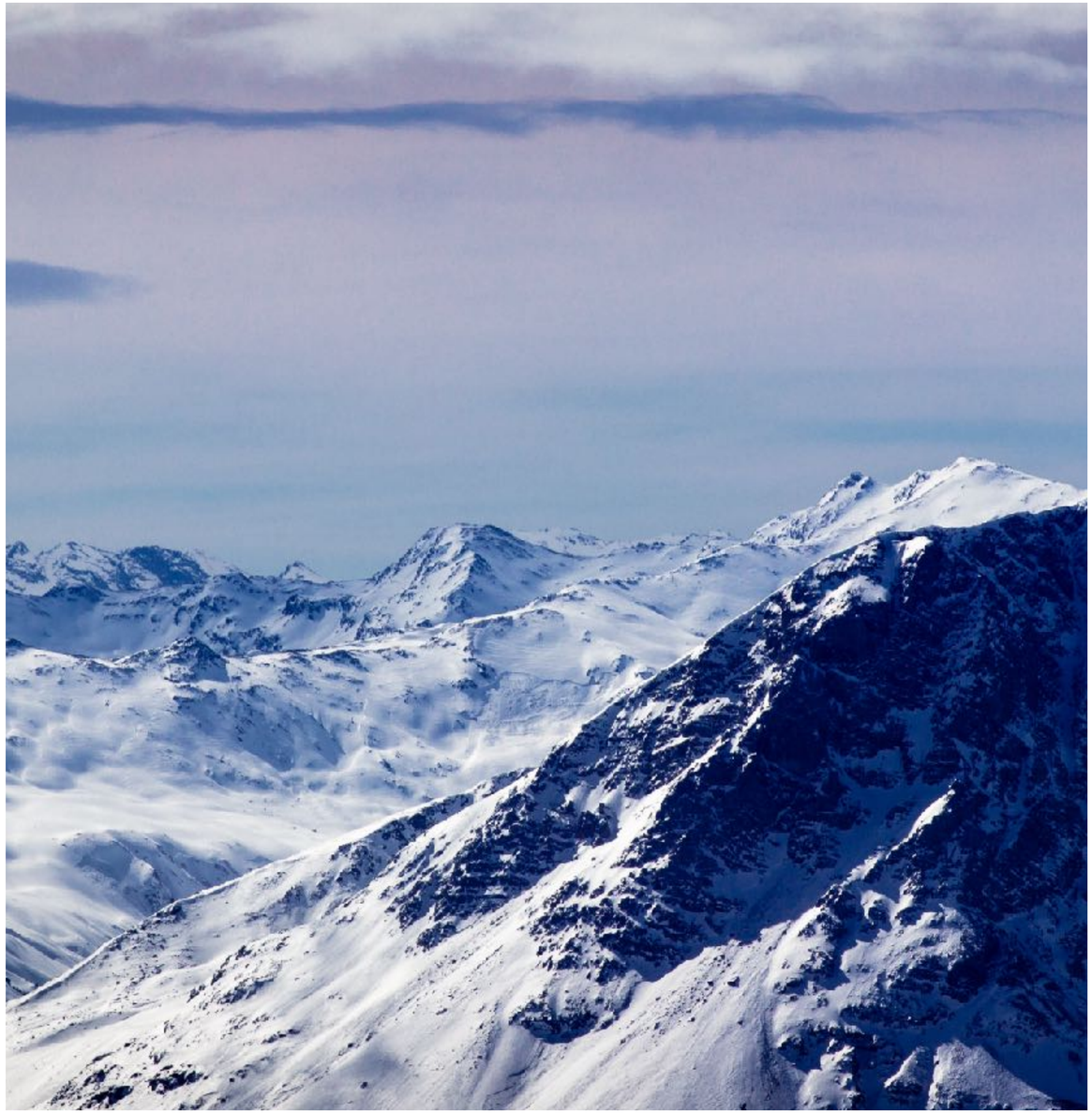
- Solve a norm equation [KLPT14] to find $\sigma \in \mathrm{Hom}(E_{chall}, E_{com})$ of degree 2^n,
- Convert $\sigma = a_1\varphi_1 + a_2\varphi_2 + a_3\varphi_3 + a_4\varphi_4$ to path of 2-isogenies

Problems:

- [KLPT14] finds **big** solution: $\deg(\sigma) = 2^n \approx p^{3.75}$
- Conversion to chain of 2-isogenies is very **costly**: signing takes billions of cycles
- Distribution of [KLPT14] output is mysterious: **not simulatable**? not zero-knowledge?

SQIsignHD

Sqiing in higher dimensions



Picture by Beppe Rijs

Computing the response isogeny

Original SQIsign [DKLPW20]**:** [KLPT14] finds big solution $\deg(\sigma) = 2^n \approx p^{3.75}$

- Smallest isogeny in $\mathrm{Hom}(E_{chall}, E_{com})$ has degree $\approx p^{0.5}$

Question: Why not output some small $\sigma \in \mathrm{Hom}(E_{chall}, E_{com})$, $\deg(\sigma) \approx p^{0.5}$?

- Representation as linear combination $a_1\varphi_1 + a_2\varphi_2 + a_3\varphi_3 + a_4\varphi_4$ is dangerous:
  - $(\varphi_1, \varphi_2, \varphi_3, \varphi_4)$ leaks $\mathrm{End}(E_{chall})$, so **leaks secret key** $\mathrm{End}(E_{pk})$

HD representation of isogenies

Attacks against SIDH [CD23, MMPPW23, Rob23]**:**

- Let $\varphi : E_1 \to E_2$ of degree d
- Let $n > (\log_2(d) + 1)/2$, and (P, Q) is a basis of $E_1[2^n]$
- Given $(d, P, Q, \varphi(P), \varphi(Q))$, one can compute $\varphi(R)$ for any $R \in E_1$ in poly. time
- **Cost:** evaluating a 2^{2n}-isogeny in dimension 2, 4 or 8

Corollary: $(d, P, Q, \varphi(P), \varphi(Q))$ is an efficient repr. of φ, "**HD representation**", or "**interpolation representation**"

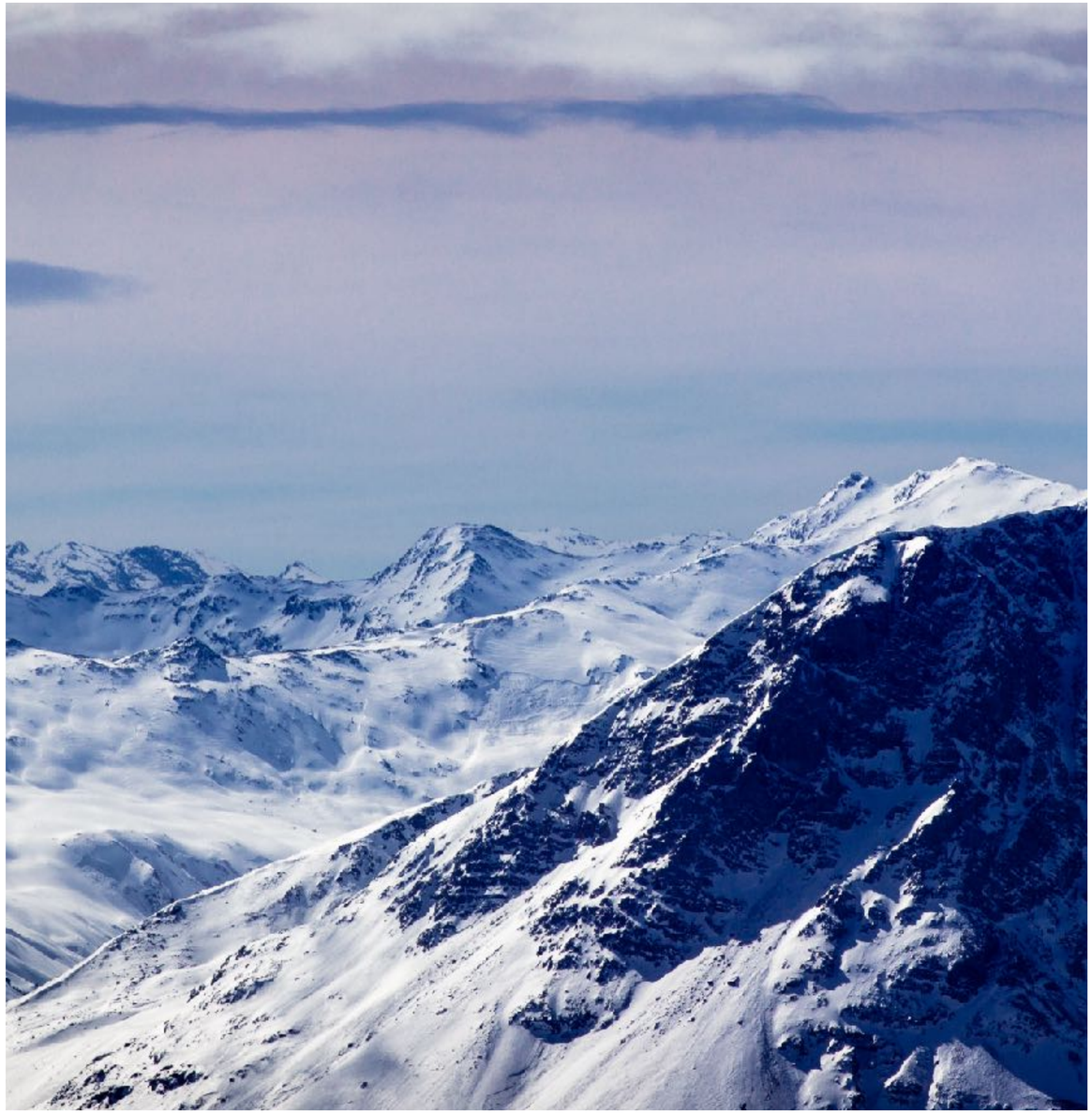
HD response isogeny

SQIsignHD [DLRW24]**:** constructive use of SIDH attacks

1. Pick random, **small** $\sigma \in \mathrm{Hom}(E_{chall}, E_{com})$ (say, $\deg(\sigma) \approx p^{0.5}$)
2. Generate basis (P, Q) of $E_{chall}[2^e]$, for $2^{2e} > 4 \cdot \deg(\sigma)$
3. Evaluate $P' = \sigma(P)$ and $Q' = \sigma(Q)$
4. $(\deg(\sigma), P, Q, P', Q')$ is an **HD representation** of σ

Zero-knowledge

From *ad hoc* to rigorous security proof



Picture by Beppe Rijs

Honest verifier zero-knowledge

Alice (prover)

Generate random
$(E_{com}, \mathrm{End}(E_{com}))$

Compute
$\sigma : E_{chall} \to E_{com}$

Bob (verifier)

Generate random
$\varphi : E_{pk} \to E_{chall}$

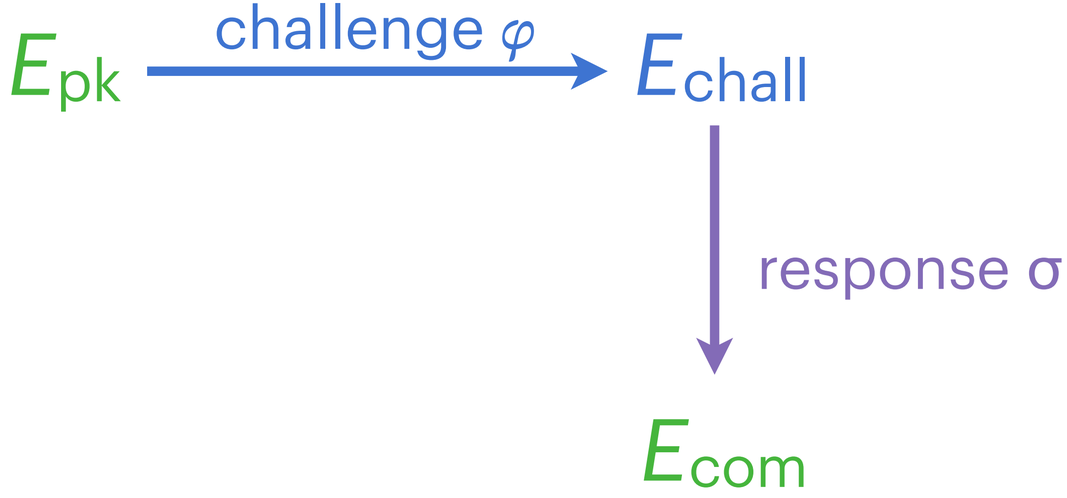
Check that σ is
an isogeny
$E_{chall} \to E_{com}$

E_{com} →

← φ

σ →

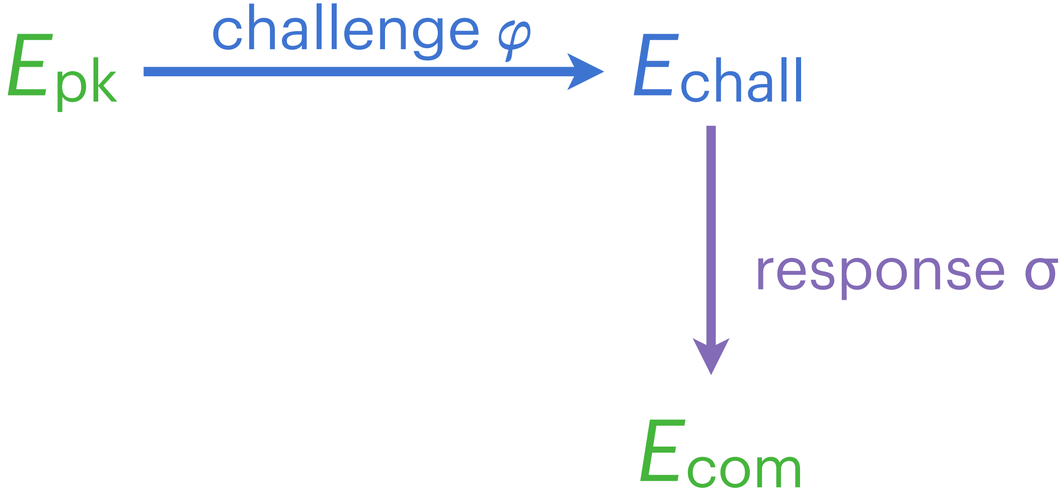
transcript



HV Zero-knowledge = **can generate transcripts with same distribution *without the secret key***

Honest transcript

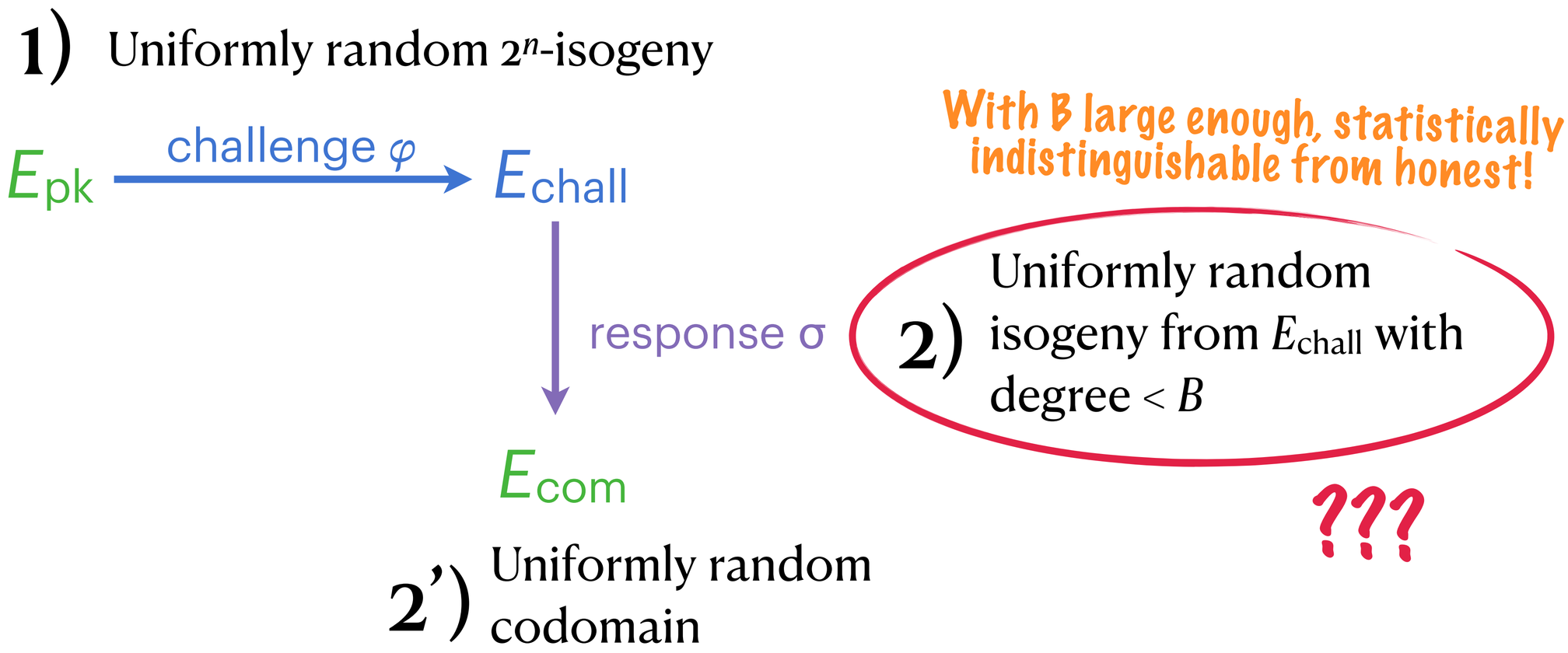
2) Uniformly random 2^n-isogeny



3) Uniformly random in $\mathrm{Hom}(E_{chall}, E_{com})$ with degree $< B$

1) Uniformly random curve

Simulated transcript



RADIO

Random Any-Degree Isogeny Oracle:

- **Input:** an elliptic curve E, a bound $B > 0$
- **Output:** An efficient representation of uniformly distributed isogeny in

$$\{\varphi : E \to ? \mid \deg(\varphi) < B\}$$

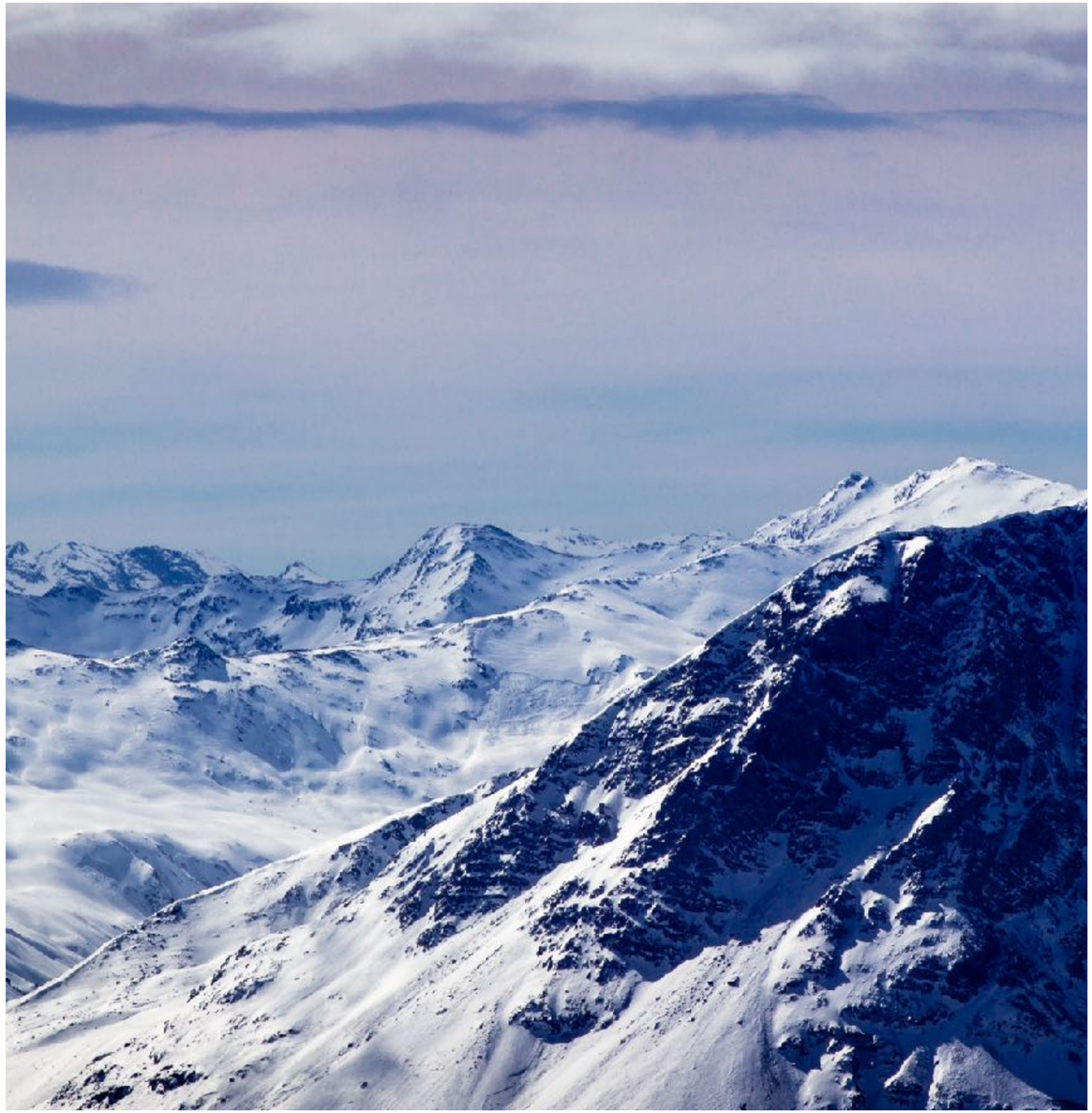
We know how to sample random isogenies of smooth degree...

RADIO only extends that power to any degree. Powerful oracle?

Assumption: EndRing still hard given a RADIO

Dimensions

4 or 8?



Picture by Beppe Rijs

How to verify?

Response: σ represented as (d, P, Q, P', Q') where

- (P, Q) is a basis of $E_{chall}[2^{e/2}]$, with $2^e > 4 \cdot \deg(\sigma)$
- $(P',Q') = (\sigma(P), \sigma(Q))$

Verification: Check that (d, P, Q, P', Q') repr. an isogeny $E_{chall} \to E_{com}$ of deg $< 2^e$:

- Robert's version of SIDH attacks: evaluate a 2^e-isogeny in **dimension 8**
- If $2^e - \deg(\sigma) = a^2 + b^2$: a 2^e-isogeny in **dimension 4** is sufficient

Force $2^e - \deg(\sigma)$ to be a prime $\equiv 1 \bmod 4$

Two versions...

SQIsign 8D: no restriction on deg(σ) + degrees large enough for "stat. indisting."

- **Provably** secure if **EndRing is hard given a RADIO**
- Verification needs isogenies in dimension 8, **impractical**

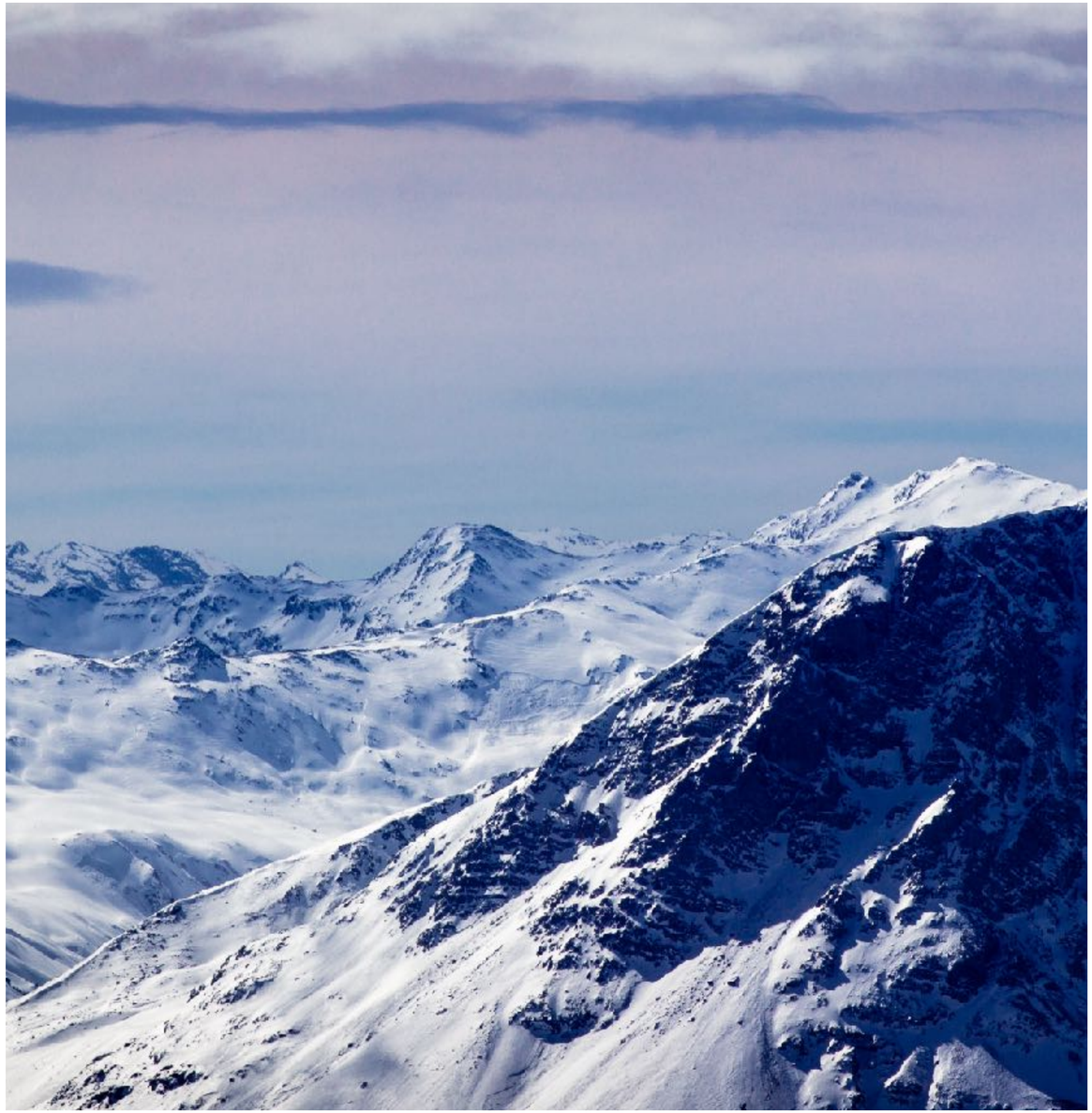
SQIsign 4D: Force 2^e – deg(σ) to be a prime ≡ 1 mod 4 + use smaller degrees

- Security needs **heuristics**, but more compelling, simpler than original SQIsign
- NIST-I level (128 bits security): **64 bytes public key**, **109 bytes signature**
- Verification needs isogenies in dimension 4; some evidence that it is practical

The most compact PK + signature in the post-quantum portfolio (9x smaller than Falcon)

Performance

Of different variants



Picture by Beppe Rijs

Performance

Performance in MCycles., for level of security NIST-I

Caution: numbers come from the papers, sometimes extrapolated, different CPUs, different kinds of optimizations. Should only be indicative of an order of magnitude

	Key gen.	Signing	Verif.
Original SQIsign	1242	4811	99
Optimized SQIsign	286	1354 400ms	21 6ms
SQIsignHD	650	260 74ms	?
SQIsign 2D	92	221 60ms	11 3ms
SQIsign 2D with heuristics	96	124 35ms	11

SQIsign 2D: ongoing work with Andrea Basso, Pierrick Dartois, Luca De Feo, Antonin Leroux, Luciano Maino, Giacomo Pope and Damien Robert