Isogeny-based key exchange

Lorenz Panny

Technische Universiteit Eindhoven

Birmingham, European Union, 16 September 2019

What is this all about?

What is this all about?

- You would like to communicate over the internet.

What is this all about?

- You would like to communicate over the internet.
- It should be secure, so you want to use encryption.

What is this all about?

- You would like to communicate over the internet.
- It should be secure, so you want to use encryption.
- But you haven't agreed on a secret key yet!

What is this all about?

- You would like to communicate over the internet.
- It should be secure, so you want to use encryption.
- But you haven't agreed on a secret key yet!

A *key exchange* is a method for (typically) two parties
to negotiate a shared secret key over an insecure channel.

What is this all about?

- You would like to communicate over the internet.
- It should be secure, so you want to use encryption.
- But you haven't agreed on a secret key yet!

A *key exchange* is a method for (typically) two parties to negotiate a shared secret key over an insecure channel.

(For now, "insecure" means someone is listening in on everything being sent. There is also a notion of *active* attackers who mess with data on the wire.)

Diffie–Hellman key exchange ’76

Diffie–Hellman key exchange '76

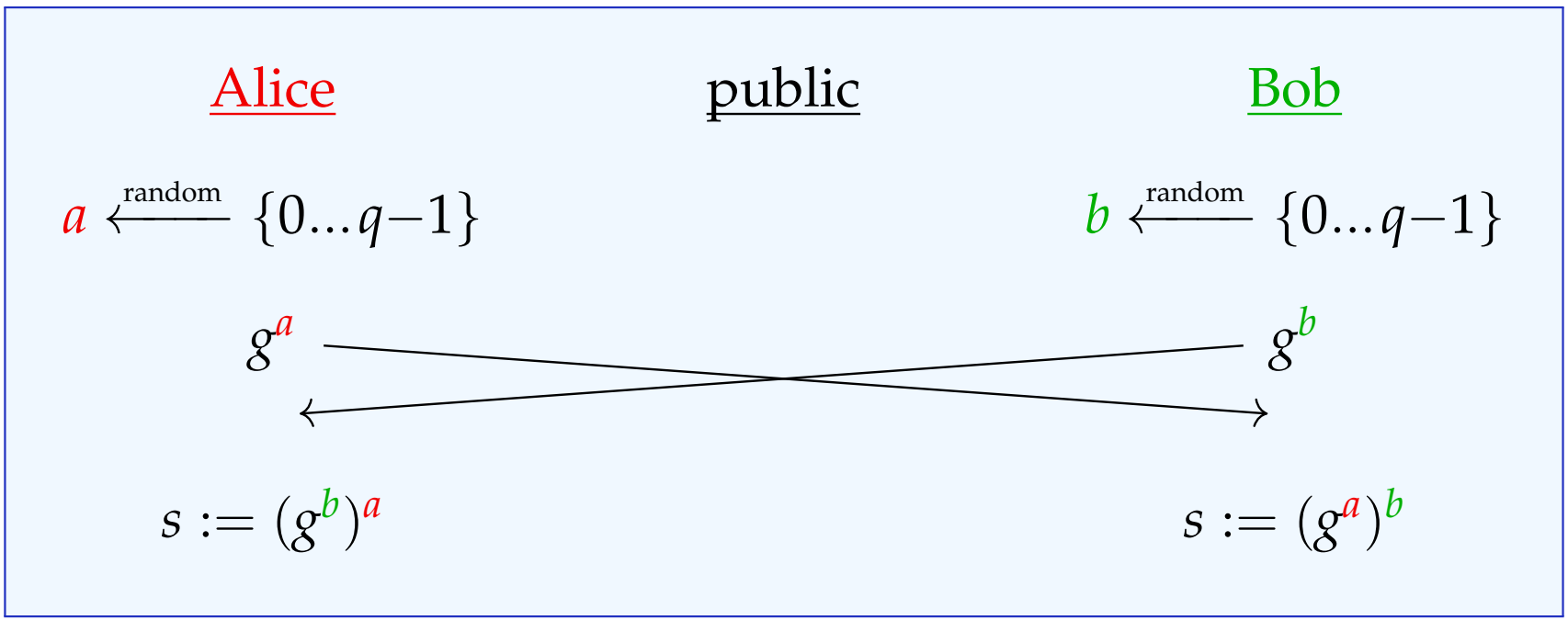
Public parameters:

- a finite group G (traditionally $\mathbb{F}_p^*$, today elliptic curves)
- an element $g \in G$ of prime order q

Diffie–Hellman key exchange '76

Public parameters:

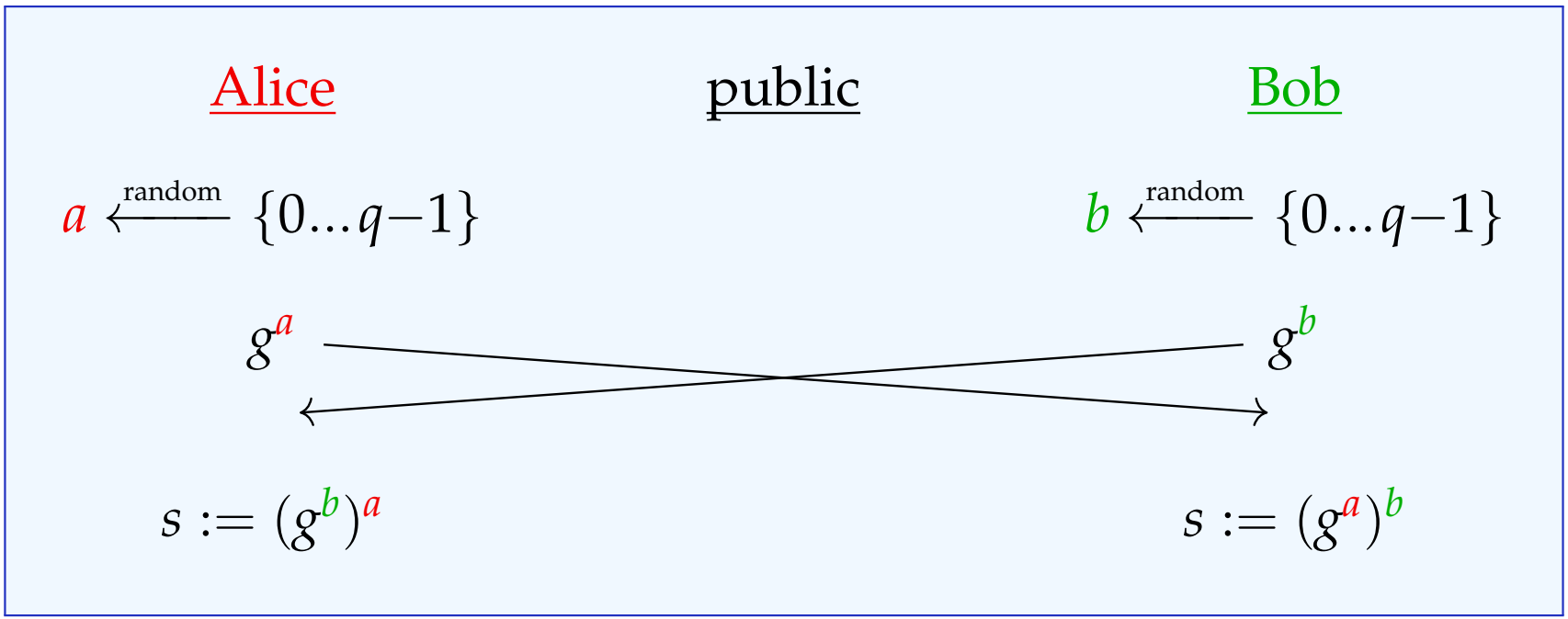
- a finite group G (traditionally $\mathbb{F}_p^*$, today elliptic curves)
- an element $g \in G$ of prime order q



Diffie–Hellman key exchange '76

Public parameters:

- a finite group G (traditionally $\mathbb{F}_p^*$, today elliptic curves)
- an element $g \in G$ of prime order q



Fundamental reason this works: $\cdot^a$ and $\cdot^b$ are commutative!

Diffie–Hellman: Bob vs. Eve

Bob

1. Set $t \leftarrow g$.
2. Set $t \leftarrow t \cdot g$.
3. Set $t \leftarrow t \cdot g$.
4. Set $t \leftarrow t \cdot g$.

...

$b-2$. Set $t \leftarrow t \cdot g$.

$b-1$. Set $t \leftarrow t \cdot g$.

b. Publish $B \leftarrow t \cdot g$.

Diffie–Hellman: Bob vs. Eve

Bob

1. Set $t \leftarrow g$.
2. Set $t \leftarrow t \cdot g$.
3. Set $t \leftarrow t \cdot g$.
4. Set $t \leftarrow t \cdot g$.

...

$b-2$. Set $t \leftarrow t \cdot g$.

$b-1$. Set $t \leftarrow t \cdot g$.

b. Publish $B \leftarrow t \cdot g$.

Is this a good idea?

Diffie–Hellman: Bob vs. Eve

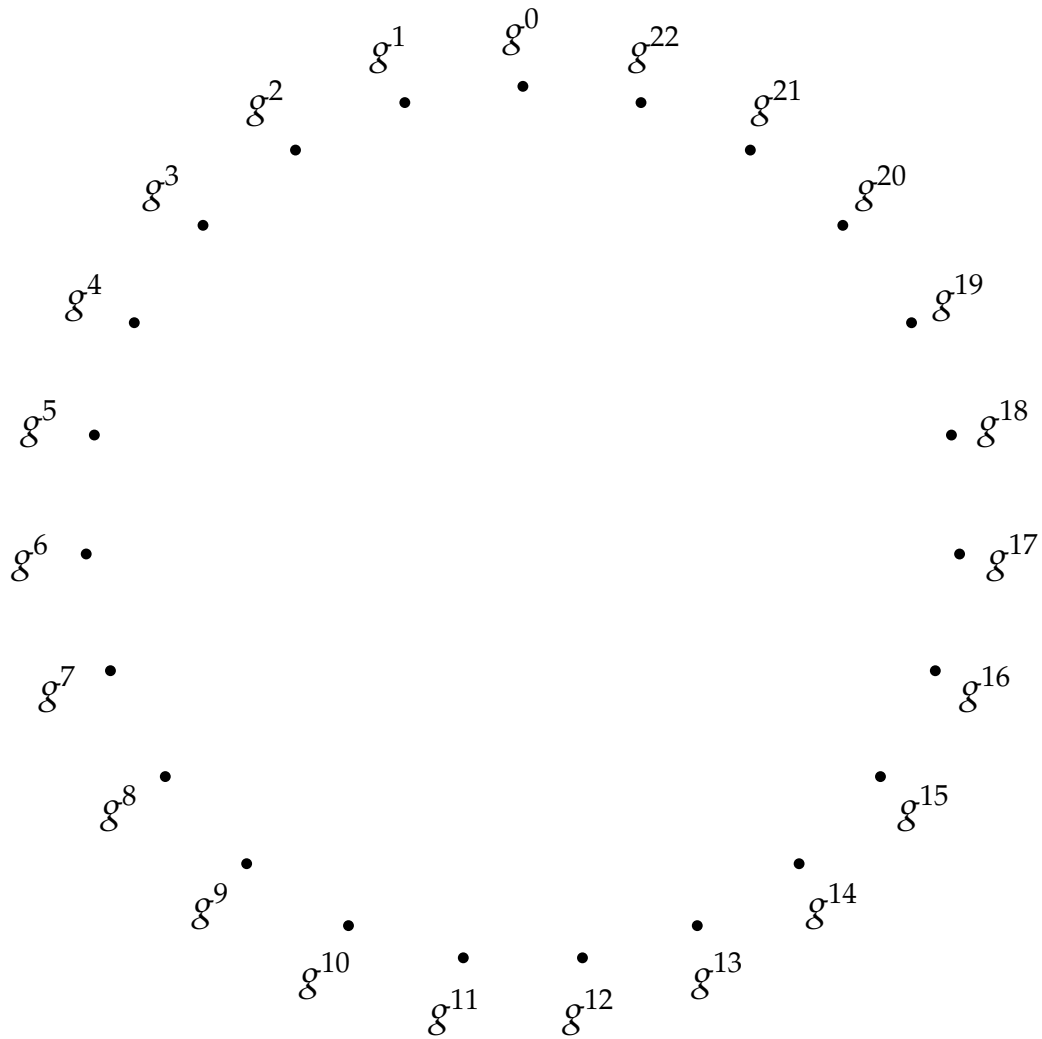
Bob	**Attacker** Eve	
1. Set $t \leftarrow g$.	1. Set $t \leftarrow g$.	If $t = B$ return 1.
2. Set $t \leftarrow t \cdot g$.	2. Set $t \leftarrow t \cdot g$.	If $t = B$ return 2.
3. Set $t \leftarrow t \cdot g$.	3. Set $t \leftarrow t \cdot g$.	If $t = B$ return 3.
4. Set $t \leftarrow t \cdot g$.	4. Set $t \leftarrow t \cdot g$.	If $t = B$ return 3.
...	...	
$b-2$. Set $t \leftarrow t \cdot g$.	$b-2$. Set $t \leftarrow t \cdot g$.	If $t = B$ return $b-2$.
$b-1$. Set $t \leftarrow t \cdot g$.	$b-1$. Set $t \leftarrow t \cdot g$.	If $t = B$ return $b-1$.
b. Publish $B \leftarrow t \cdot g$.	b. Set $t \leftarrow t \cdot g$.	If $t = B$ return b.
	$b+1$. Set $t \leftarrow t \cdot g$.	If $t = B$ return $b+1$.
	$b+2$. Set $t \leftarrow t \cdot g$.	If $t = B$ return $b+2$.
	...	

Diffie–Hellman: Bob vs. Eve

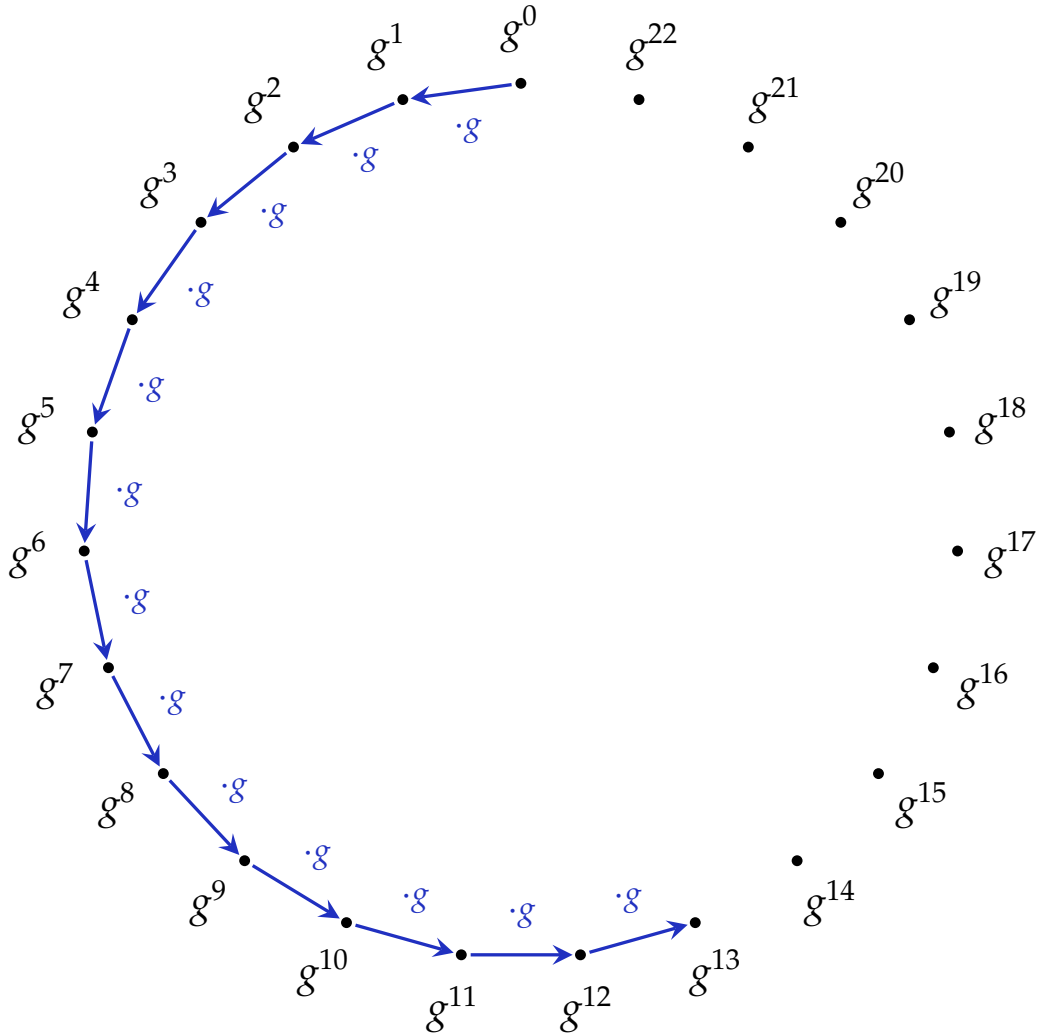
Bob	**Attacker** Eve
1. Set $t \leftarrow g$.	1. Set $t \leftarrow g$. If $t = B$ return 1.
2. Set $t \leftarrow t \cdot g$.	2. Set $t \leftarrow t \cdot g$. If $t = B$ return 2.
3. Set $t \leftarrow t \cdot g$.	3. Set $t \leftarrow t \cdot g$. If $t = B$ return 3.
4. Set $t \leftarrow t \cdot g$.	4. Set $t \leftarrow t \cdot g$. If $t = B$ return 3.
...	...
$b-2$. Set $t \leftarrow t \cdot g$.	$b-2$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b-2$.
$b-1$. Set $t \leftarrow t \cdot g$.	$b-1$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b-1$.
b. Publish $B \leftarrow t \cdot g$.	b. Set $t \leftarrow t \cdot g$. If $t = B$ return b.
	$b+1$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b+1$.
	$b+2$. Set $t \leftarrow t \cdot g$. If $t = B$ return $b+2$.
	...

Effort for both: $O(\#G)$. Bob needs to be smarter.

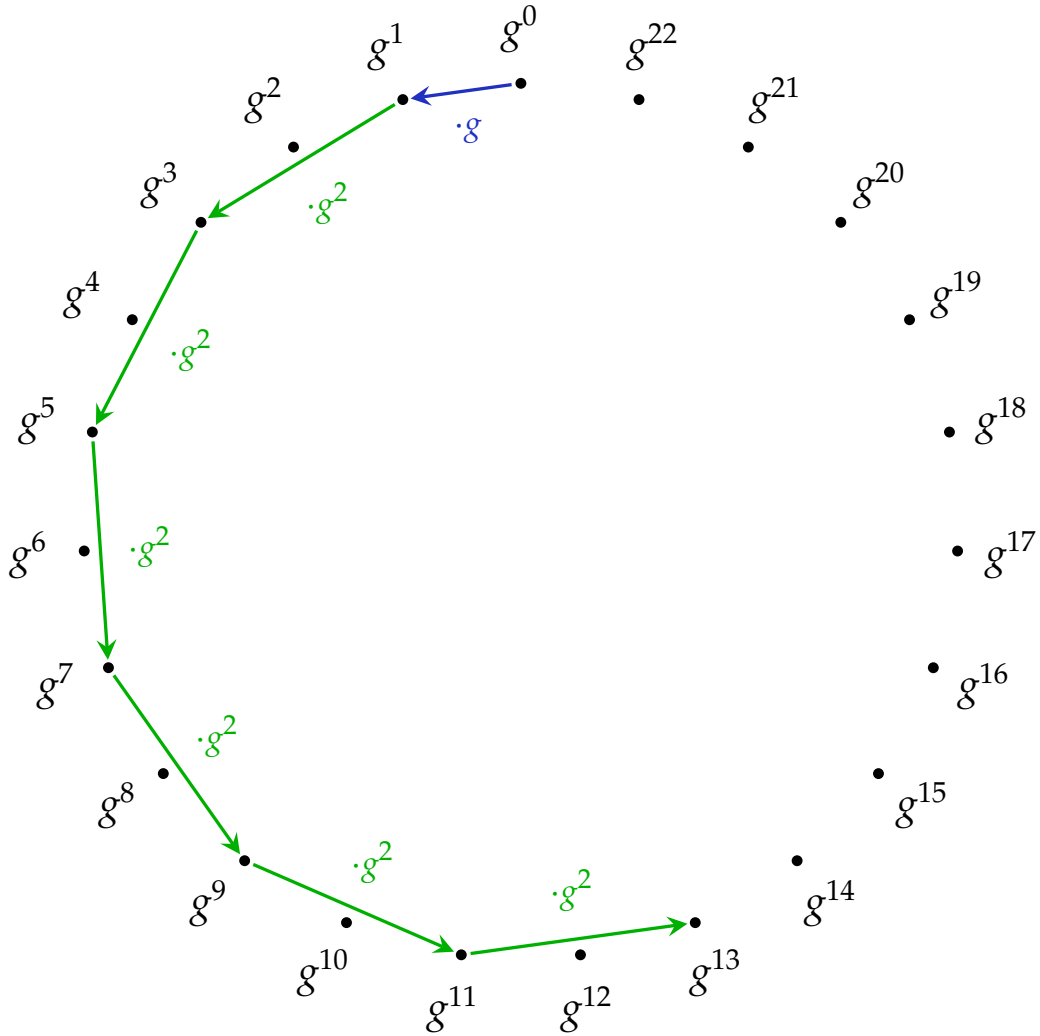
(This attacker is also kind of dumb, but that doesn't matter for my point here.)


g^0
g^1
g^2
g^3
g^4
g^5
g^6
g^7
g^8
g^9
g^{10}
g^{11}
g^{12}
g^{13}
g^{14}
g^{15}
g^{16}
g^{17}
g^{18}
g^{19}
g^{20}
g^{21}
g^{22}

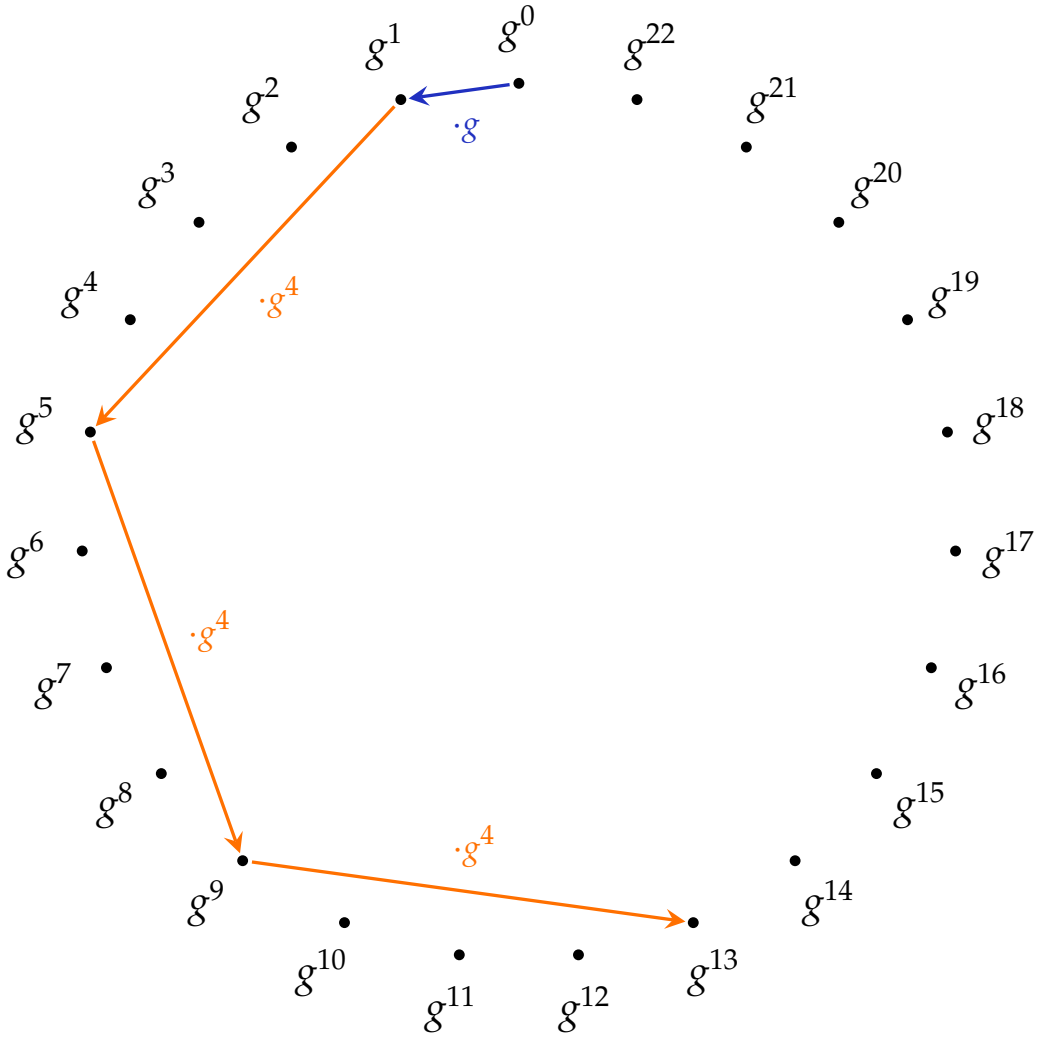
multiply


g^0
g^1
g^2
g^3
g^4
g^5
g^6
g^7
g^8
g^9
g^{10}
g^{11}
g^{12}
g^{13}
g^{14}
g^{15}
g^{16}
g^{17}
g^{18}
g^{19}
g^{20}
g^{21}
g^{22}
$\cdot g$

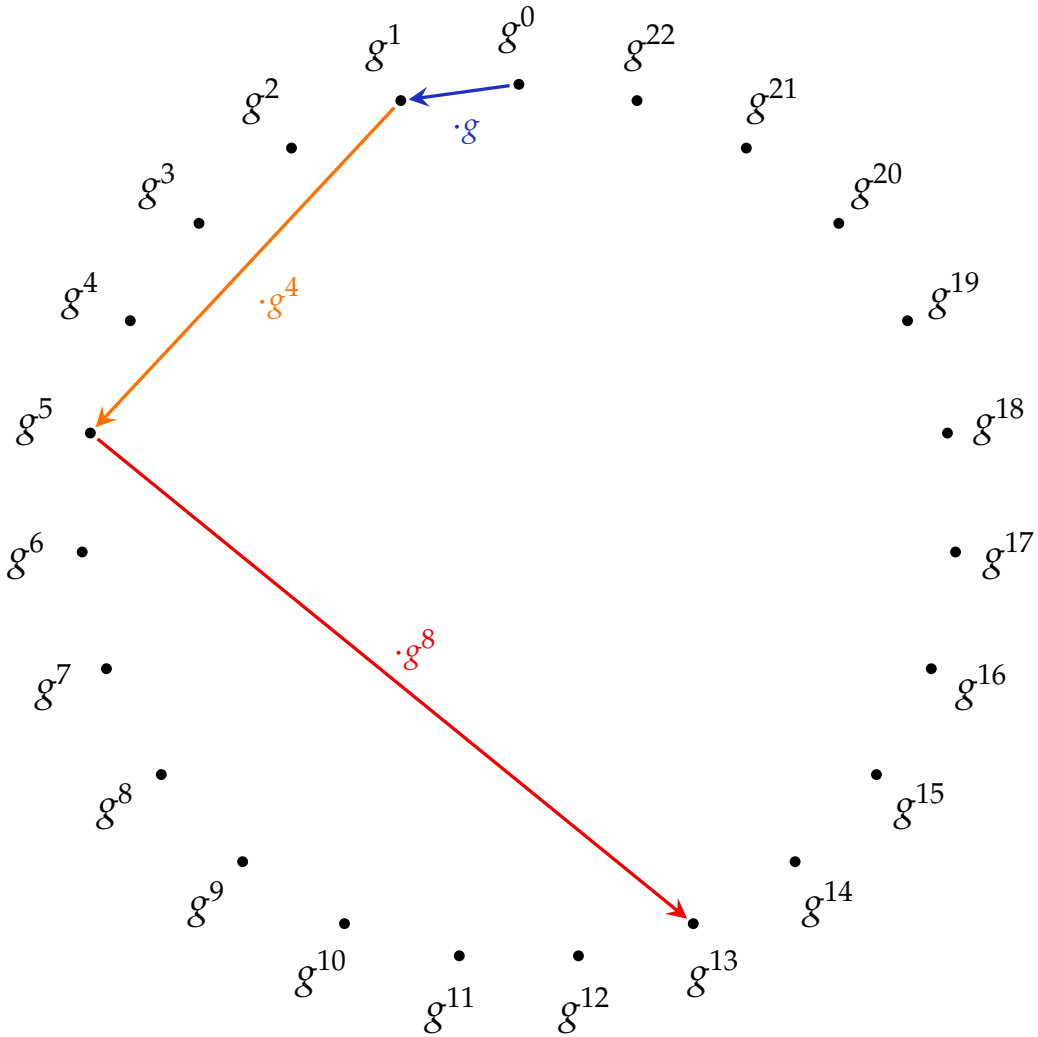
Square-and-multiply



Square-and-multiply-and-square-and-multiply



Square-and-multiply-and-square-and-multiply-and-squ



With square-and-multiply, applying b takes $\Theta(\log \#G)$.

For well-chosen groups, recovering b takes $\Theta(\sqrt{\#G})$.

$\rightsquigarrow$ Exponential separation!

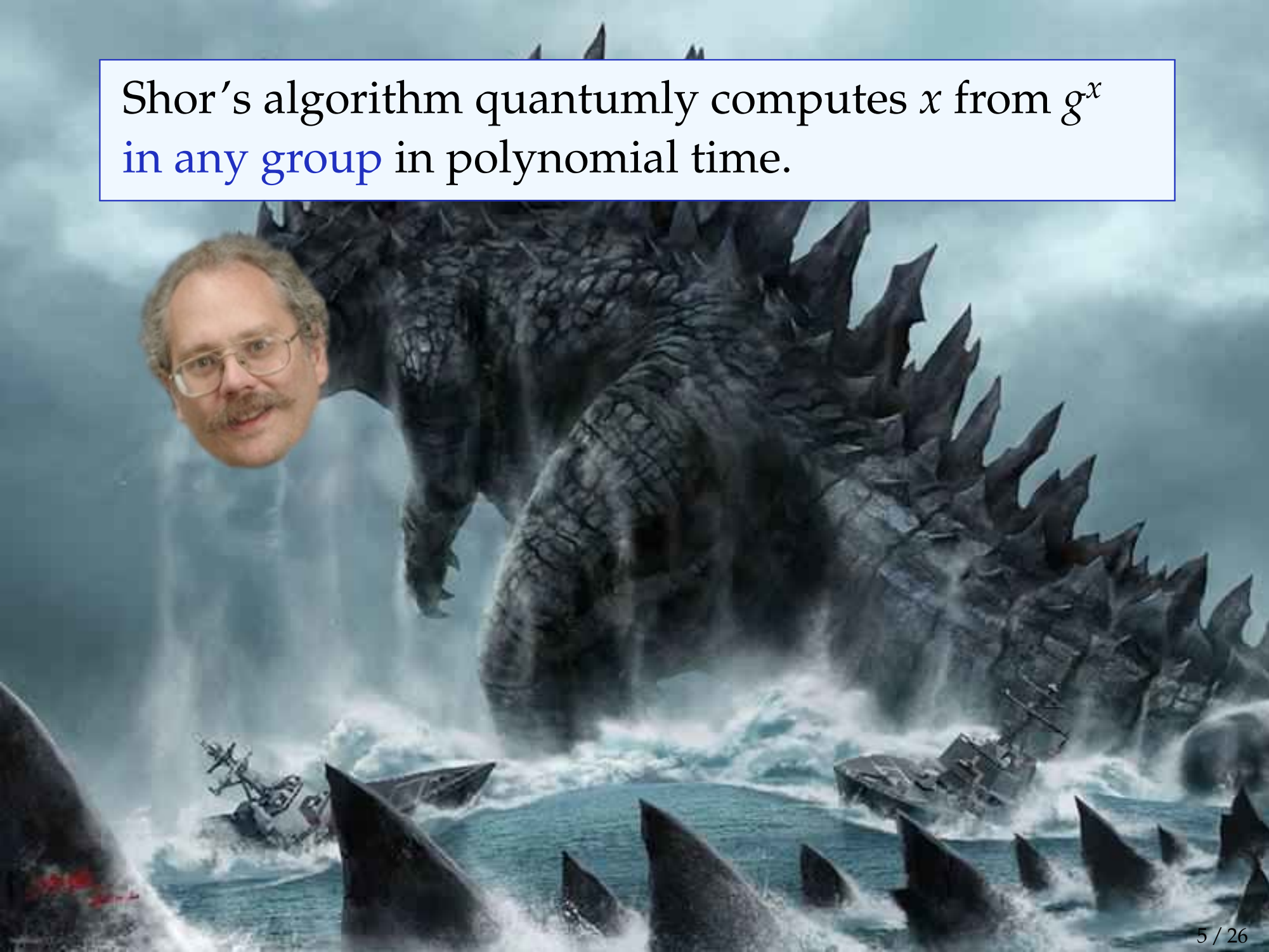
With square-and-multiply, applying b takes $\Theta(\log \#G)$.

For well-chosen groups, recovering b takes $\Theta(\sqrt{\#G})$.

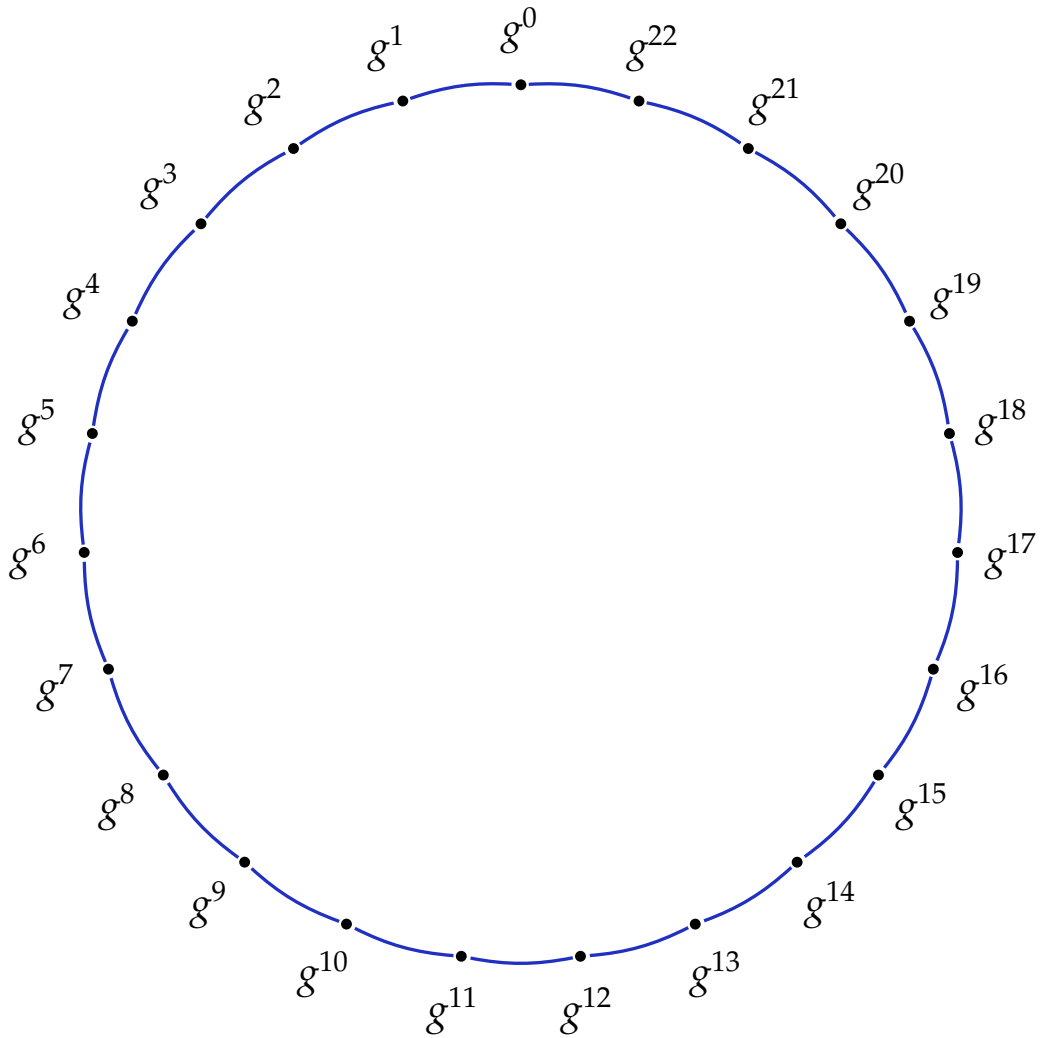
$\rightsquigarrow$ Exponential separation!

...and they lived happily ever after?

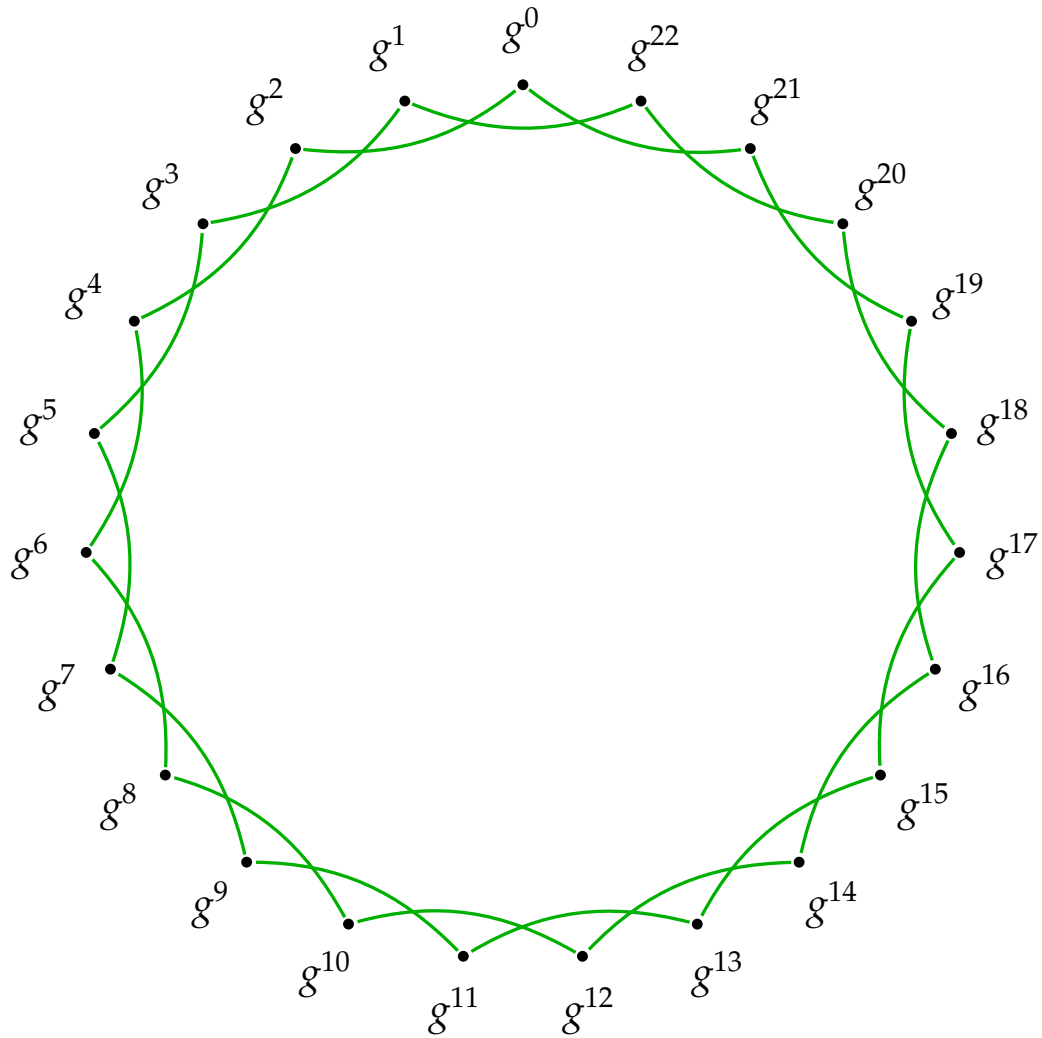
Shor's algorithm quantumly computes x from g^x in any group in polynomial time.



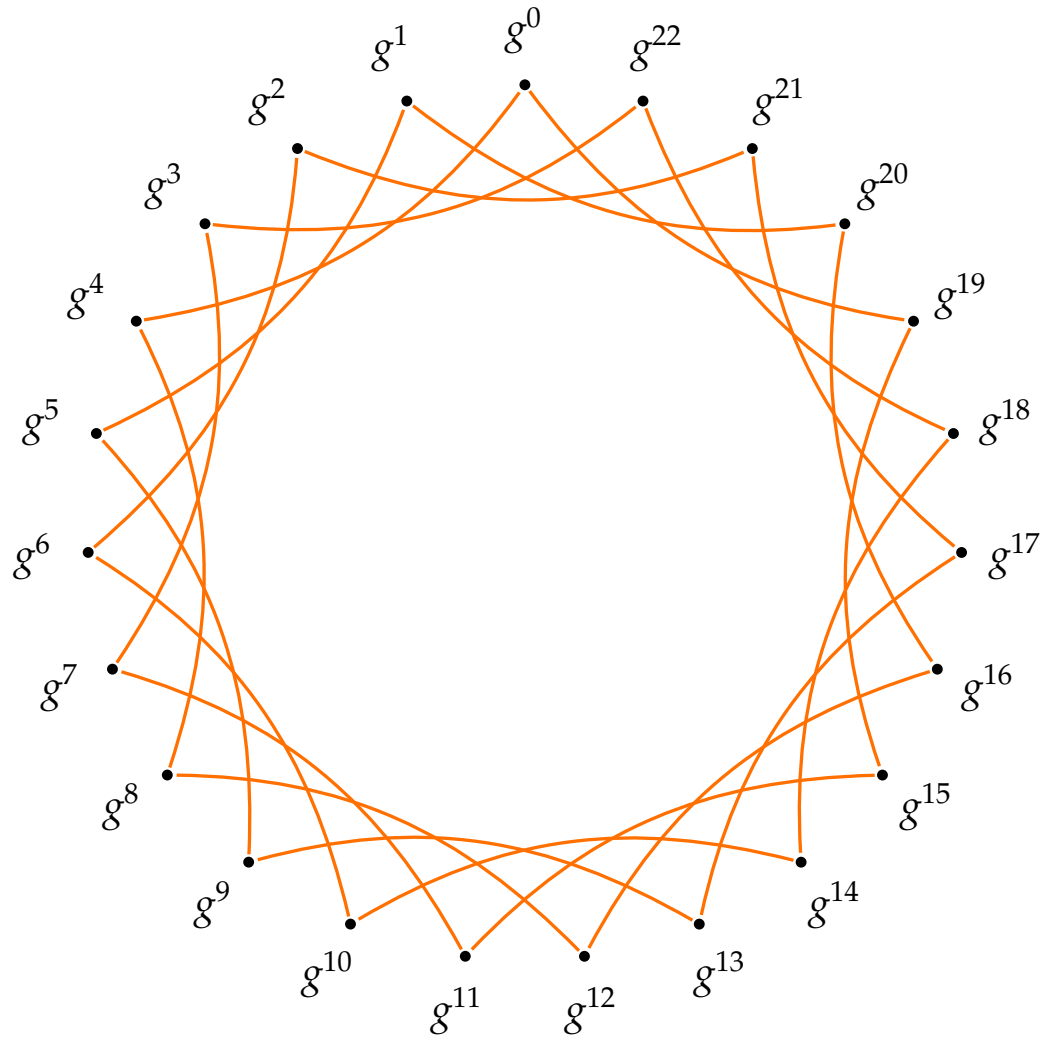
Square-and-multiply as graphs



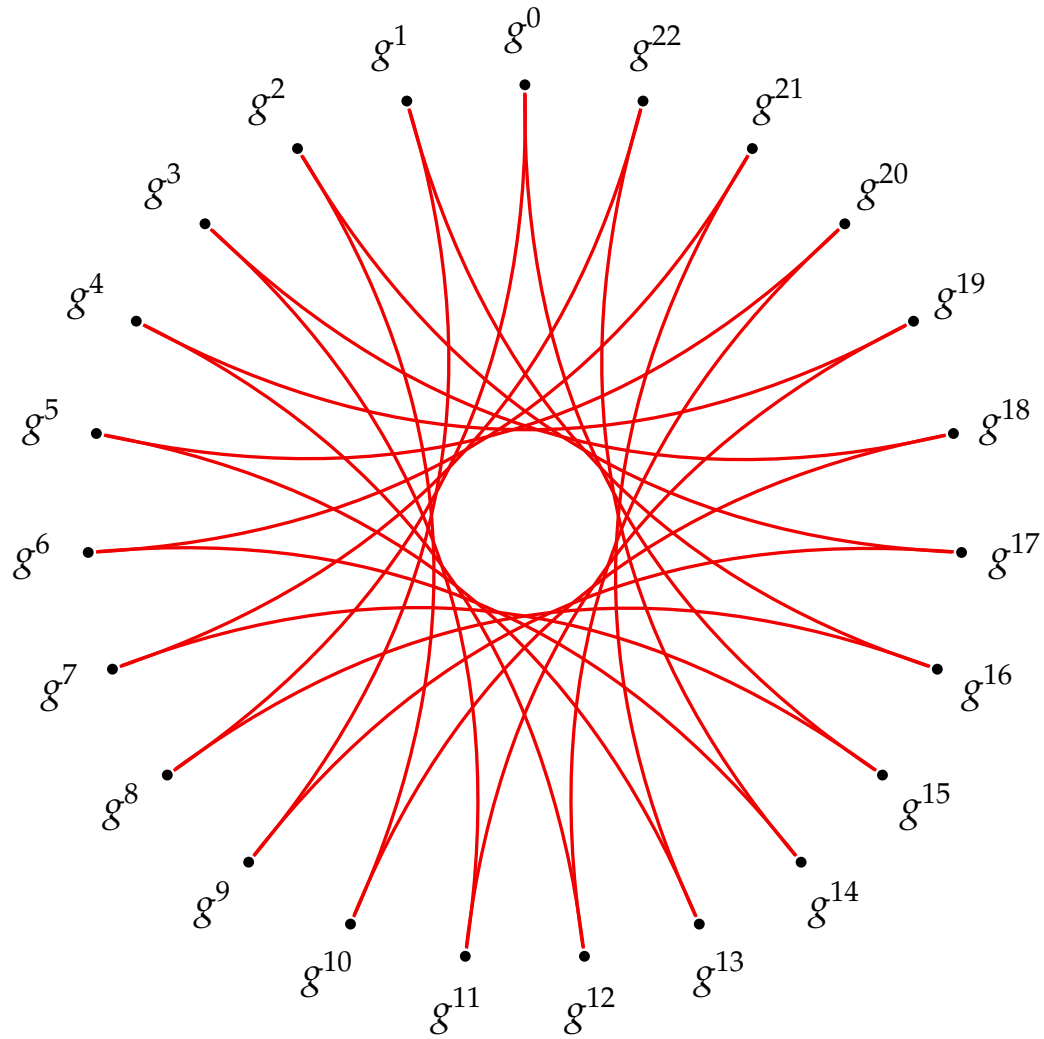
Square-and-multiply as graphs



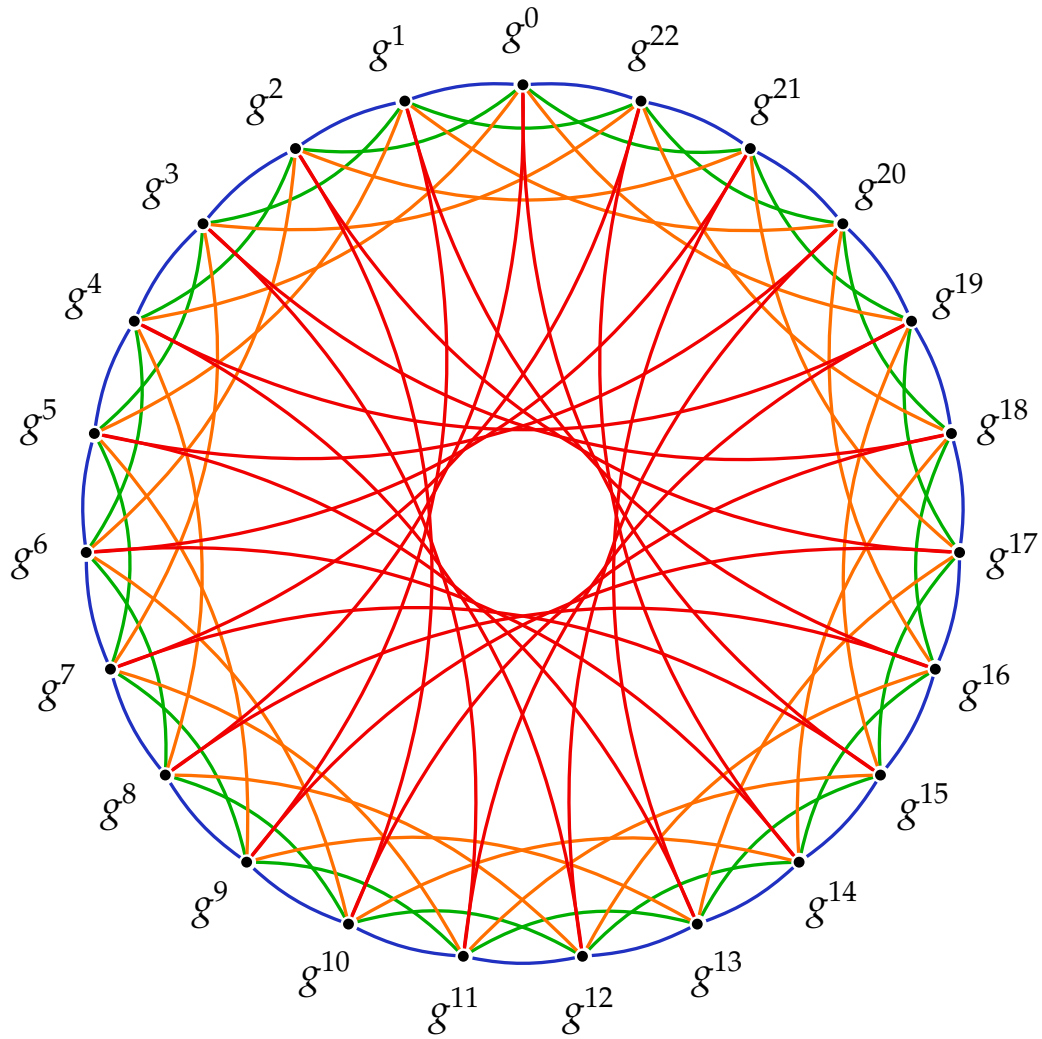
Square-and-multiply as graphs



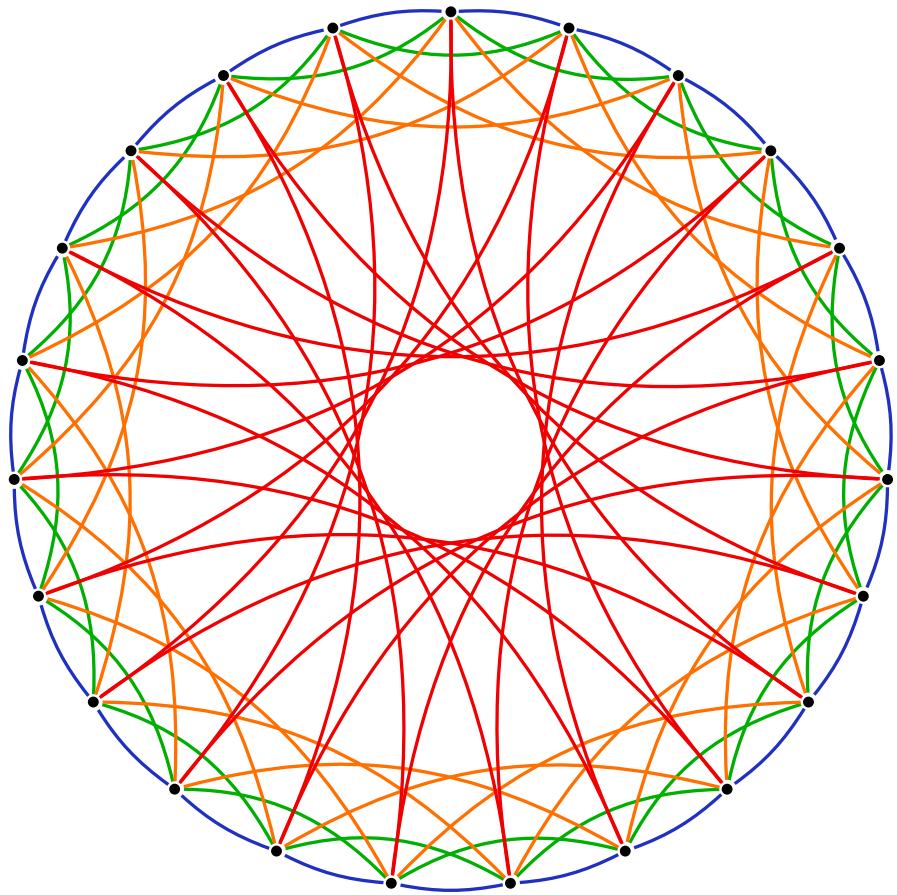
Square-and-multiply as graphs



Square-and-multiply as a graph



Square-and-multiply as a graph



Fast mixing: paths of length $\log(\#\,\text{nodes})$ to everywhere.

New plan: Get rid of the group, keep the graph.

Big picture 🔍🔍

- Isogenies are a source of exponentially-sized graphs.

Big picture 🔍🔍

- Isogenies are a source of exponentially-sized graphs.
- We can walk efficiently on these graphs.

Big picture 🔎🔎

- Isogenies are a source of exponentially-sized graphs.
- We can walk efficiently on these graphs.
- Fast mixing: short paths to (almost) all nodes.

Big picture 🔎🔎

- Isogenies are a source of exponentially-sized graphs.
- We can walk efficiently on these graphs.
- Fast mixing: short paths to (almost) all nodes.
- No efficient* algorithms to recover paths from endpoints.
  (*Both* classical and quantum!)

Big picture 🔎🔎

- Isogenies are a source of exponentially-sized graphs.
- We can walk efficiently on these graphs.
- Fast mixing: short paths to (almost) all nodes.
- No efficient* algorithms to recover paths from endpoints.
  (*Both* classical and quantum!)
- Enough structure to navigate the graph meaningfully.
  That is: some *well-behaved* "directions" to describe paths.

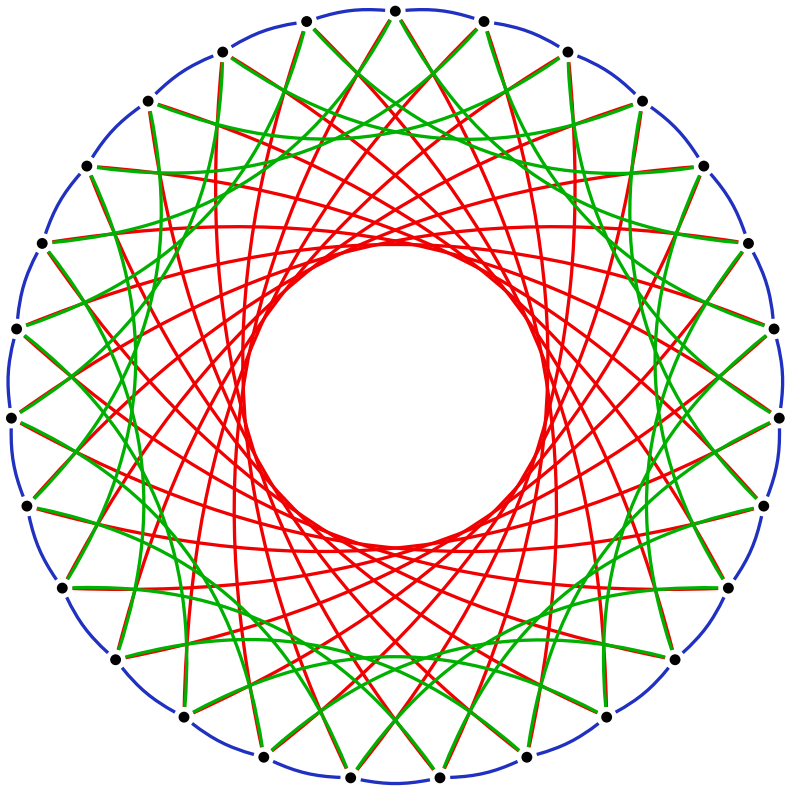
Big picture 🔍🔍

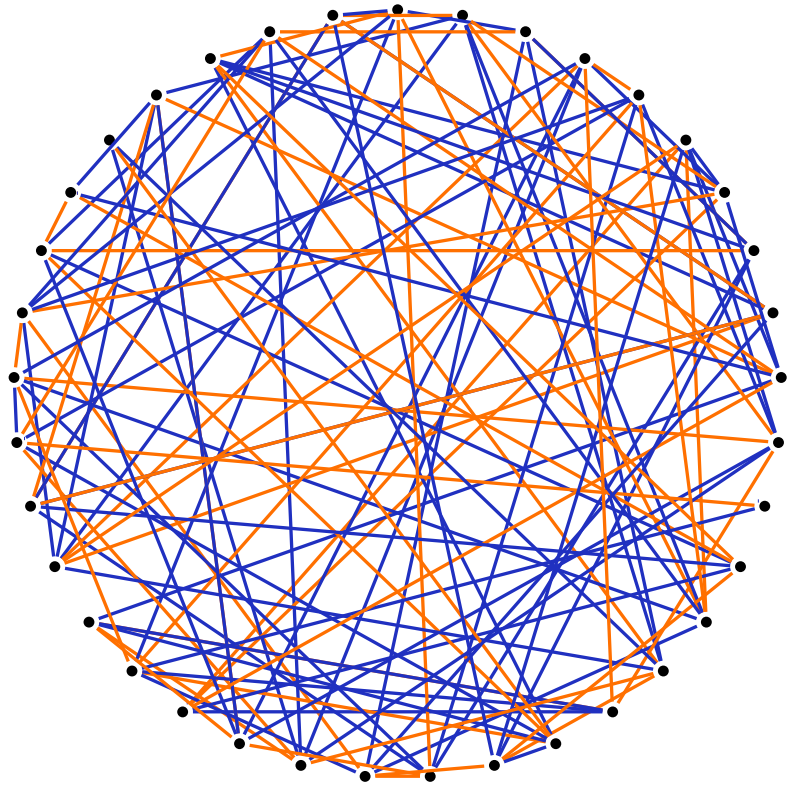
- Isogenies are a source of exponentially-sized graphs.
- We can walk efficiently on these graphs.
- Fast mixing: short paths to (almost) all nodes.
- No efficient* algorithms to recover paths from endpoints.
  (*Both* classical and quantum!)
- Enough structure to navigate the graph meaningfully.
  That is: some *well-behaved* "directions" to describe paths.

It is easy to construct graphs that satisfy *almost* all of these —
but getting all at once seems rare. Isogenies!

The beauty and the beast

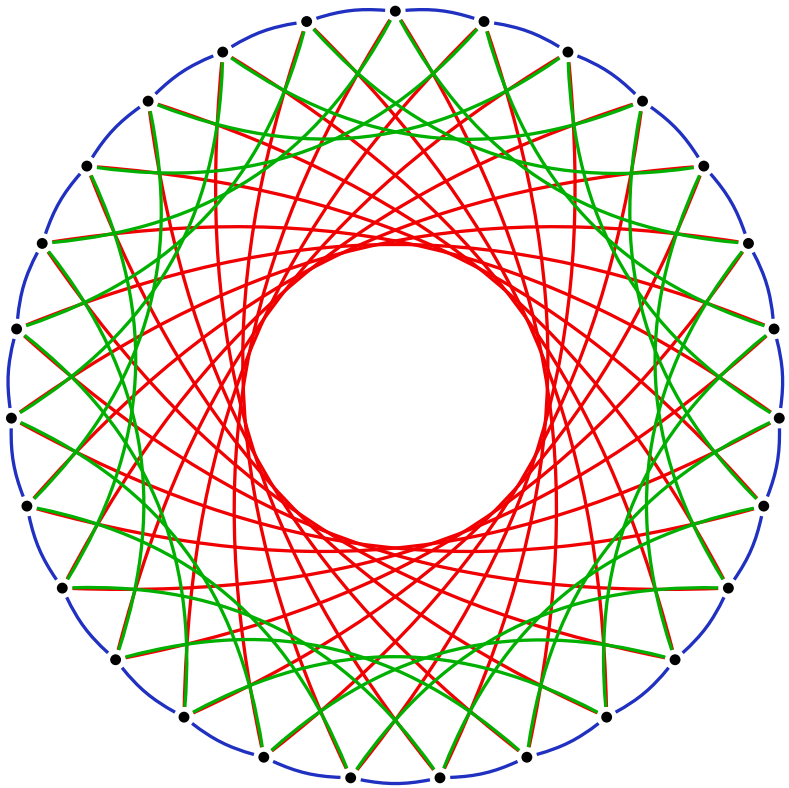
Components of well-chosen isogeny graphs look like this:

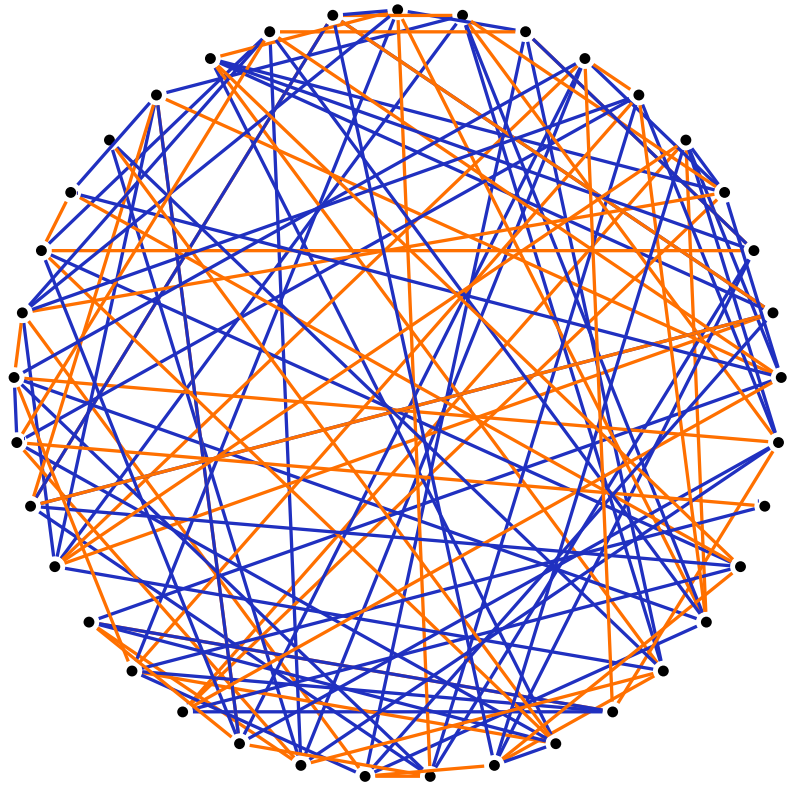




The beauty and the beast

Components of well-chosen isogeny graphs look like this:

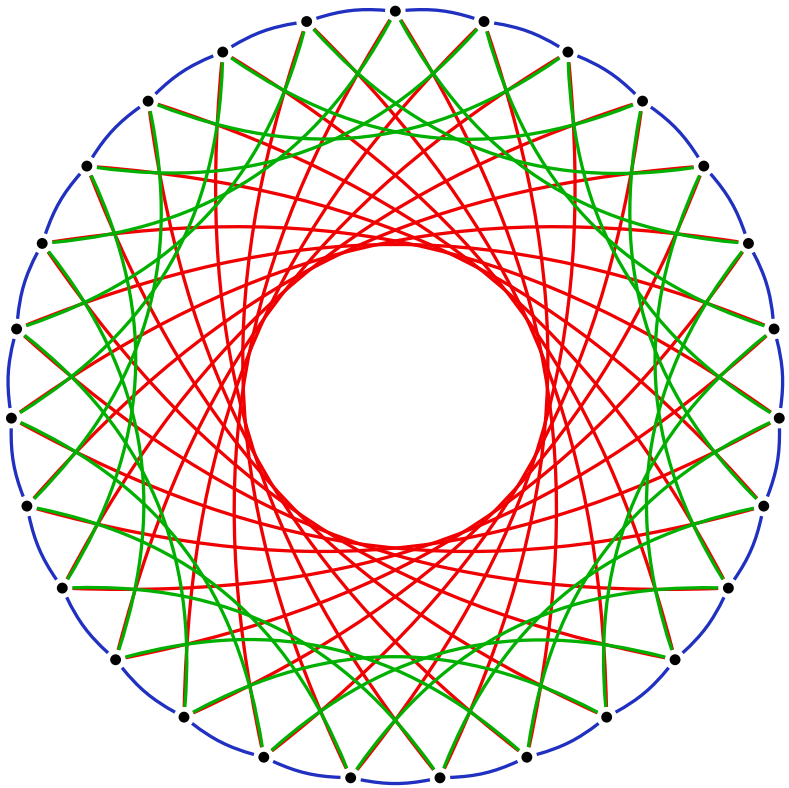


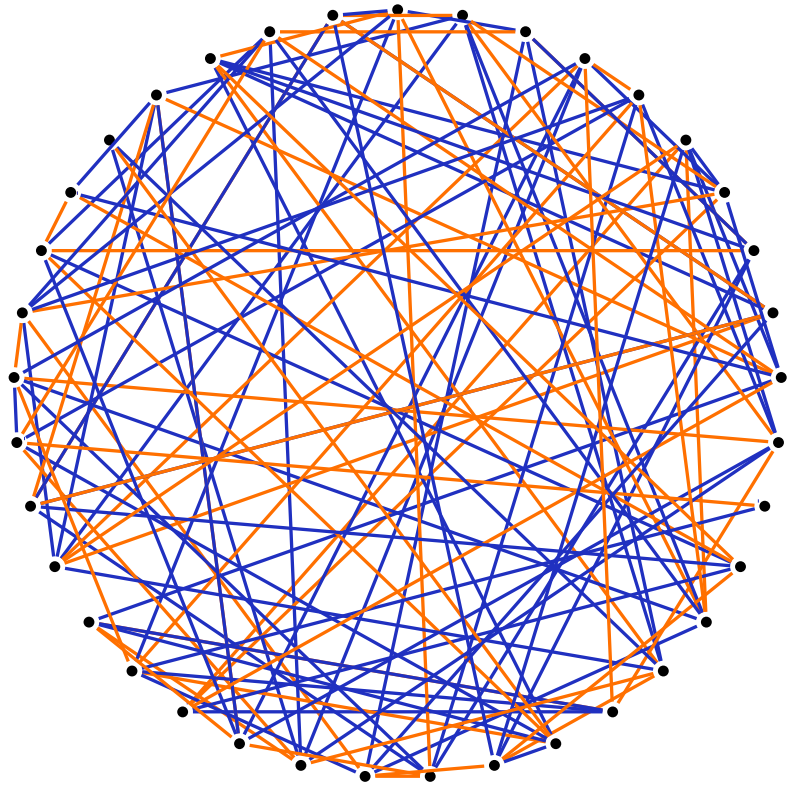


Which of these is good for crypto?

The beauty and the beast

Components of well-chosen isogeny graphs look like this:

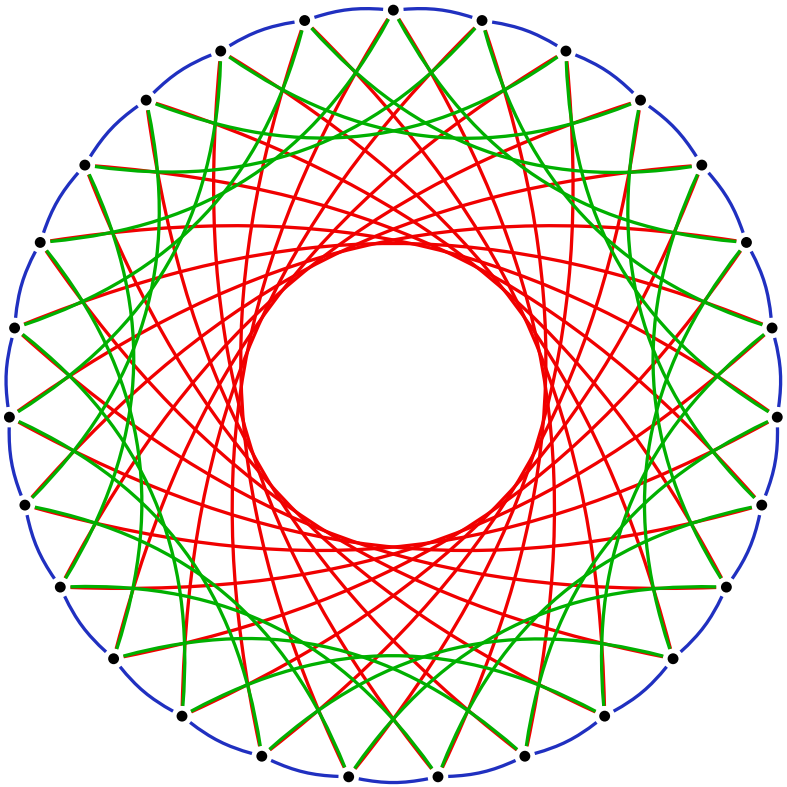




Which of these is good for crypto? **Both.**

The beauty and the beast

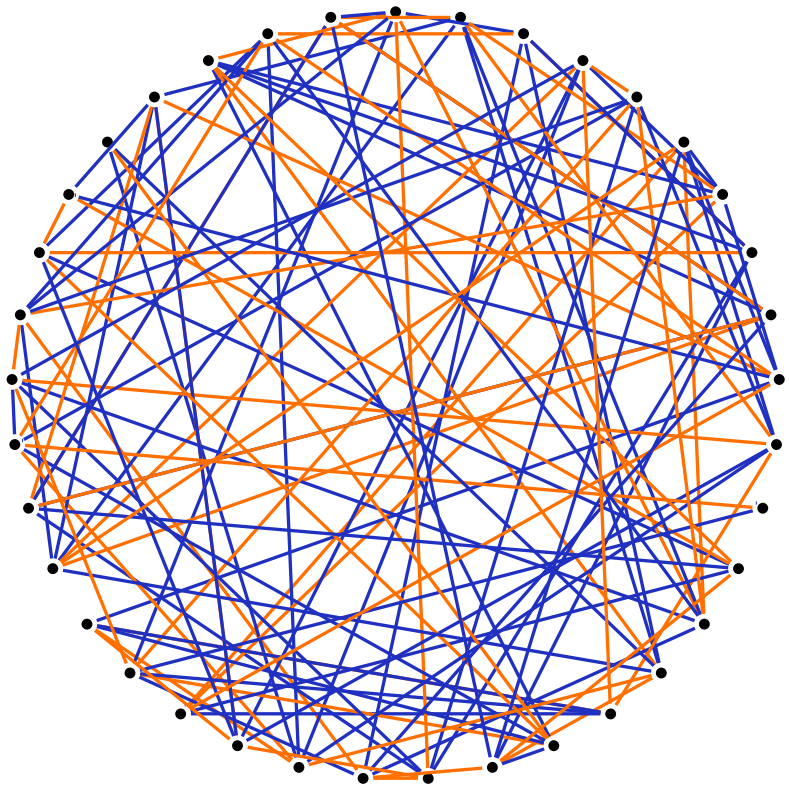
At this time, there are <u>two distinct families</u> of systems:



$q = p$

CSIDH ['siːˌsaɪd]

`https://csidh.isogeny.org`



$q = p^2$

SIDH

`https://sike.org`

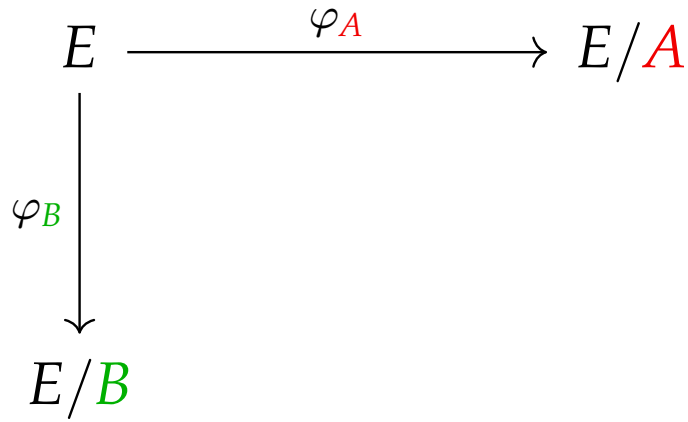
Isogeny key exchange: High-level view

E

Isogeny key exchange: High-level view

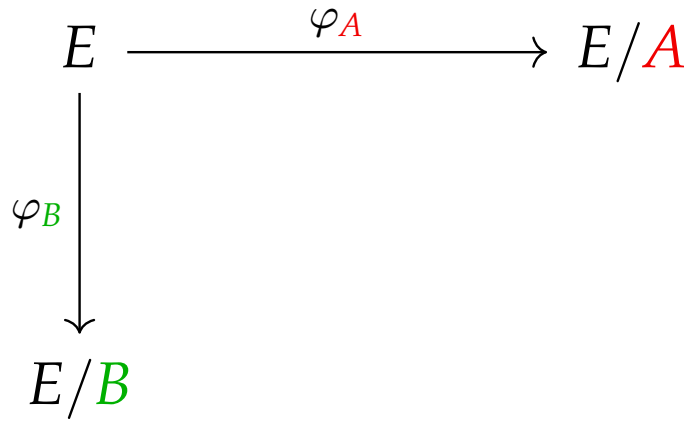
- Alice & Bob pick secret subgroups A and B of E.

Isogeny key exchange: High-level view



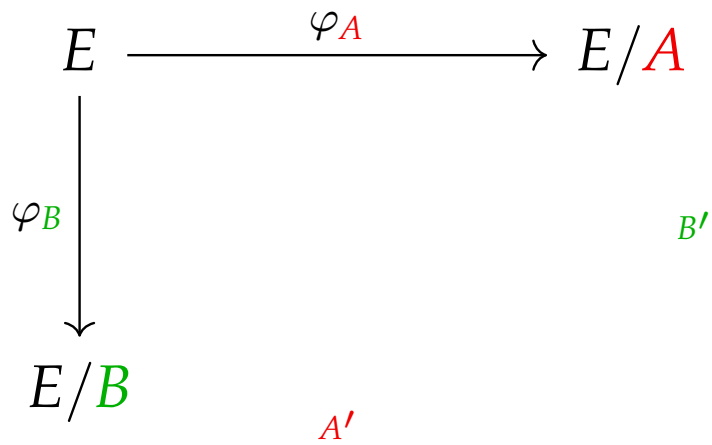
- Alice & Bob pick secret subgroups A and B of E.
- Alice computes $\varphi_A : E \to E/A$; Bob computes $\varphi_B : E \to E/B$.
  (These isogenies correspond to walking on the isogeny graph.)

Isogeny key exchange: High-level view



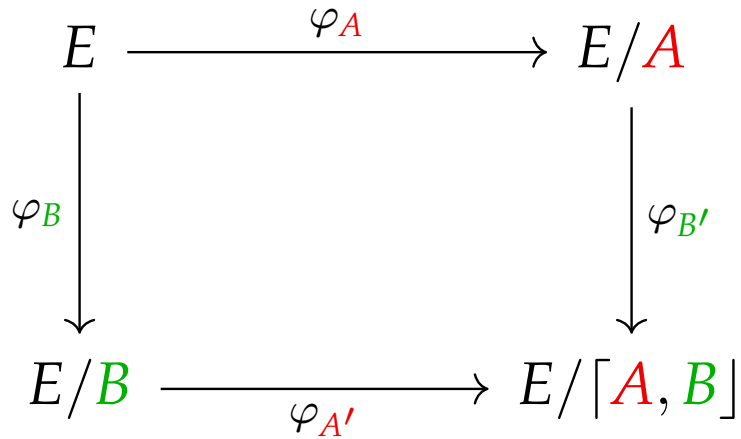
- Alice & Bob pick secret subgroups A and B of E.
- Alice computes $\varphi_A : E \to E/A$; Bob computes $\varphi_B : E \to E/B$.
  (These isogenies correspond to walking on the isogeny graph.)
- Alice and Bob transmit the values E/A and E/B.

Isogeny key exchange: High-level view



- Alice & Bob pick secret subgroups A and B of E.
- Alice computes $\varphi_A : E \to E/A$; Bob computes $\varphi_B : E \to E/B$.
  (These isogenies correspond to walking on the isogeny graph.)
- Alice and Bob transmit the values E/A and E/B.
- Alice somehow obtains $A' := \text{shift\_}\varphi_B(A)$. (Similar for Bob.)

Isogeny key exchange: High-level view



- Alice & Bob pick secret subgroups A and B of E.
- Alice computes $\varphi_A : E \to E/A$; Bob computes $\varphi_B : E \to E/B$.
  (These isogenies correspond to walking on the isogeny graph.)
- Alice and Bob transmit the values E/A and E/B.
- Alice somehow obtains $A' := \text{shift\_}\varphi_B(A)$. (Similar for Bob.)
- They both compute the shared secret
  $$(E/B)/A' \cong E/\lceil A, B \rfloor \cong (E/A)/B'.$$


CSIDH ['siːˌsaɪd]

A brief history of the seaside

A brief history of the seaside

> And God said, Let the waters under the heaven be gathered together unto one place, and let the dry land appear: and it was so.
>
> And God called the dry land Earth; and the gathering together of the waters called he Seas: and God saw that it was good.
>
> [King James Bible, Genesis 1:9–10]

A brief history of ~~the seaside~~ CSIDH

Recall from Luca's talk:

> Sometimes, there is a (free & transitive) group action of $\mathrm{cl}(\mathcal{O})$ on a set of curves with endomorphism ring $\mathcal{O}$.

A brief history of ~~the seaside~~ CSIDH

Recall from Luca's talk:

> Sometimes, there is a (free & transitive) group action of $\mathrm{cl}(\mathcal{O})$ on a set of curves with endomorphism ring $\mathcal{O}$.

[Couveignes '97/'06], independently [Rostovtsev–Stolbunov '06]:

> Use this group action on ordinary curves for Diffie–Hellman.

A brief history of ~~the seaside~~ CSIDH

Recall from Luca's talk:

> Sometimes, there is a (free & transitive) group action of $\mathrm{cl}(\mathcal{O})$ on a set of curves with endomorphism ring $\mathcal{O}$.

[Couveignes '97/'06], independently [Rostovtsev–Stolbunov '06]:

> Use this group action on ordinary curves for Diffie–Hellman.

[De Feo–Kieffer–Smith '18]:

> Massive speedups, but still unbearably slow.

A brief history of ~~the seaside~~ CSIDH

Recall from Luca's talk:

> Sometimes, there is a (free & transitive) group action of $\mathrm{cl}(\mathcal{O})$ on a set of curves with endomorphism ring $\mathcal{O}$.

[Couveignes '97/'06], independently [Rostovtsev–Stolbunov '06]:

> Use this group action on ordinary curves for Diffie–Hellman.

[De Feo–Kieffer–Smith '18]:

> Massive speedups, but still unbearably slow.

[Castryck–Lange–Martindale–Panny–Renes '18]:

> Switch to supersingular curves $\implies$ "practical" performance.

CSIDH in one slide

CSIDH in one slide

- Choose some small odd primes $\ell_1, ..., \ell_n$.
- Make sure $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ is prime.

CSIDH in one slide

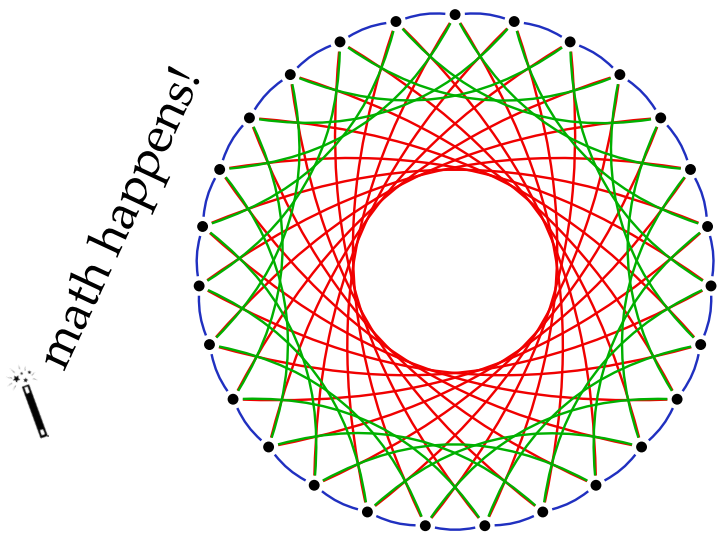
- Choose some small odd primes $\ell_1, \ldots, \ell_n$.
- Make sure $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ is prime.
- Let $X = \{y^2 = x^3 + Ax^2 + x$ over $\mathbb{F}_p$ with $p+1$ points$\}$.

CSIDH in one slide

- Choose some small odd primes $\ell_1, \ldots, \ell_n$.
- Make sure $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ is prime.
- Let $X = \{y^2 = x^3 + Ax^2 + x$ over $\mathbb{F}_p$ with $p+1$ points$\}$.
- Look at the ℓ_i-isogenies defined over $\mathbb{F}_p$ within X.

CSIDH in one slide

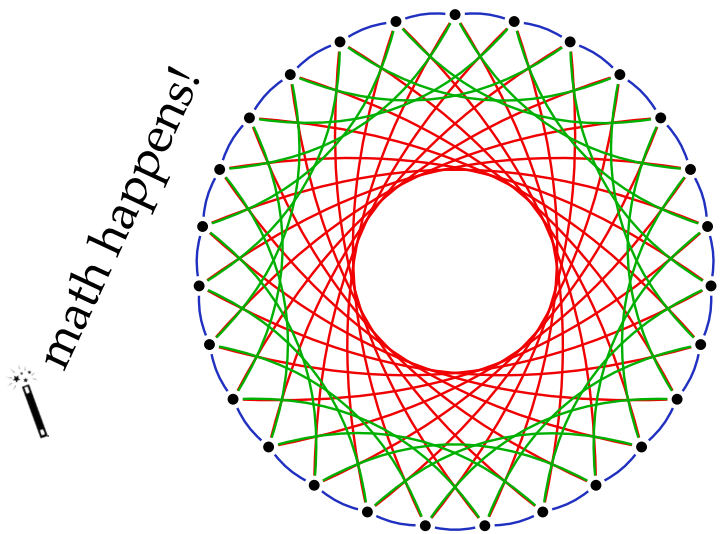
- Choose some small odd primes $\ell_1, \ldots, \ell_n$.
- Make sure $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ is prime.
- Let $X = \{y^2 = x^3 + Ax^2 + x$ over $\mathbb{F}_p$ with $p+1$ points$\}$.
- Look at the ℓ_i-isogenies defined over $\mathbb{F}_p$ within X.



$p = 419$
$\ell_1 = 3$
$\ell_2 = 5$
$\ell_3 = 7$

CSIDH in one slide

- Choose some small odd primes $\ell_1, \ldots, \ell_n$.
- Make sure $p = 4 \cdot \ell_1 \cdots \ell_n - 1$ is prime.
- Let $X = \{y^2 = x^3 + Ax^2 + x$ over $\mathbb{F}_p$ with $p+1$ points$\}$.
- Look at the ℓ_i-isogenies defined over $\mathbb{F}_p$ within X.



$p = 419$
$\ell_1 = 3$
$\ell_2 = 5$
$\ell_3 = 7$

- Walking "left" and "right" on any ℓ_i-subgraph is efficient.

Walking in the CSIDH graph

- Recall $p + 1 = 4 \cdot \ell_1 \cdots \ell_n$.
  Special p yields supersingular curves of very smooth order.
- Note $\pi^2 = -p$, so the ideals (ℓ_i) split as $(\ell_i, \pi - 1) \cdot (\ell_i, \pi + 1)$.

Walking in the CSIDH graph

- Recall $p + 1 = 4 \cdot \ell_1 \cdots \ell_n$.
  Special p yields supersingular curves of very smooth order.
- Note $\pi^2 = -p$, so the ideals (ℓ_i) split as $(\ell_i, \pi - 1) \cdot (\ell_i, \pi + 1)$.

Computing the action of $\mathfrak{l}_i = (\ell_i, \pi - 1)$:

1. Find a point $(x, y) \in E$ of order ℓ_i with $x, y \in \mathbb{F}_p$.
2. Compute the isogeny with kernel $\langle (x, y) \rangle$.

Walking in the CSIDH graph

- Recall $p + 1 = 4 \cdot \ell_1 \cdots \ell_n$.
  Special p yields supersingular curves of very smooth order.
- Note $\pi^2 = -p$, so the ideals (ℓ_i) split as $(\ell_i, \pi - 1) \cdot (\ell_i, \pi + 1)$.

Computing the action of $\mathfrak{l}_i = (\ell_i, \pi - 1)$:

1. Find a point $(x, y) \in E$ of order ℓ_i with $x, y \in \mathbb{F}_p$.
2. Compute the isogeny with kernel $\langle (x, y) \rangle$.

Computing the action of $\bar{\mathfrak{l}}_i = (\ell_i, \pi + 1)$:

1. Find a point $(x, y) \in E$ of order ℓ_i with $x \in \mathbb{F}_p$ but $y \notin \mathbb{F}_p$.
2. Compute the isogeny with kernel $\langle (x, y) \rangle$.

Walking in the CSIDH graph

- Recall $p + 1 = 4 \cdot \ell_1 \cdots \ell_n$.
  Special p yields supersingular curves of very smooth order.
- Note $\pi^2 = -p$, so the ideals (ℓ_i) split as $(\ell_i, \pi - 1) \cdot (\ell_i, \pi + 1)$.

Computing the action of $\mathfrak{l}_i = (\ell_i, \pi - 1)$:

1. Find a point $(x, y) \in E$ of order ℓ_i with $x, y \in \mathbb{F}_p$.
2. Compute the isogeny with kernel $\langle (x, y) \rangle$.

Computing the action of $\bar{\mathfrak{l}}_i = (\ell_i, \pi + 1)$:

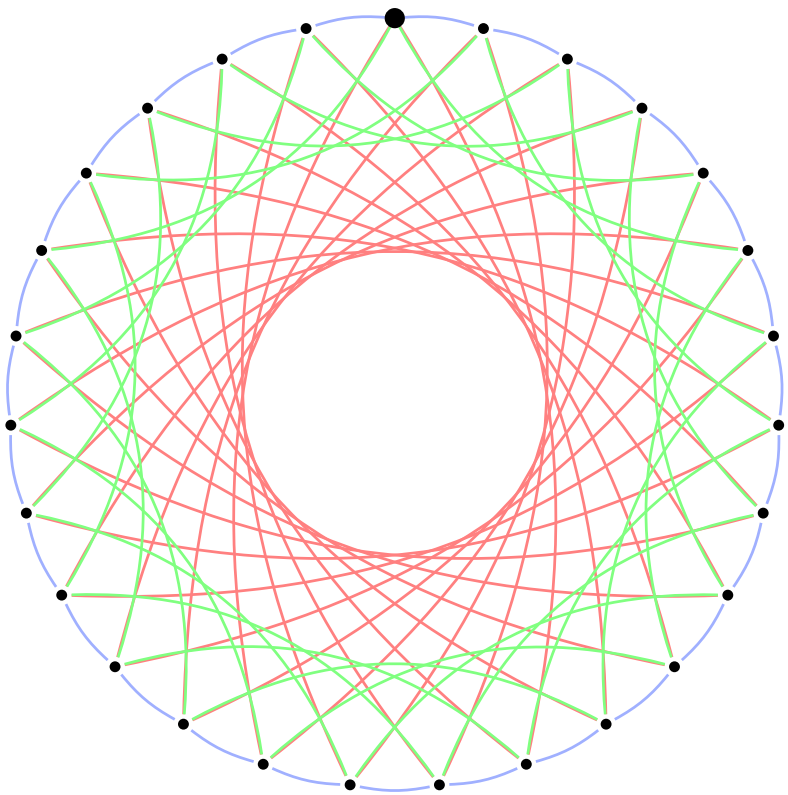
1. Find a point $(x, y) \in E$ of order ℓ_i with $x \in \mathbb{F}_p$ but $y \notin \mathbb{F}_p$.
2. Compute the isogeny with kernel $\langle (x, y) \rangle$.

Net result: With x-only arithmetic everything happens over $\mathbb{F}_p$.
$\implies$ Efficient to implement!

CSIDH key exchange

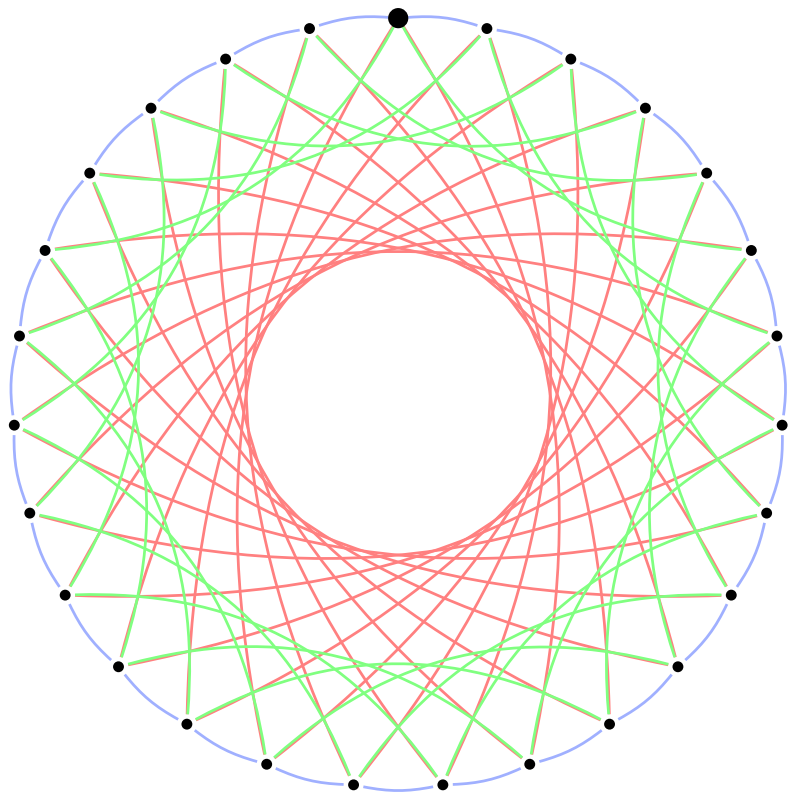
Alice

$[+, +, -, -]$



Bob

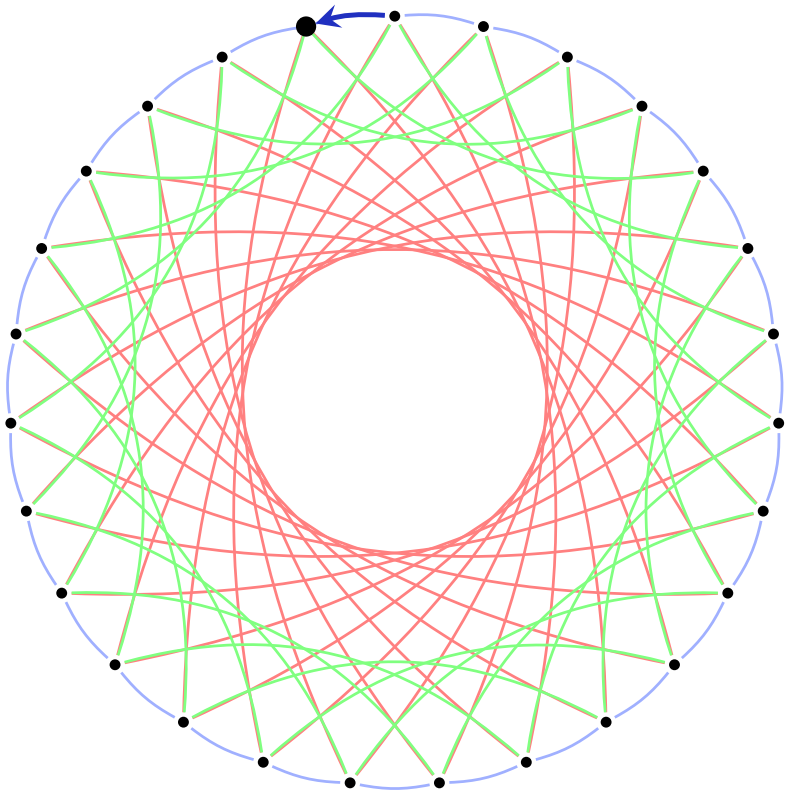
$[-, +, -, -]$



CSIDH key exchange

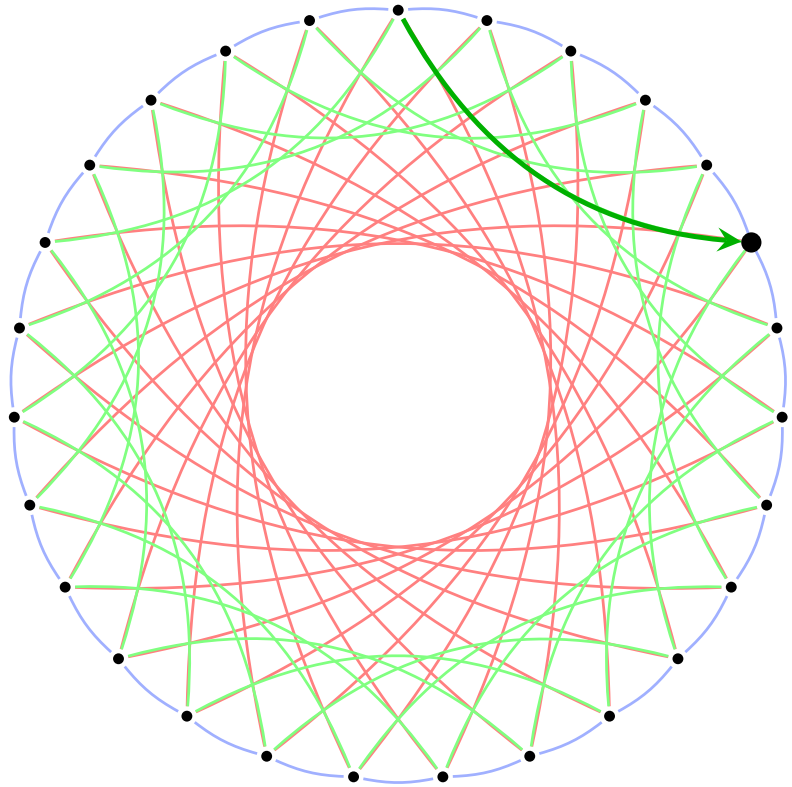
Alice

$[+, +, -, -]$



Bob

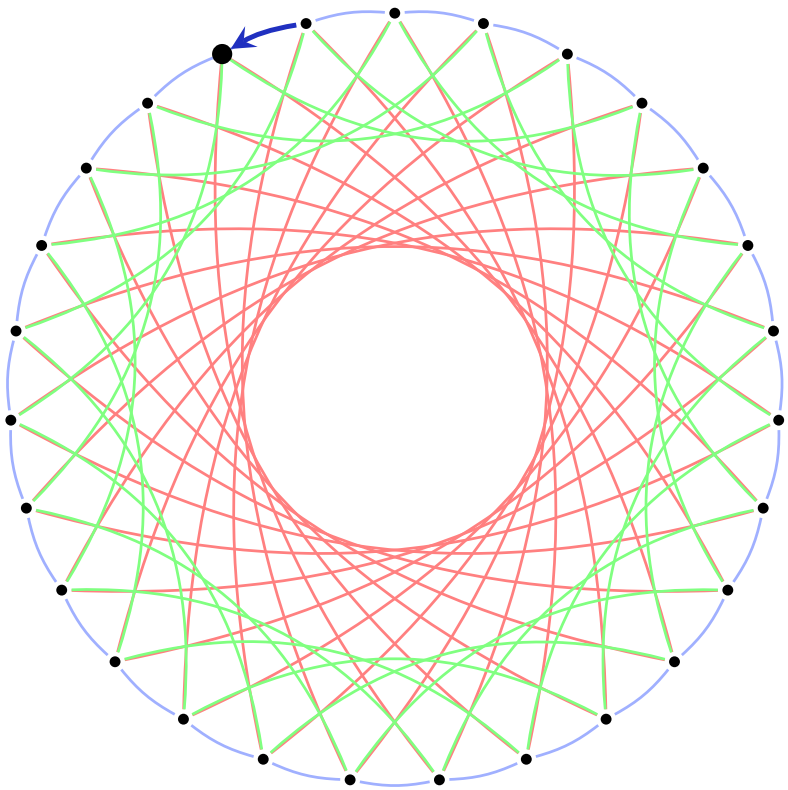
$[-, +, -, -]$



CSIDH key exchange

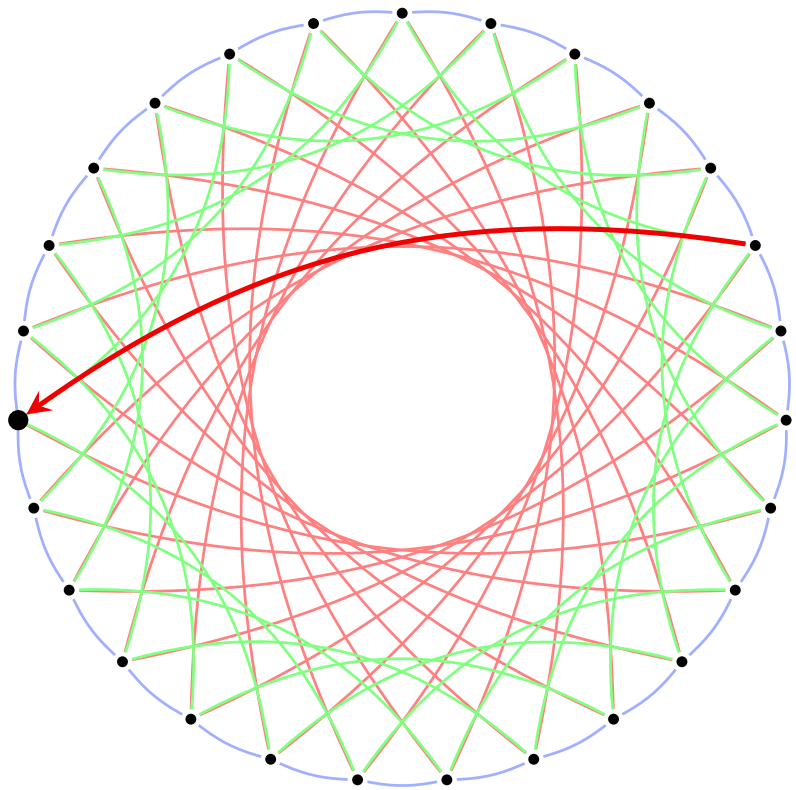
Alice

$[+, +, -, -]$



Bob

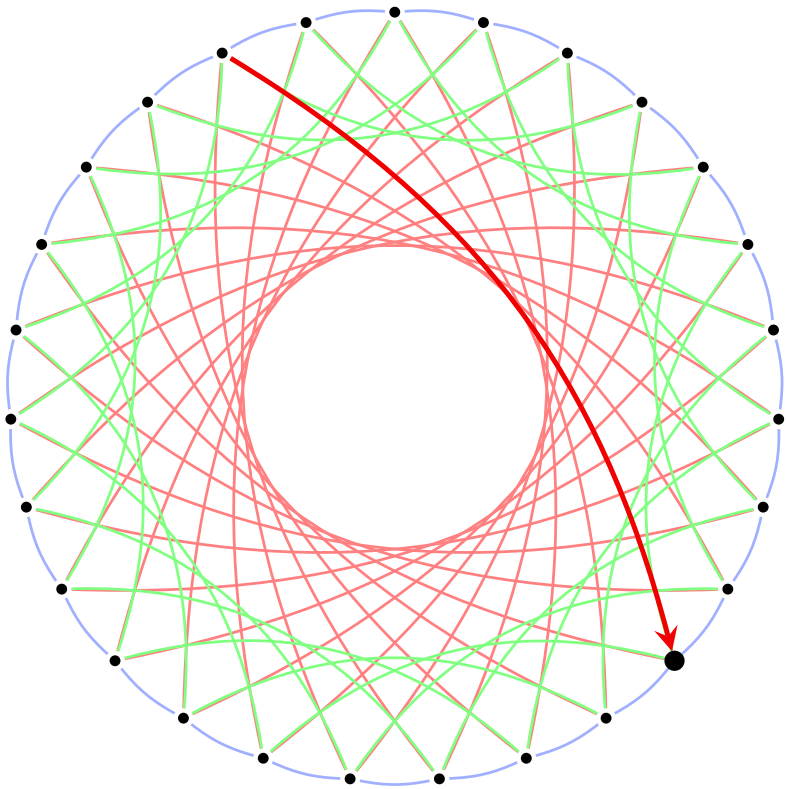
$[-, +, -, -]$



CSIDH key exchange

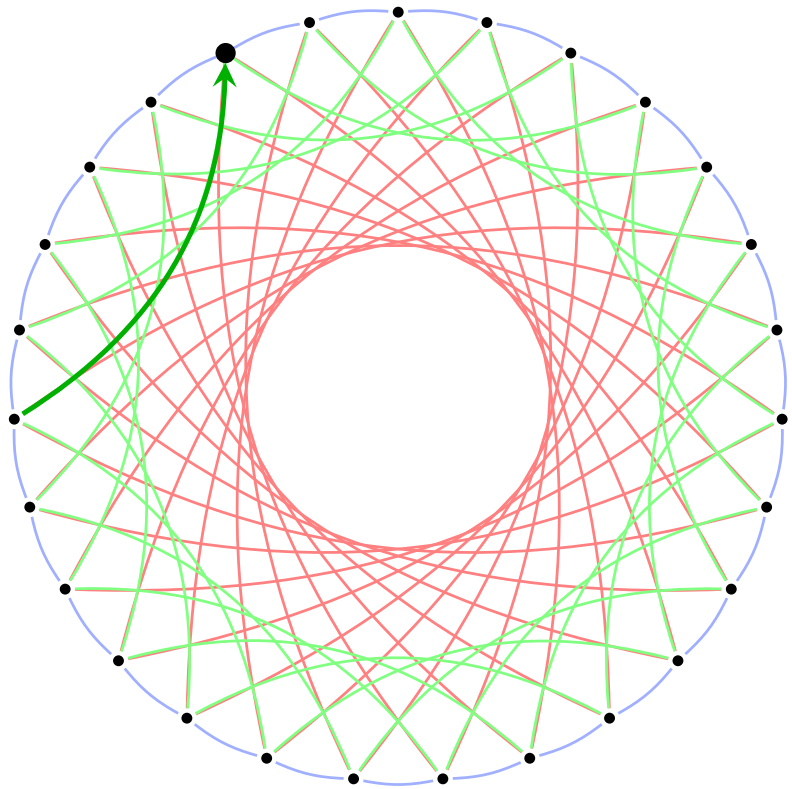
Alice

$[+, +, -, -]$



Bob

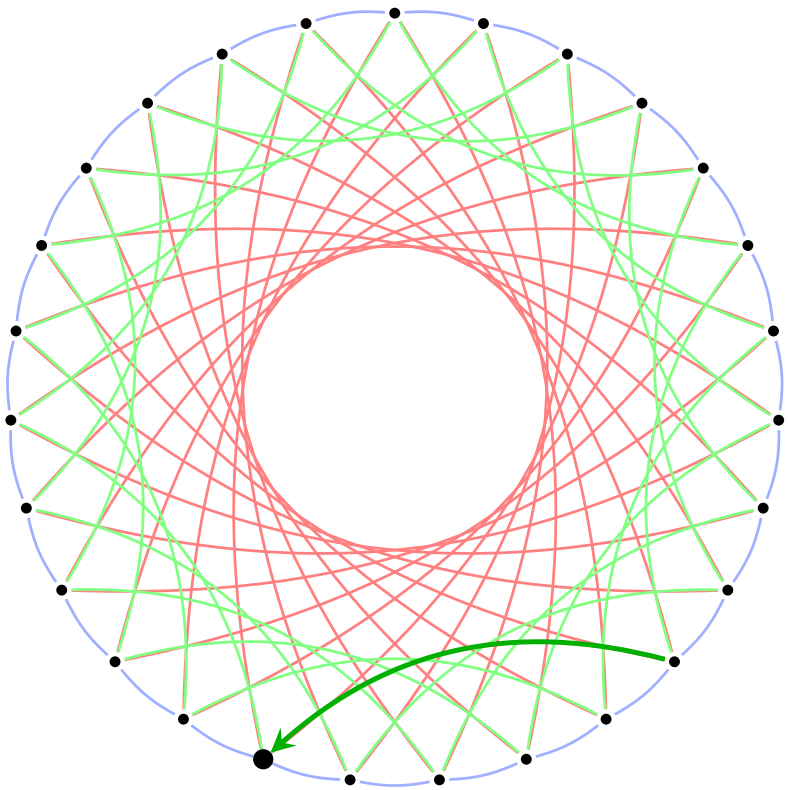
$[-, +, -, -]$



CSIDH key exchange

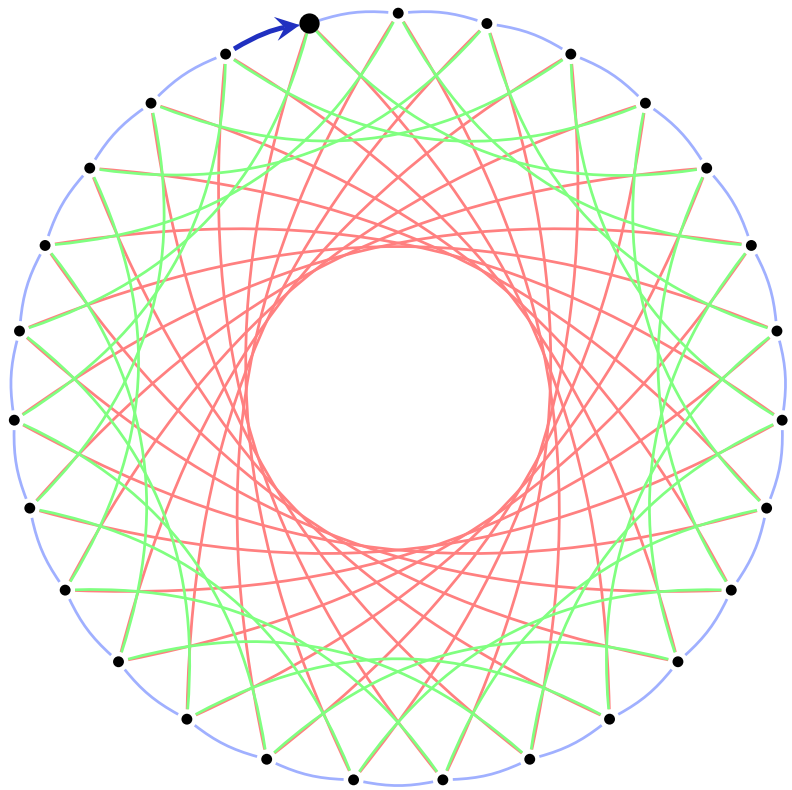
Alice

$[+, +, -, -]$



Bob

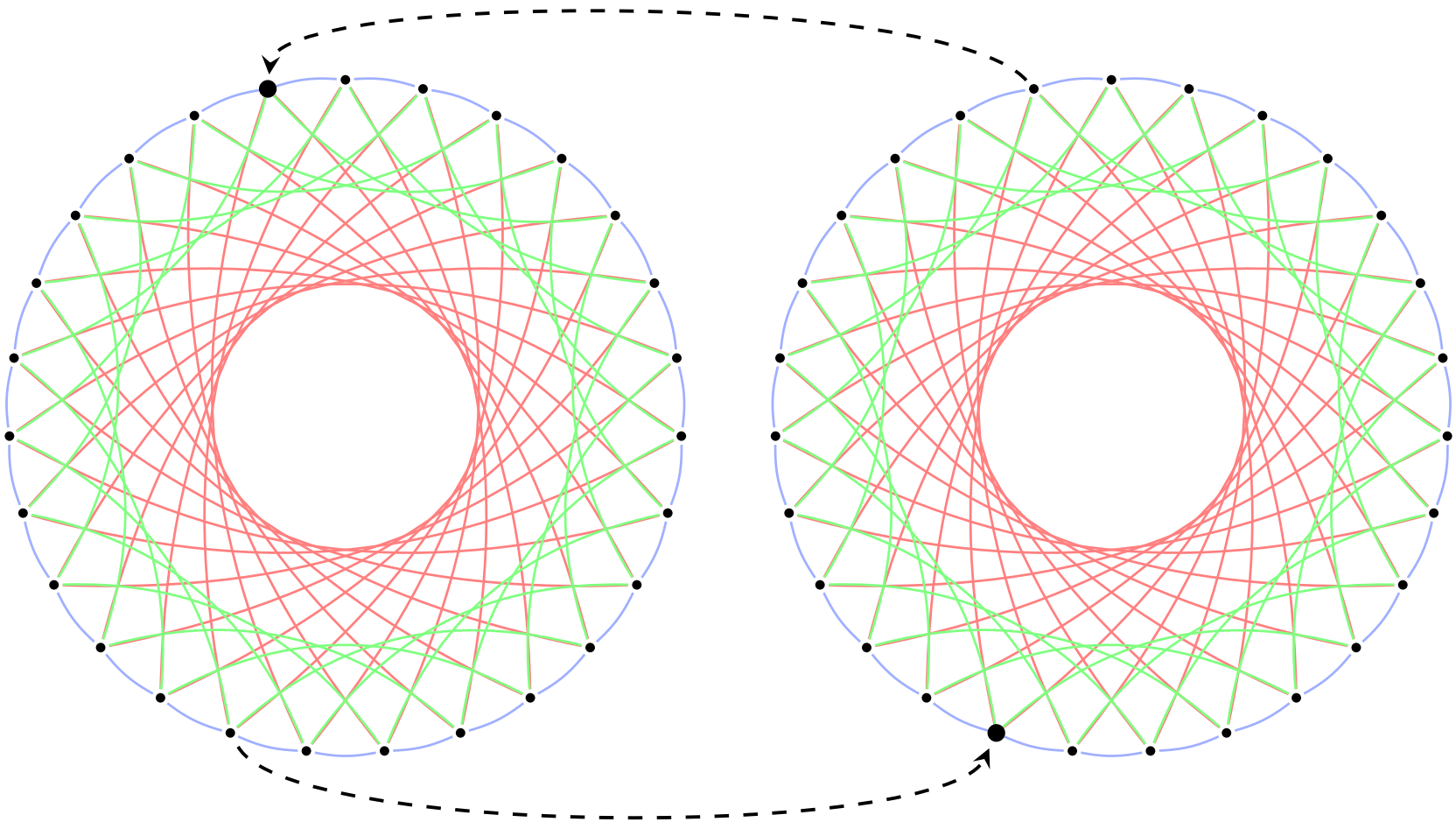
$[-, +, -, -]$



CSIDH key exchange

Alice
$[+, +, -, -]$

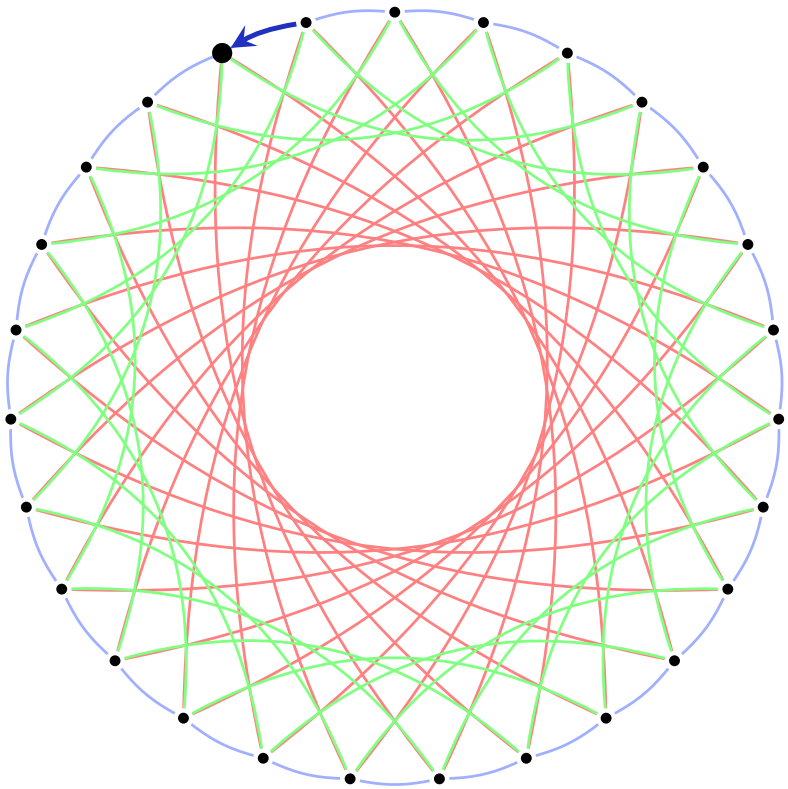
Bob
$[-, +, -, -]$



CSIDH key exchange

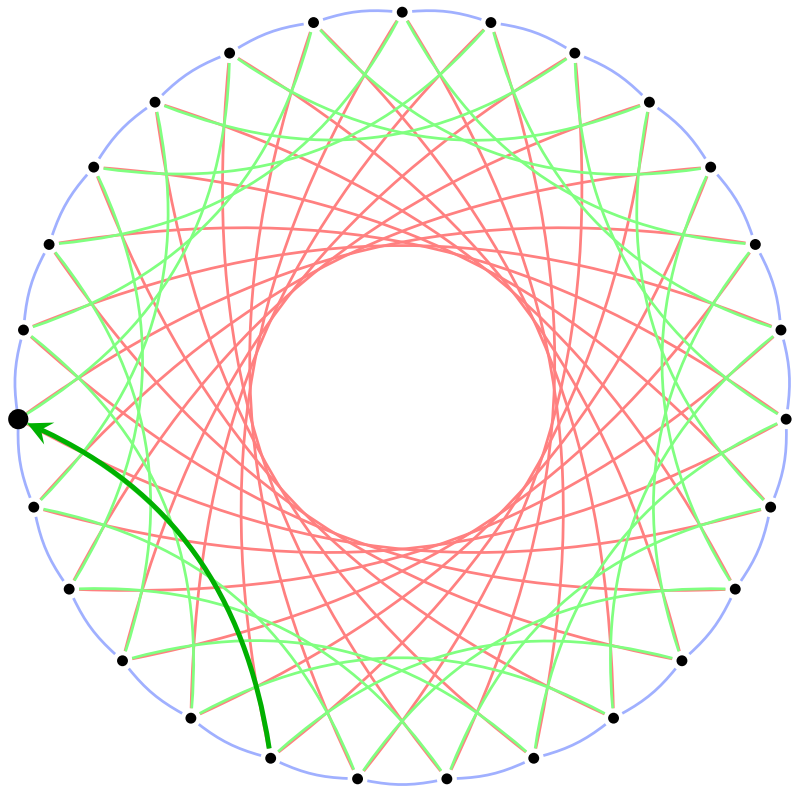
Alice

$[+, +, -, -]$



Bob

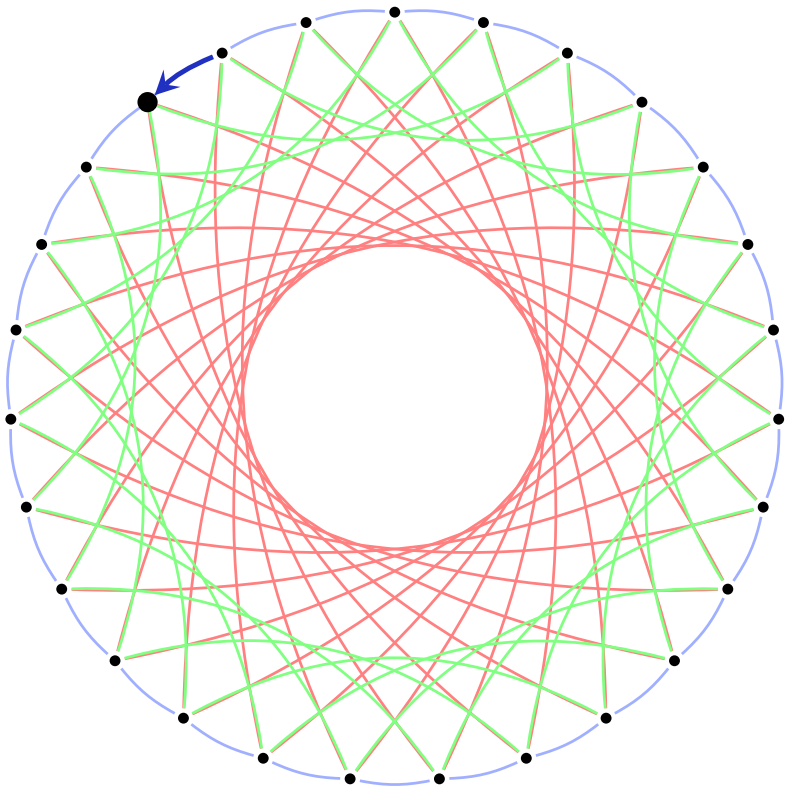
$[-, +, -, -]$



CSIDH key exchange

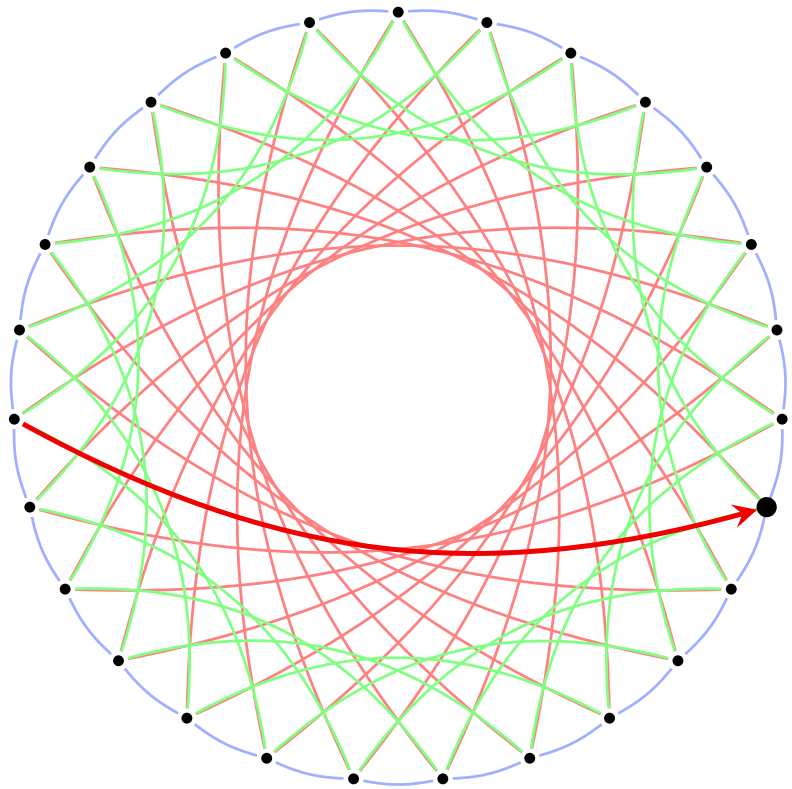
Alice

$[+, \underset{\uparrow}{+}, -, -]$



Bob

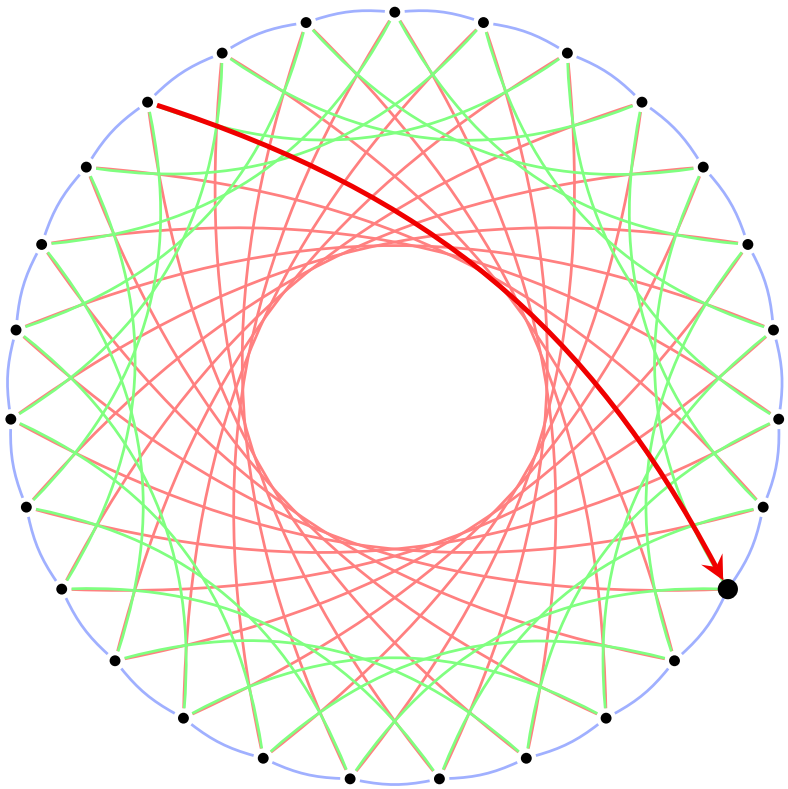
$[-, \underset{\uparrow}{+}, -, -]$



CSIDH key exchange

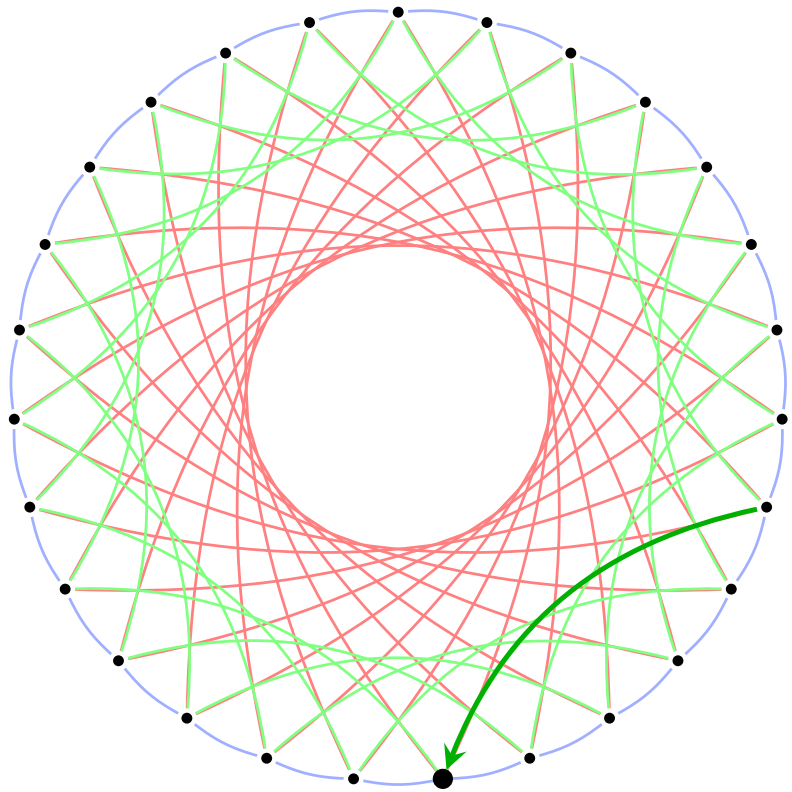
Alice

$[+, +, -, -]$



Bob

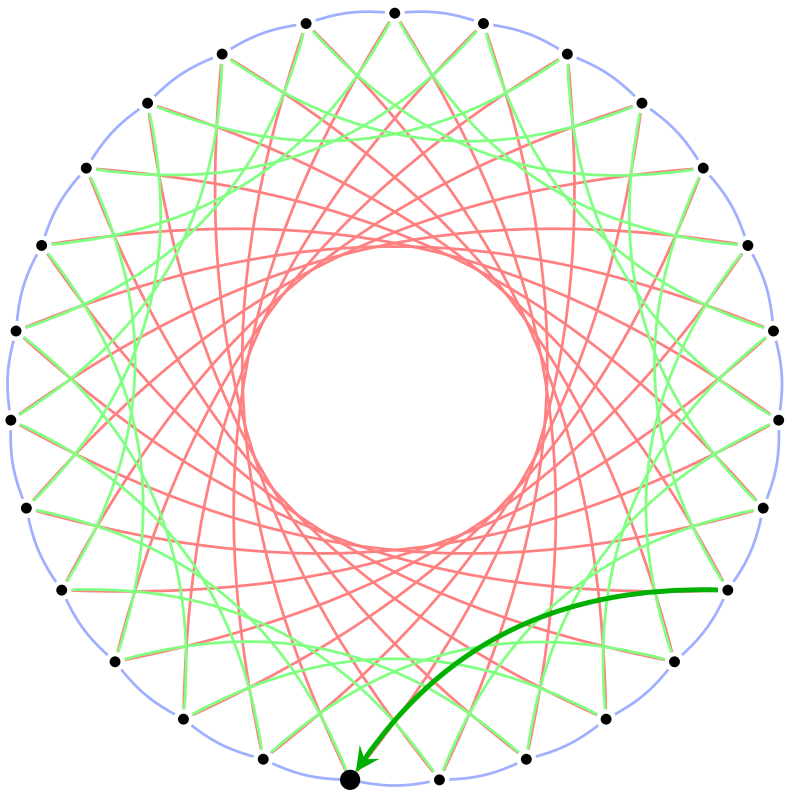
$[-, +, -, -]$



CSIDH key exchange

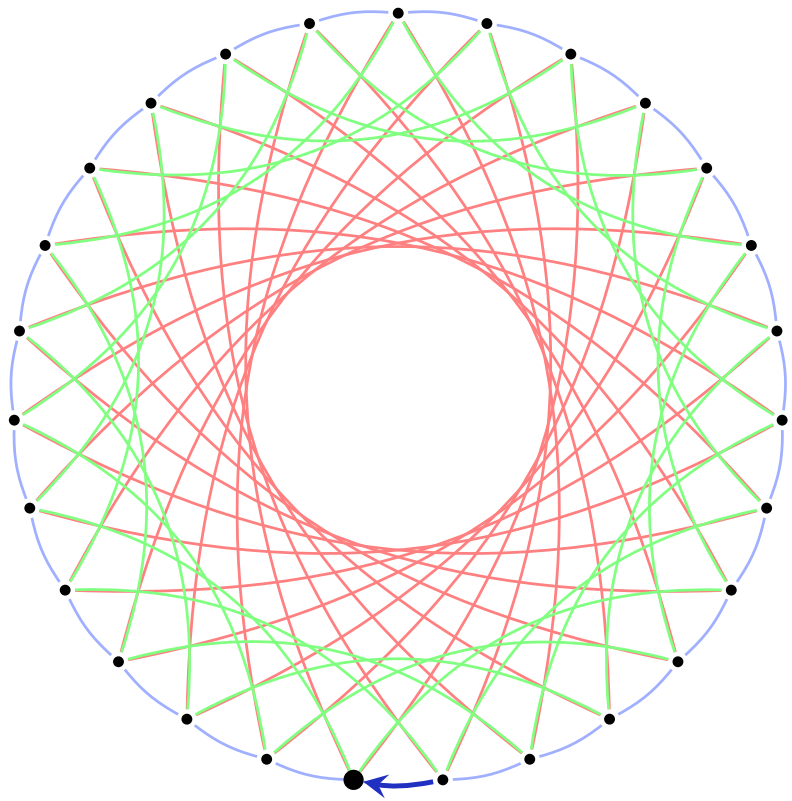
Alice

$[+, +, -, -]$



Bob

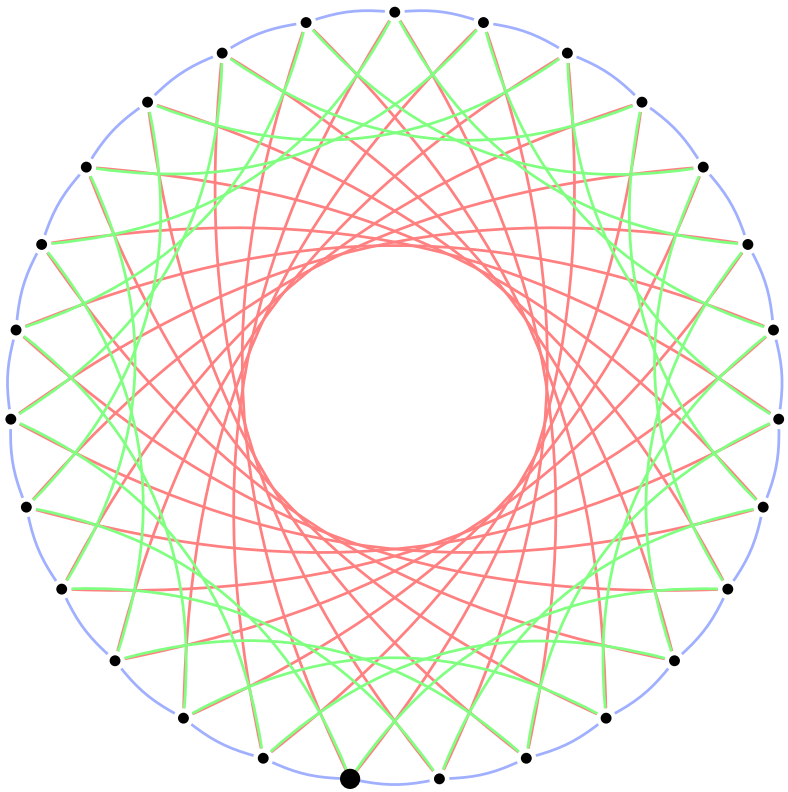
$[-, +, -, -]$



CSIDH key exchange

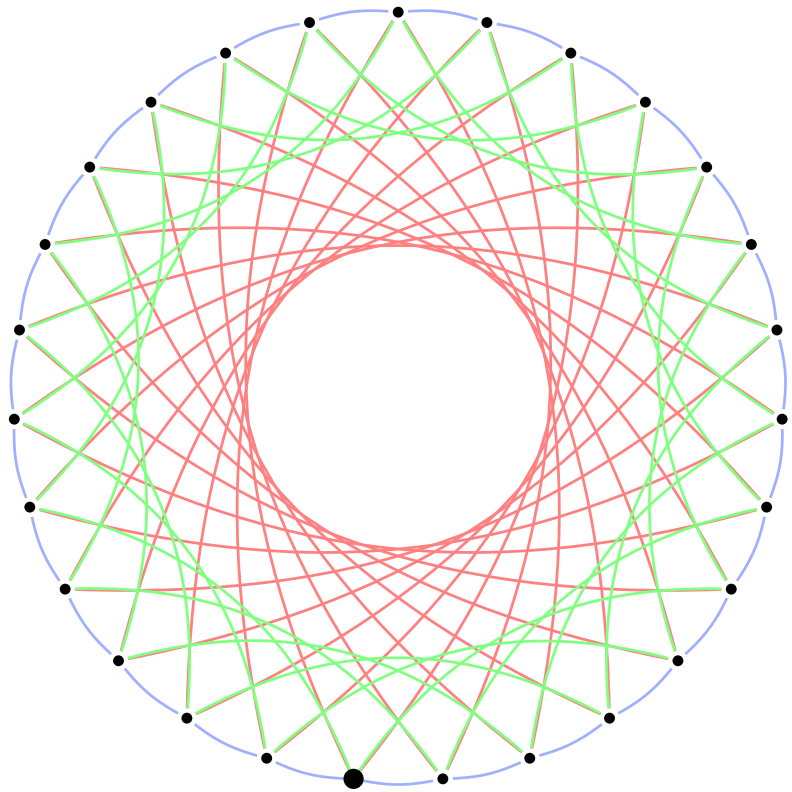
Alice

$[+, +, -, -]$



Bob

$[-, +, -, -]$



What is this magic?

- The Frobenius action "carves out" distinguished one-dimensional subgroups of the ℓ_i-torsion.

What is this magic?

- The Frobenius action "carves out" distinguished one-dimensional subgroups of the ℓ_i-torsion.
- Rational isogenies commute with π, hence the choice of these subgroups is compatible between different curves.

What is this magic?

- The Frobenius action "carves out" distinguished one-dimensional subgroups of the ℓ_i-torsion.
- Rational isogenies commute with π, hence the choice of these subgroups is compatible between different curves.
- In particular, this allows us to construct "commuting" isogenies from local information only.

What is this magic?

- The Frobenius action "carves out" distinguished one-dimensional subgroups of the ℓ_i-torsion.
- Rational isogenies commute with π, hence the choice of these subgroups is compatible between different curves.
- In particular, this allows us to construct "commuting" isogenies from local information only.
- Fun fact: There's really nothing too special about π here; it's just the one endomorphism we can always find easily.

  $\rightsquigarrow$ Generalization "OSIDH" [Colò–Kohel '19].

Why no Shor?

Shor computes α from $h = g^\alpha$ by finding the kernel of the map

$$f\colon\ \mathbb{Z}^2 \to G,\ (x, y) \mapsto g^x \cdot h^y\,.$$

Why no Shor?

Shor computes α from $h = g^\alpha$ by finding the kernel of the map

$$f\colon\ \mathbb{Z}^2 \to G,\ (x, y) \mapsto g^x \underset{\uparrow}{\cdot}\ h^y\,.$$

For group actions, we generally cannot compose $a * s$ and $b * s$!

Security of CSIDH

Core problem:
Given $E, E' \in X$, find a smooth-degree isogeny $E \to E'$.

Security of CSIDH

Core problem:
Given $E, E' \in X$, find a smooth-degree isogeny $E \to E'$.

The size of X is $\#\mathrm{cl}(\mathbb{Z}[\sqrt{-p}]) \approx \sqrt{p}$.

$\rightsquigarrow$ best known classical attack: meet-in-the-middle, $\tilde{\mathcal{O}}(p^{1/4})$.

Fully exponential: Complexity $\exp\big((\log p)^{1+o(1)}\big)$.

Security of CSIDH

Core problem:
Given $E, E' \in X$, find a smooth-degree isogeny $E \to E'$.

The size of X is $\#\mathrm{cl}(\mathbb{Z}[\sqrt{-p}]) \approx \sqrt{p}$.

$\rightsquigarrow$ best known classical attack: meet-in-the-middle, $\tilde{\mathcal{O}}(p^{1/4})$.

Fully exponential: Complexity $\exp\big((\log p)^{1+o(1)}\big)$.

Solving abelian hidden shift breaks CSIDH.

$\rightsquigarrow$ non-devastating quantum attack (Kuperberg's algorithm).

Subexponential: Complexity $\exp\big((\log p)^{1/2+o(1)}\big)$.

Can we avoid Kuperberg's algorithm?

The supersingular isogeny graph over $\mathbb{F}_{p^2}$ has less structure.

- **SIDH** uses the full $\mathbb{F}_{p^2}$-isogeny graph. No group action!

Can we avoid Kuperberg's algorithm?

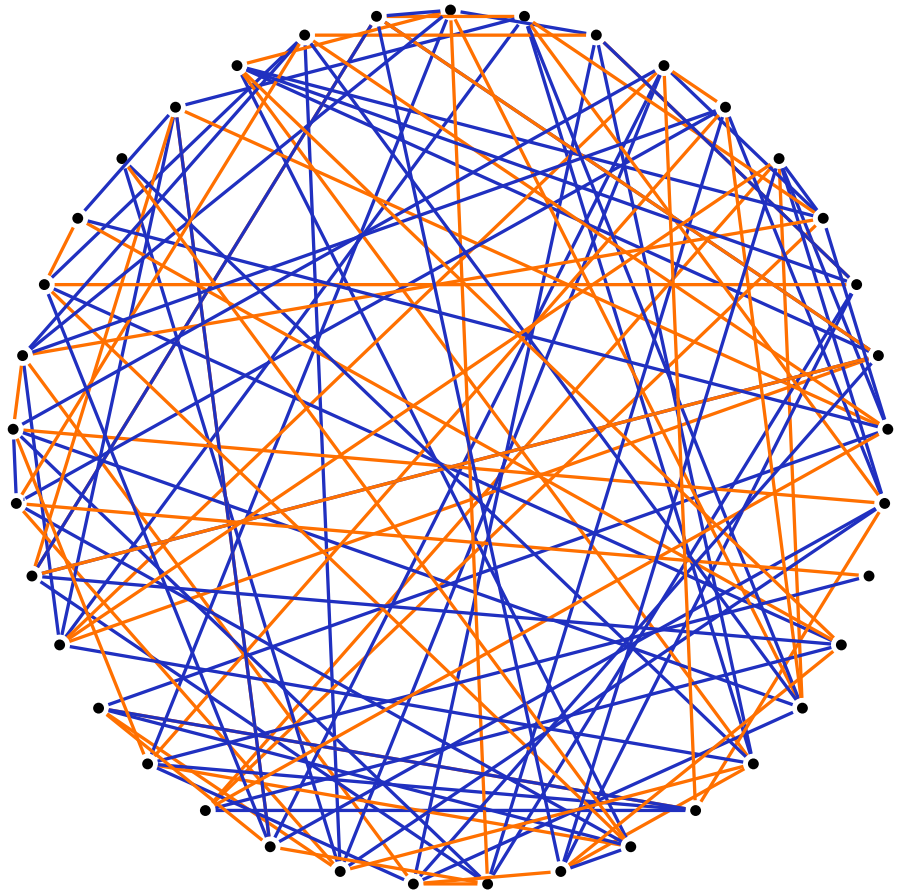
The supersingular isogeny graph over $\mathbb{F}_{p^2}$ has less structure.

- **SIDH** uses the full $\mathbb{F}_{p^2}$-isogeny graph. No group action!

- Problem: also no more intrinsic sense of direction.

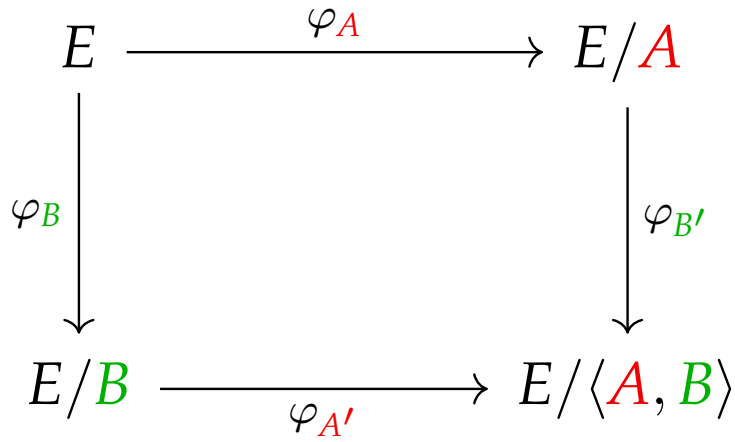
  "It all bloody looks the same!" — a famous isogeny cryptographer

$\rightsquigarrow$ need extra information to let Alice & Bob's walks commute.



Now: SIDH (Jao, De Feo; 2011)

Reminder: High-level view



- Alice & Bob pick secret subgroups A and B of E.
- Alice computes $\varphi_A : E \to E/A$; Bob computes $\varphi_B : E \to E/B$.
  (These isogenies correspond to walking on the isogeny graph.)
- Alice and Bob transmit the values E/A and E/B.
- Alice somehow obtains $A' := \text{shift\_}\varphi_B(A)$. (Similar for Bob.)
- They both compute the shared secret
  $$(E/B)/A' \cong E/\langle A, B\rangle \cong (E/A)/B'.$$

SIDH's auxiliary points

"Alice somehow obtains $A' := \text{shift\_}\varphi_B(A)$."

...but Alice knows only A, Bob knows only φ_B. Hm.

CSIDH's solution: use distinguished subgroups (eigenspaces of π).

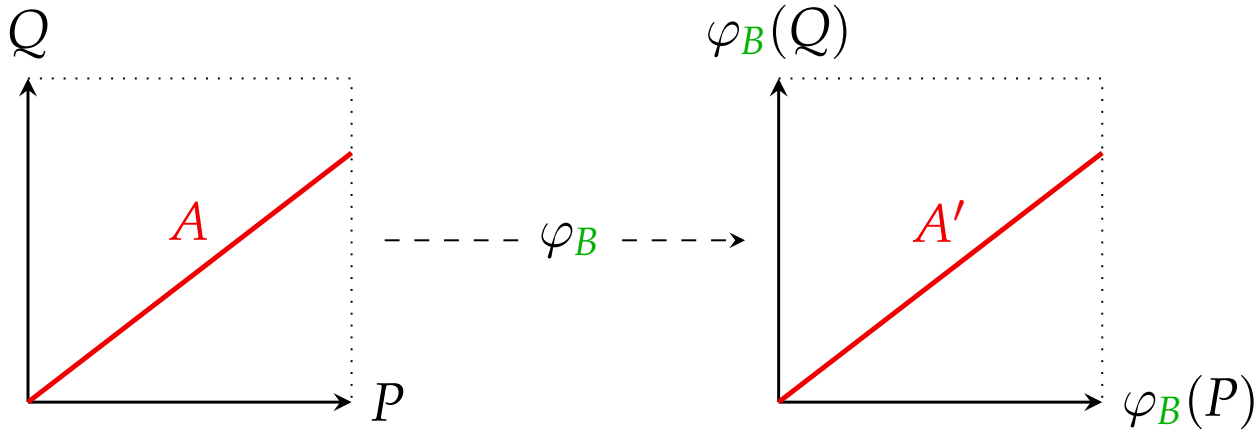
SIDH's auxiliary points

"Alice somehow obtains $A' := \text{shift\_}\varphi_B(A)$."

...but Alice knows only A, Bob knows only φ_B. Hm.

CSIDH's solution: use distinguished subgroups (eigenspaces of π).

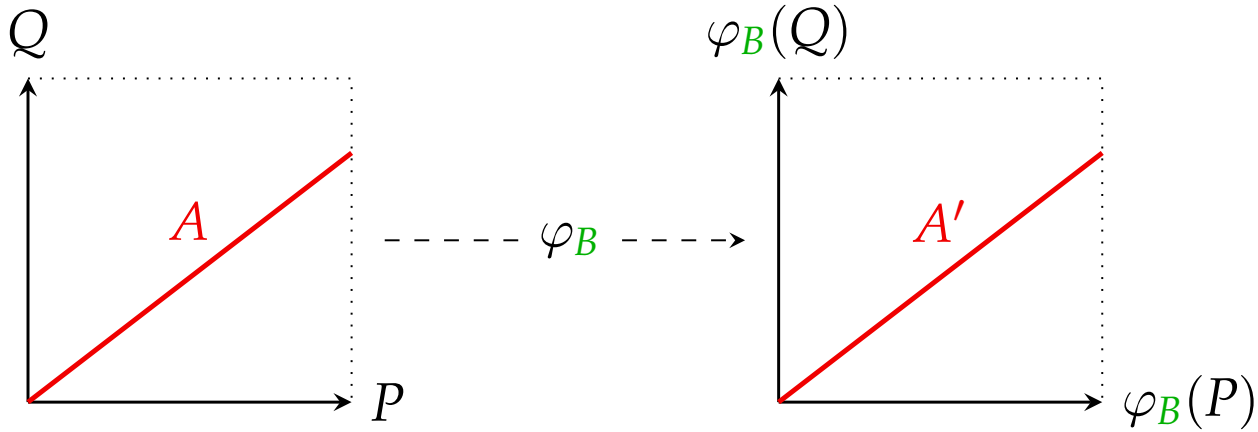
SIDH's solution: φ_B is a group homomorphism!



SIDH's auxiliary points

"Alice somehow obtains $A' := \text{shift\_}\varphi_B(A)$."
...but Alice knows only A, Bob knows only φ_B. Hm.
CSIDH's solution: use distinguished subgroups (eigenspaces of π).

SIDH's solution: φ_B is a group homomorphism! (and $A \cap B = \{\infty\}$)



- Alice picks A as $\langle P + [a]Q \rangle$ for fixed public $P, Q \in E$.
- Bob includes $\varphi_B(P)$ and $\varphi_B(Q)$ in his public key.

$\Longrightarrow$ Now Alice can compute A' as $\langle \varphi_B(P) + [a]\varphi_B(Q) \rangle$.

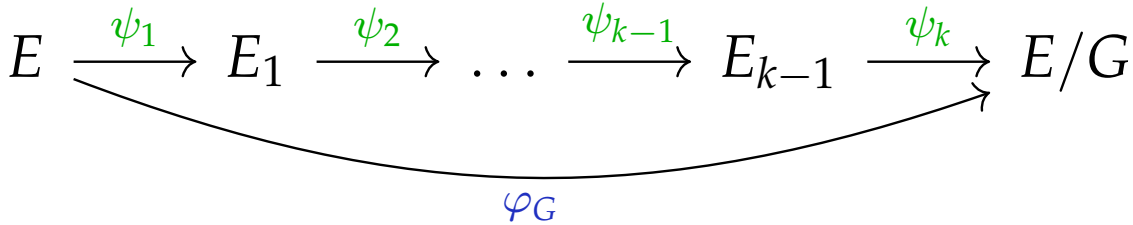
Computational aspects of SIDH

- In SIDH, $\#A$ and $\#B$ are "crypto-sized".
  Vélu's formulas take $\Theta(\#G)$ to compute $\varphi_G \colon E \to E/G$.

Computational aspects of SIDH

- In SIDH, $\#A = 2^n$ and $\#B = 3^m$ are "crypto-sized".
  Vélu's formulas take $\Theta(\#G)$ to compute $\varphi_G \colon E \to E/G$.

!! Evaluate φ_G as a chain of small-degree isogenies:
For $G \cong \mathbb{Z}/\ell^k$, set $\ker \psi_i := [\ell^{k-i}](\psi_{i-1} \circ \cdots \circ \psi_1)(G)$.



Computational aspects of SIDH

- In SIDH, $\#A = 2^n$ and $\#B = 3^m$ are "crypto-sized".
  Vélu's formulas take $\Theta(\#G)$ to compute $\varphi_G \colon E \to E/G$.

!! Evaluate φ_G as a chain of small-degree isogenies:
For $G \cong \mathbb{Z}/\ell^k$, set $\ker \psi_i := [\ell^{k-i}](\psi_{i-1} \circ \cdots \circ \psi_1)(G)$.

$$E \xrightarrow{\psi_1} E_1 \xrightarrow{\psi_2} \dots \xrightarrow{\psi_{k-1}} E_{k-1} \xrightarrow{\psi_k} E/G$$
$$\varphi_G$$

$\rightsquigarrow$ Complexity: $O(k^2 \cdot \ell)$. Exponentially smaller than ℓ^k!
"Optimal strategy" improves this to $O(k \log k \cdot \ell)$.

Computational aspects of SIDH

- In SIDH, $\#A = 2^n$ and $\#B = 3^m$ are "crypto-sized".
  Vélu's formulas take $\Theta(\#G)$ to compute $\varphi_G \colon E \to E/G$.

!! Evaluate φ_G as a chain of small-degree isogenies:
For $G \cong \mathbb{Z}/\ell^k$, set $\ker \psi_i := [\ell^{k-i}](\psi_{i-1} \circ \cdots \circ \psi_1)(G)$.

$$E \xrightarrow{\psi_1} E_1 \xrightarrow{\psi_2} \ldots \xrightarrow{\psi_{k-1}} E_{k-1} \xrightarrow{\psi_k} E/G$$
$$\varphi_G$$

$\rightsquigarrow$ Complexity: $O(k^2 \cdot \ell)$. Exponentially smaller than ℓ^k!
"Optimal strategy" improves this to $O(k \log k \cdot \ell)$.

- Also choose special p such that everything stays over $\mathbb{F}_{p^2}$.

SIDH in one slide

Public parameters:

- a large prime $p = 2^n 3^m - 1$ and a supersingular $E/\mathbb{F}_p$
- bases (P, Q) and (R, S) of $E[2^n]$ and $E[3^m]$ (recall $E[k] \cong \mathbb{Z}/k \times \mathbb{Z}/k$)

Alice	public	Bob
$a \xleftarrow{\text{random}} \{0 \dots 2^n - 1\}$		$b \xleftarrow{\text{random}} \{0 \dots 3^m - 1\}$
$A := \langle P + [a]Q \rangle$		$B := \langle R + [b]S \rangle$
compute $\varphi_A : E \to E/A$		compute $\varphi_B : E \to E/B$
$E/A,\ \varphi_A(R),\ \varphi_A(S)$	$\longrightarrow$	
	$\longleftarrow$	$E/B,\ \varphi_B(P),\ \varphi_B(Q)$
$A' := \langle \varphi_B(P) + [a]\varphi_B(Q) \rangle$		$B' := \langle \varphi_A(R) + [b]\varphi_A(S) \rangle$
$s := j\big((E/B)/A'\big)$		$s := j\big((E/A)/B'\big)$

Security of SIDH

The SIDH graph has size $\lfloor p/12 \rfloor + \varepsilon$.
Alice & Bob can choose from about $\sqrt{p}$ secret keys each.

Security of SIDH

The SIDH graph has size $\lfloor p/12 \rfloor + \varepsilon$.
Alice & Bob can choose from about $\sqrt{p}$ secret keys each.

Classical attacks:

- Meet-in-the-middle: $\tilde{\mathcal{O}}(p^{1/4})$ time *& space*.
- Collision finding: $\tilde{\mathcal{O}}(p^{3/8}/\sqrt{memory}/cores)$.

Security of SIDH

The SIDH graph has size $\lfloor p/12 \rfloor + \varepsilon$.
Alice & Bob can choose from about $\sqrt{p}$ secret keys each.

Classical attacks:

- Meet-in-the-middle: $\tilde{\mathcal{O}}(p^{1/4})$ time *& space*.
- Collision finding: $\tilde{\mathcal{O}}(p^{3/8}/\sqrt{memory}/cores)$.

Quantum attacks:

- Claw finding: claimed $\tilde{\mathcal{O}}(p^{1/6})$.
  [JS19] says this is *more expensive than classical attacks*.

Security of SIDH

The SIDH graph has size $\lfloor p/12 \rfloor + \varepsilon$.
Alice & Bob can choose from about $\sqrt{p}$ secret keys each.

Classical attacks:

- Meet-in-the-middle: $\tilde{\mathcal{O}}(p^{1/4})$ time *& space*.
- Collision finding: $\tilde{\mathcal{O}}(p^{3/8}/\sqrt{\mathit{memory}}/\mathit{cores})$.

Quantum attacks:

- Claw finding: claimed $\tilde{\mathcal{O}}(p^{1/6})$.
  [JS19] says this is *more expensive than classical attacks*.

Bottom line: Fully exponential. Complexity $\exp\big((\log p)^{1+o(1)}\big)$.

Questions?