CSIDH:

An Efficient Post-Quantum Commutative Group Action

Wouter Castryck[1] Tanja Lange[2] Chloe Martindale[2]
Lorenz Panny[2] Joost Renes[3]

[1]KU Leuven [2]TU Eindhoven [3]Radboud Universiteit

Hilversum, 20 March 2019

Netherlands Organisation
for Scientific Research

??? math words

I guess it's fast?

CSIDH:
An Efficient Post-Quantum Commutative Group Action

Wouter Castryck[1] Tanja Lange[2] Chloe Martindale[2]

Lorenz Panny[2] Joost Renes[3]

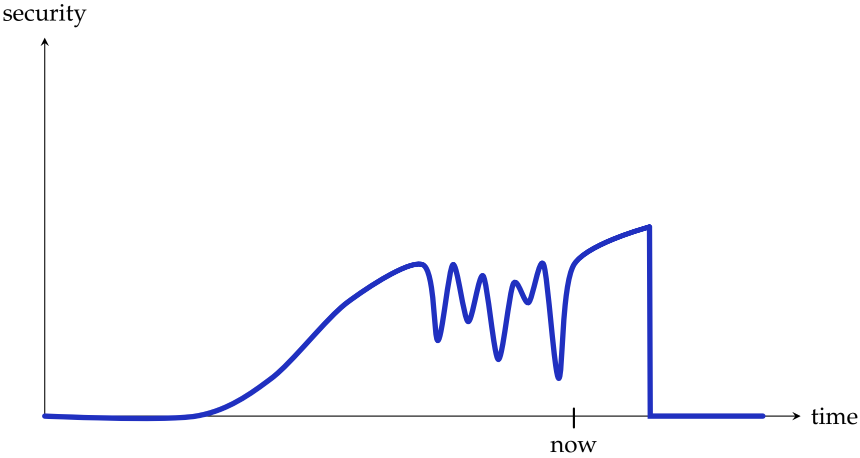
[1]KU Leuven [2]TU Eindhoven [3]Radboud Universiteit

???

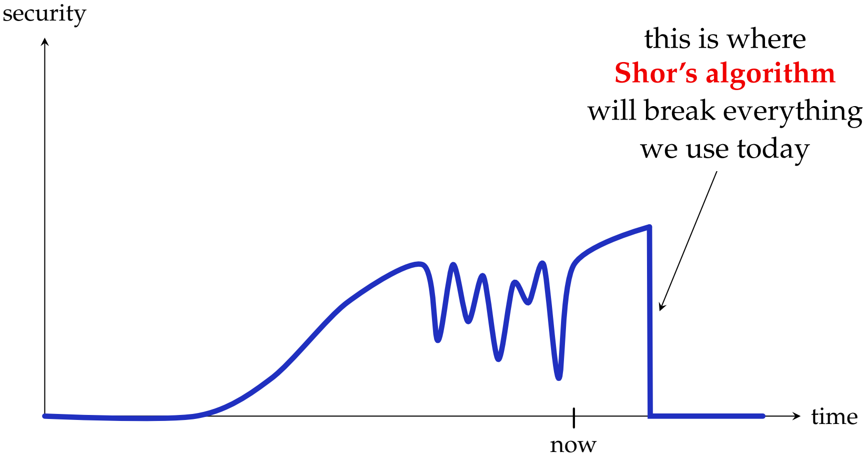
Hilversum, 20 March 2019

[ˈsiːˌsaɪd]

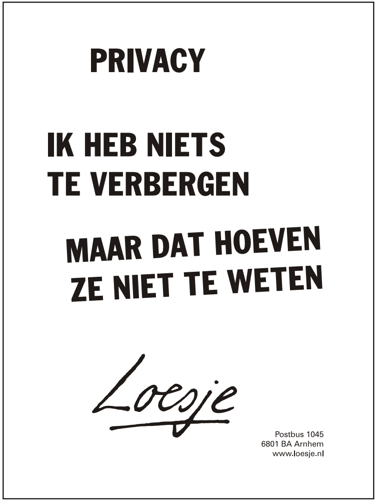
Timeline of internet security (not to scale)



Timeline of internet security (not to scale)



this is where **Shor's algorithm** will break everything we use today



PRIVACY

IK HEB NIETS
TE VERBERGEN

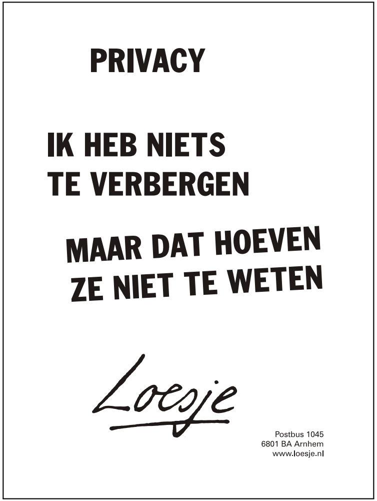
MAAR DAT HOEVEN
ZE NIET TE WETEN

Loesje

Postbus 1045
6801 BA Arnhem
www.loesje.nl

Figure 1: A brief introduction to privacy.

Quantum attacks



PRIVACY

IK HEB NIETS
TE VERBERGEN

MAAR DAT HOEVEN
ZE NIET TE WETEN

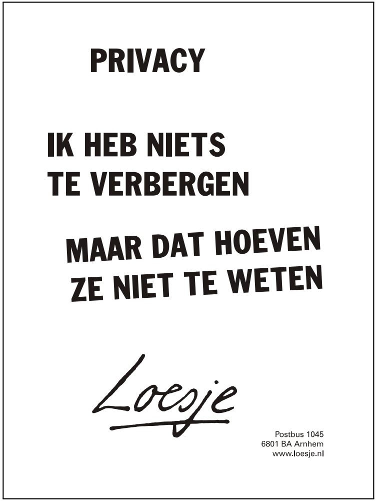
Loesje

Postbus 1045
6801 BA Arnhem
www.loesje.nl

Figure 1: A brief introduction to privacy.

- Quantum computers will break all common public-key crypto.

Quantum attacks



PRIVACY

IK HEB NIETS
TE VERBERGEN

MAAR DAT HOEVEN
ZE NIET TE WETEN

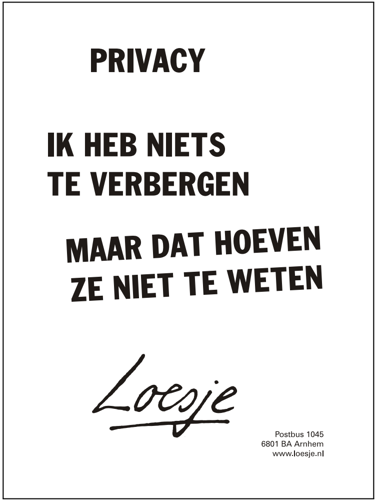
Loesje

Postbus 1045
6801 BA Arnhem
www.loesje.nl

Figure 1: A brief introduction to privacy.

- Quantum computers will break all common public-key crypto.
- The good news: Nobody has a big enough quantum computer yet.

Quantum attacks



PRIVACY

IK HEB NIETS
TE VERBERGEN

MAAR DAT HOEVEN
ZE NIET TE WETEN

Loesje

Postbus 1045
6801 BA Arnhem
www.loesje.nl

Figure 1: A brief introduction to privacy.

- Quantum computers will break all common public-key crypto.
- The good news: Nobody has a big enough quantum computer yet.
- The bad news: Attackers run a massive collect-now-decrypt-later effort.
  - Havoc will break loose once they can decipher important secrets...

Shor's algorithm ('94)

Figure 2: Peter W. Shor attacking the crypto in TLS.

Shor's algorithm ('94)

Figure 2: Peter W. Shor attacking the crypto in TLS, and an actual picture of him.

Is all hope lost?

Crypto is probably going to be fine — **if** we act now(-ish).

Is all hope lost?

Crypto is probably going to be fine — **if** we act now(-ish).

- Common misconception:

  "Quantum computers can do *everything* super fast."
- **Not true!** Many computations have little or no known quantum speedups.

Is all hope lost?

Crypto is probably going to be fine — **if** we act now(-ish).

- Common misconception:

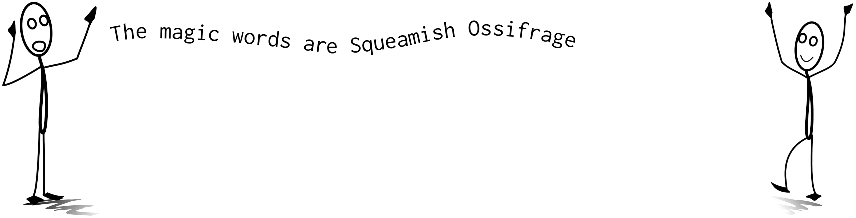
  “Quantum computers can do *everything* super fast.”
- **Not true!** Many computations have little or no known quantum speedups.

Post-quantum cryptography

uses computational problems where no devastating quantum attacks are known.

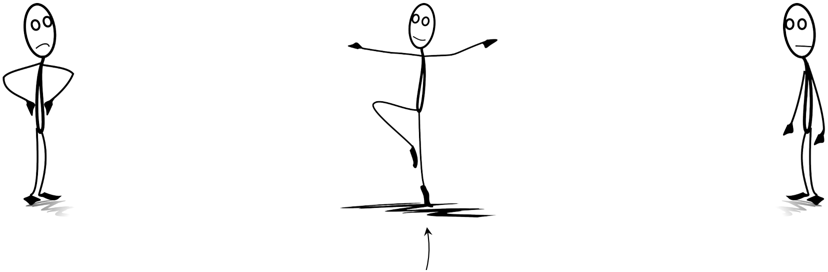
The Diffie–Hellman key exchange

Suppose Alice and Bob want to share a secret.



The Diffie–Hellman key exchange

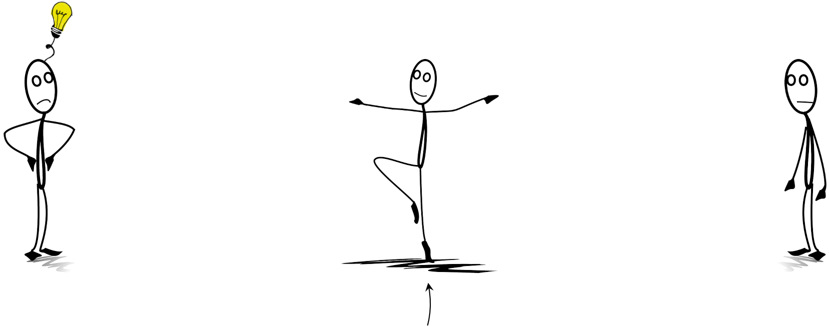
Suppose Alice and Bob want to share a secret.



evil eavesdropper Eve!

The Diffie–Hellman key exchange

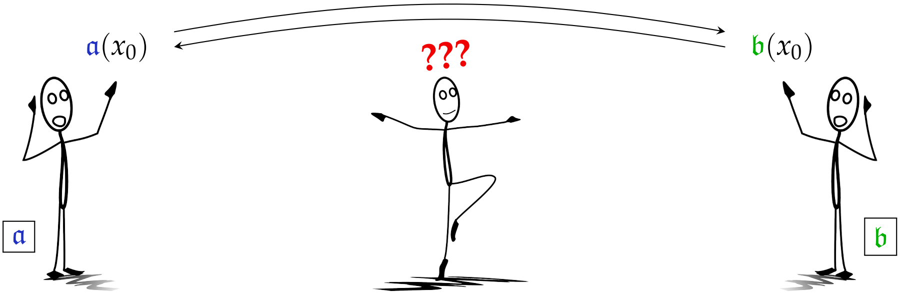
Suppose Alice and Bob want to share a secret.



evil eavesdropper Eve!

The Diffie–Hellman key exchange

Suppose Alice and Bob want to share a secret.



$\mathfrak{a}(x_0)$

???

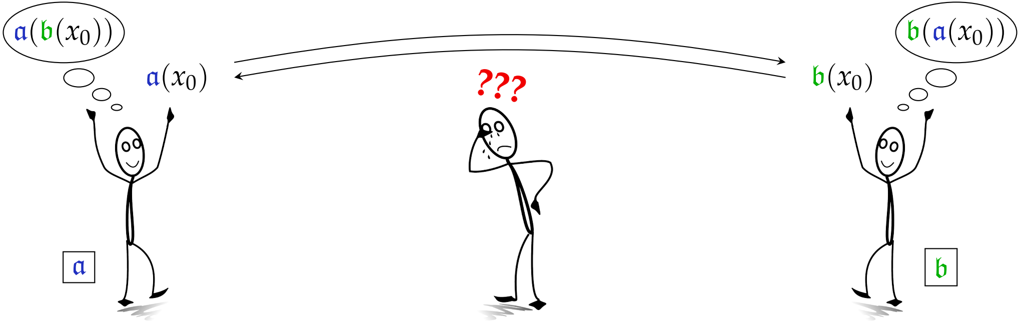
$\mathfrak{b}(x_0)$

$\boxed{\mathfrak{a}}$

$\boxed{\mathfrak{b}}$

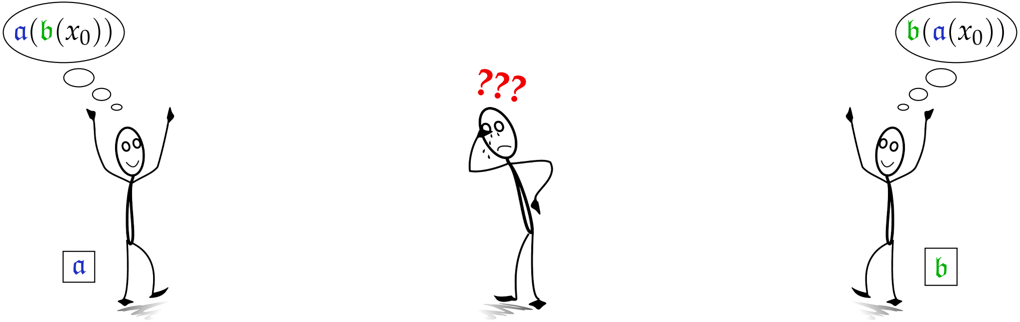
The Diffie–Hellman key exchange

Suppose Alice and Bob want to share a secret.



The Diffie–Hellman key exchange

Suppose Alice and Bob want to share a secret.


$\mathfrak{a}(\mathfrak{b}(x_0))$
???
$\mathfrak{b}(\mathfrak{a}(x_0))$
$\mathfrak{a}$
$\mathfrak{b}$

- By ~~magic~~ math, $\mathfrak{a}(\mathfrak{b}(x_0)) = \mathfrak{b}(\mathfrak{a}(x_0))$! ...but Eve doesn't know this secret.
- Now Alice and Bob can use their secret to encrypt messages back and forth.

Non-interactive key exchange

The method on the previous slide is an example of a non-interactive key exchange:

Non-interactive key exchange

The method on the previous slide is an example of a non-interactive key exchange:

Everything sent by Alice and Bob is independent of who they are talking to!
They can simply make $\mathfrak{a}(x_0)$ and $\mathfrak{b}(x_0)$ at *some* point in time and publish them.

Non-interactive key exchange

The method on the previous slide is an example of a non-interactive key exchange:

Everything sent by Alice and Bob is independent of who they are talking to!
They can simply make $\mathfrak{a}(x_0)$ and $\mathfrak{b}(x_0)$ at *some* point in time and publish them.

Alice can obtain a *shared secret* by applying her secret $\mathfrak{a}$ to Bob's public key $\mathfrak{b}(x_0)$, and vice-versa. No interaction required after the initial key generation!

Our work: a post-quantum NIKE

Short summary before our work:

All NIKEs either broken by quantum computers or extremely inefficient.

Our work: a post-quantum NIKE

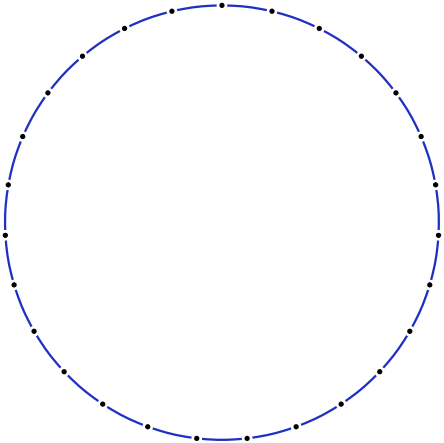
Short summary before our work:

All NIKEs either broken by quantum computers or extremely inefficient.

Short summary now:

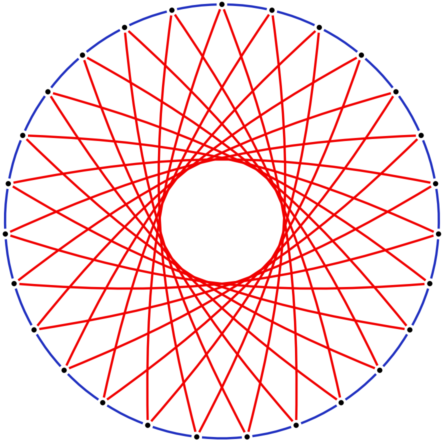
CSIDH seems post-quantum secure and is reasonably fast!

Our work: the CSIDH graph



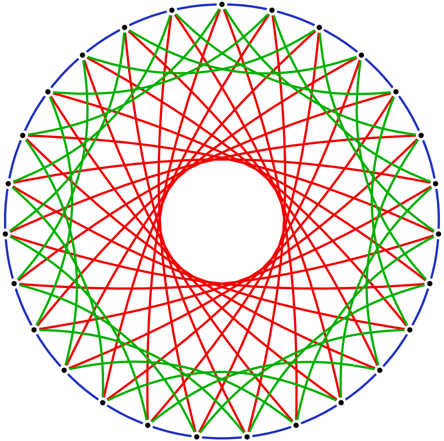
(this is just a tiny toy example)

Our work: the CSIDH graph



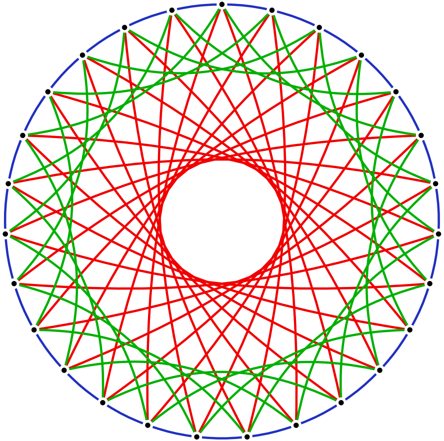
(this is just a tiny toy example)

Our work: the CSIDH graph



(this is just a tiny toy example)

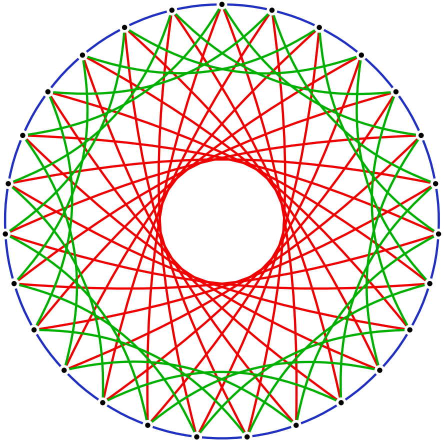
Our work: the CSIDH graph



(this is just a tiny toy example)

- You can 'walk' on this graph:
  right, left, left, left, right, left, right.

Our work: the CSIDH graph

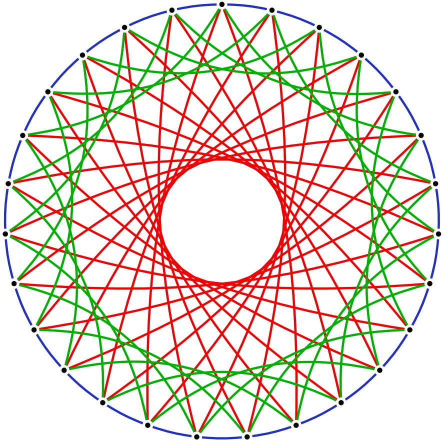


(this is just a tiny toy example)

- You can 'walk' on this graph:
  right, left, left, left, right, left, right.

- The cyclic subgraphs are compatible:
  Only the number (not the order) of steps on each color matters for where you land.

Our work: the CSIDH graph



(this is just a tiny toy example)

- You can ‘walk’ on this graph:
  right, left, left, left, right, left, right.

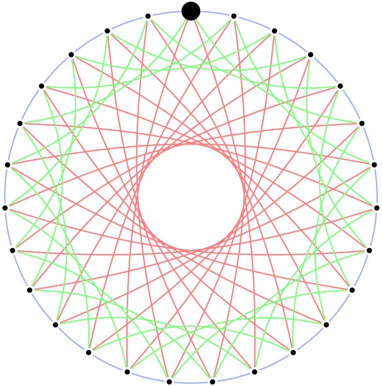
- The cyclic subgraphs are compatible:
  Only the number (not the order) of steps on each color matters for where you land.

- Alice and Bob can make a key exchange by choosing directions as their secrets $\mathfrak{a}$ and $\mathfrak{b}$ and publishing the end points of walking from a common starting node x_0.

Our work: key exchange on the CSIDH graph

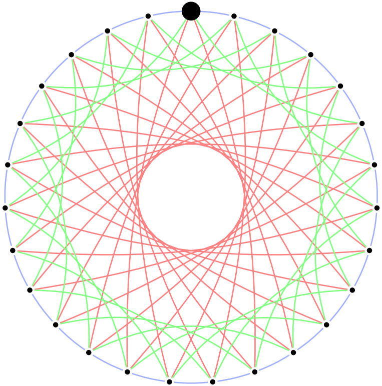
Alice

[+, +, −, −]



Bob

[−, +, −, −]

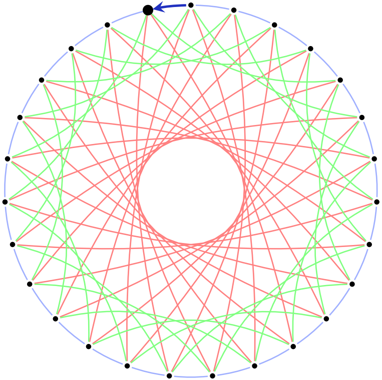


Our work: key exchange on the CSIDH graph

Alice

[+, +, −, −]

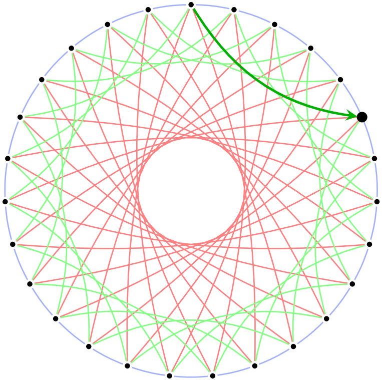
↑



Bob

[−, +, −, −]

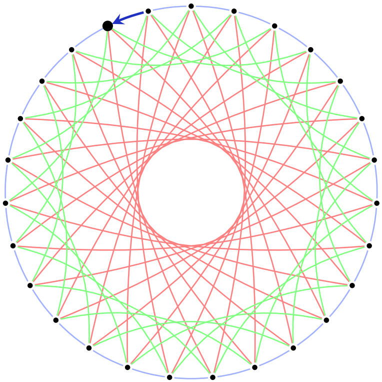
↑



Our work: key exchange on the CSIDH graph

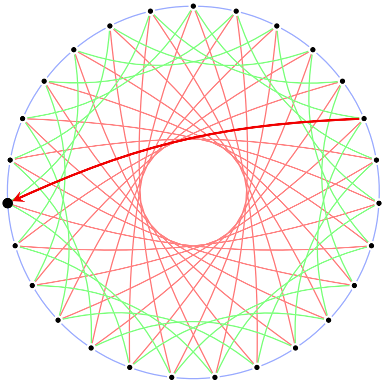
Alice

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



Bob

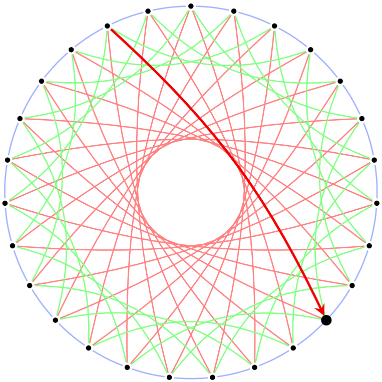
$[-, \underset{\uparrow}{+}, -, -]$



Our work: key exchange on the CSIDH graph

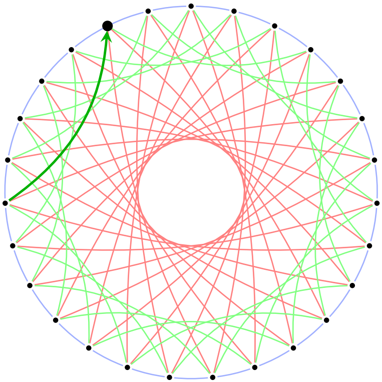
Alice

[+, +, −, −]
↑



Bob

[−, +, −, −]
↑

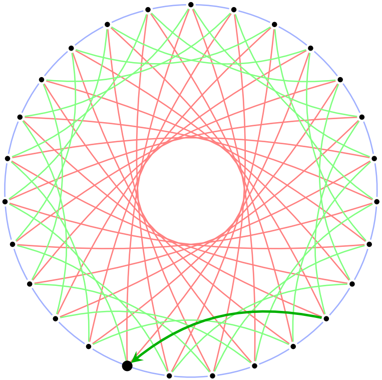


Our work: key exchange on the CSIDH graph

Alice

[+, +, −, −]

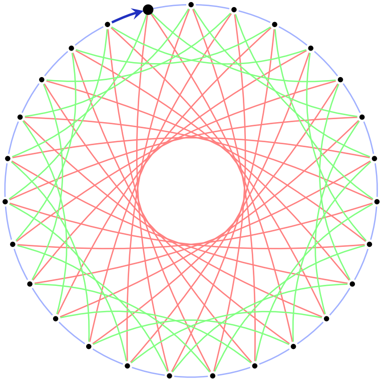
↑



Bob

[−, +, −, −]

↑



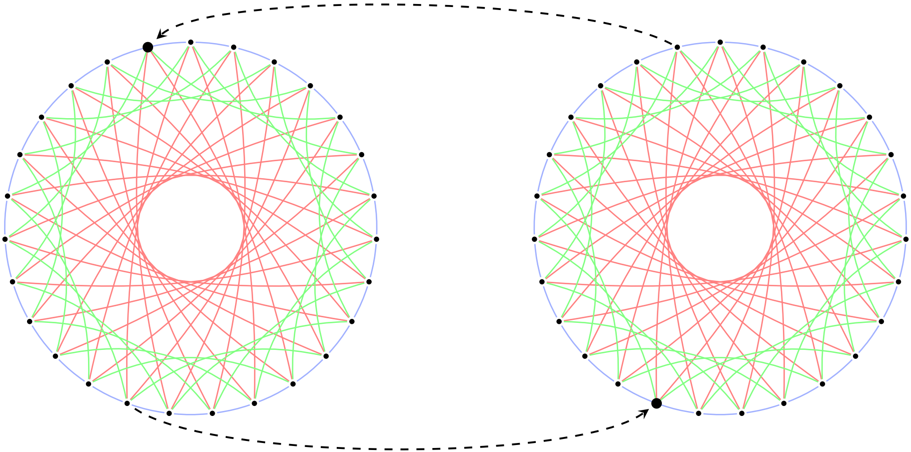
Our work: key exchange on the CSIDH graph

Alice

[+, +, −, −]

Bob

[−, +, −, −]

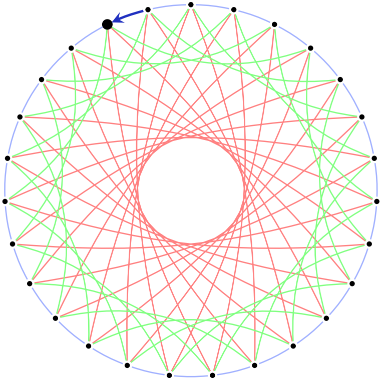


Our work: key exchange on the CSIDH graph

Alice

[+, +, −, −]

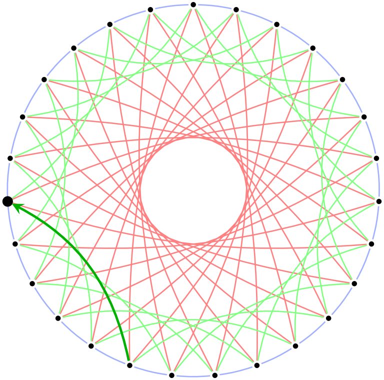
↑



Bob

[−, +, −, −]

↑

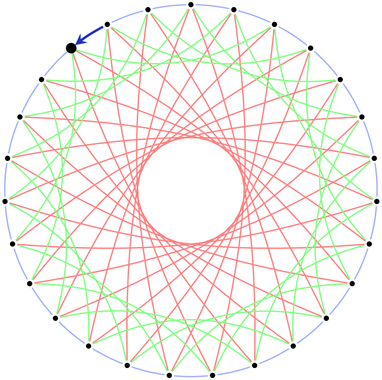


Our work: key exchange on the CSIDH graph

Alice

$[+, +, -, -]$

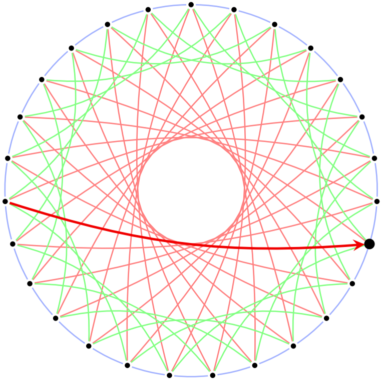
↑



Bob

$[-, +, -, -]$

↑

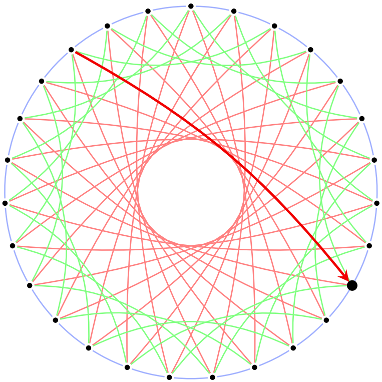


Our work: key exchange on the CSIDH graph

Alice

$[+, +, -, -]$

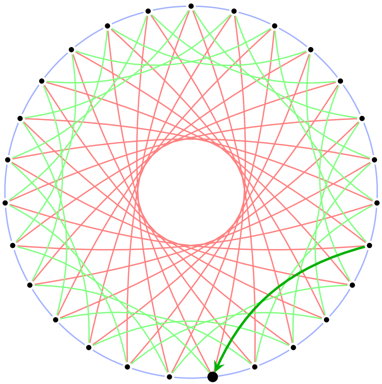
↑



Bob

$[-, +, -, -]$

↑

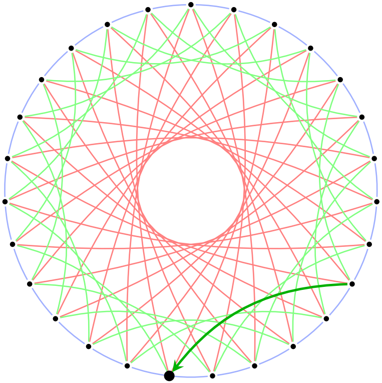


Our work: key exchange on the CSIDH graph

Alice

$[+, +, -, -]$

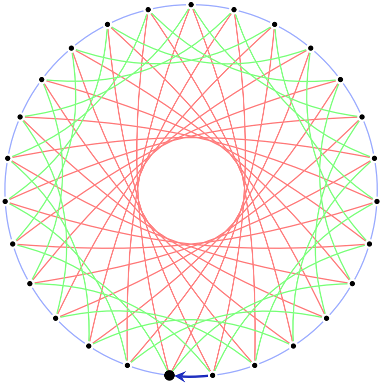
↑



Bob

$[-, +, -, -]$

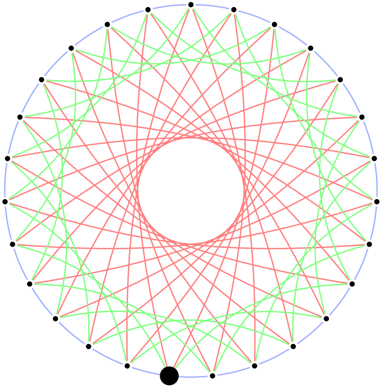
↑



Our work: key exchange on the CSIDH graph

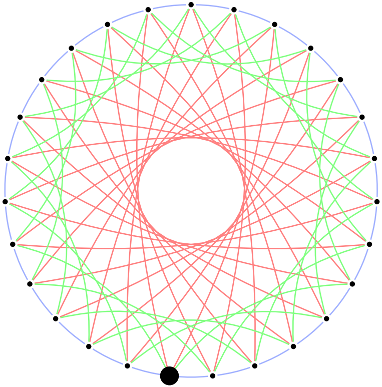
Alice

[+, +, −, −]



Bob

[−, +, −, −]



Our work: conclusion

We have constructed an exponentially-sized[1] graph such that:

[1] (Nobody can enumerate or even store the whole thing.)

Our work: conclusion

We have constructed an exponentially-sized[1] graph such that:

- It is efficient to walk on the graph.

[1] (Nobody can enumerate or even store the whole thing.)

Our work: conclusion

We have constructed an exponentially-sized[1] graph such that:

- It is efficient to walk on the graph.
- It seems hard to find paths between two given nodes.

[1] (Nobody can enumerate or even store the whole thing.)

Our work: conclusion

We have constructed an exponentially-sized[1] graph such that:

- It is efficient to walk on the graph.
- It seems hard to find paths between two given nodes.
- The graph is structured enough to support $\mathfrak{a} \circ \mathfrak{b} = \mathfrak{b} \circ \mathfrak{a}$...

[1] (Nobody can enumerate or even store the whole thing.)

Our work: conclusion

We have constructed an exponentially-sized[1] graph such that:

- It is efficient to walk on the graph.
- It seems hard to find paths between two given nodes.
- The graph is structured enough to support $\mathfrak{a} \circ \mathfrak{b} = \mathfrak{b} \circ \mathfrak{a}$...
- ...but not regular enough to be broken by any known method.

Implications:

[1] (Nobody can enumerate or even store the whole thing.)

Our work: conclusion

We have constructed an exponentially-sized[1] graph such that:

- It is efficient to walk on the graph.
- It seems hard to find paths between two given nodes.
- The graph is structured enough to support $\mathfrak{a} \circ \mathfrak{b} = \mathfrak{b} \circ \mathfrak{a}$...
- ...but not regular enough to be broken by any known method.

Implications:

- An efficient post-quantum non-interactive key exchange.
  $\Longrightarrow$ more flexible security mechanisms for a `cyber future`!

[1] (Nobody can enumerate or even store the whole thing.)

Our work: conclusion

We have constructed an exponentially-sized[1] graph such that:

- It is efficient to walk on the graph.
- It seems hard to find paths between two given nodes.
- The graph is structured enough to support $\mathfrak{a} \circ \mathfrak{b} = \mathfrak{b} \circ \mathfrak{a}$...
- ...but not regular enough to be broken by any known method.

Implications:

- An efficient post-quantum non-interactive key exchange.
  $\Longrightarrow$ more flexible security mechanisms for a `cyber future`!
- I forgot to say that it also has really small keys.

[1] (Nobody can enumerate or even store the whole thing.)

Our work: conclusion

We have constructed an exponentially-sized[1] graph such that:

- It is efficient to walk on the graph.
- It seems hard to find paths between two given nodes.
- The graph is structured enough to support $\mathfrak{a} \circ \mathfrak{b} = \mathfrak{b} \circ \mathfrak{a}$...
- ...but not regular enough to be broken by any known method.

Implications:

- An efficient post-quantum non-interactive key exchange.
  $\Longrightarrow$ more flexible security mechanisms for a `cyber future`!
- I forgot to say that it also has really small keys.
- Lots of nice math (elliptic curves & isogenies & class groups)!

[1] (Nobody can enumerate or even store the whole thing.)

Our work: conclusion

We have constructed an exponentially-sized[1] graph such that:

- It is efficient to walk on the graph.
- It seems hard to find paths between two given nodes.
- The graph is structured enough to support $\mathfrak{a} \circ \mathfrak{b} = \mathfrak{b} \circ \mathfrak{a}$...
- ...but not regular enough to be broken by any known method.

Implications:

- An efficient post-quantum non-interactive key exchange. $\implies$ more flexible security mechanisms for a `cyber future`!
- I forgot to say that it also has really small keys.
- Lots of nice math (elliptic curves & isogenies & class groups)!

Thank you!

[1] (Nobody can enumerate or even store the whole thing.)