Revisiting cycles of pairing-friendly elliptic curves

Marta Bellés-Muñoz, Jorge Jiménez Urroz, **Javier Silva**

CRYPTO 2023

A cycle of elliptic curves

The problem:

$$E/\mathbb{F}_q \qquad \text{such that} \qquad \#E(\mathbb{F}_q) = p$$
$$E'/\mathbb{F}_p \qquad \qquad \qquad \#E'(\mathbb{F}_p) = q$$

A cycle of elliptic curves

The problem:

$$E/\mathbb{F}_q \qquad \text{such that} \qquad \#E(\mathbb{F}_q) = p$$
$$E'/\mathbb{F}_p \qquad \qquad \qquad \#E'(\mathbb{F}_p) = q$$

- This problem is easy.

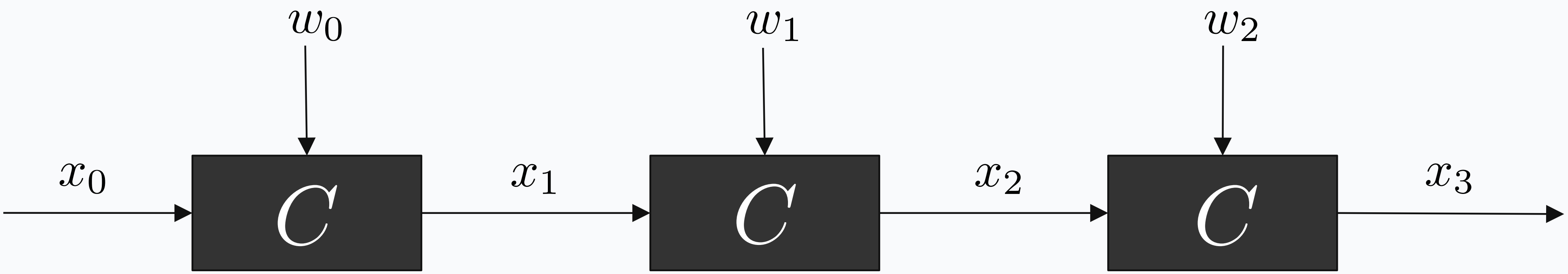
A cycle of elliptic curves

The problem:

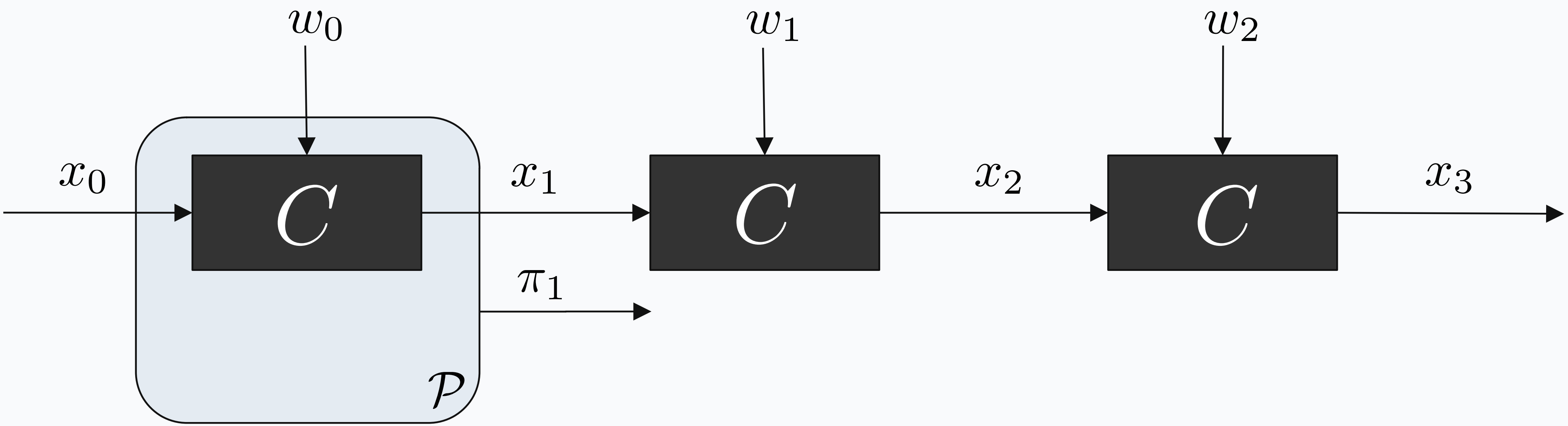
$$E/\mathbb{F}_q \qquad \text{such that} \qquad \#E(\mathbb{F}_q) = p$$
$$E'/\mathbb{F}_p \qquad \qquad \qquad \#E'(\mathbb{F}_p) = q$$

- This problem is easy.
- But if we require E, E’ to be **pairing-friendly**, the problem becomes hard.

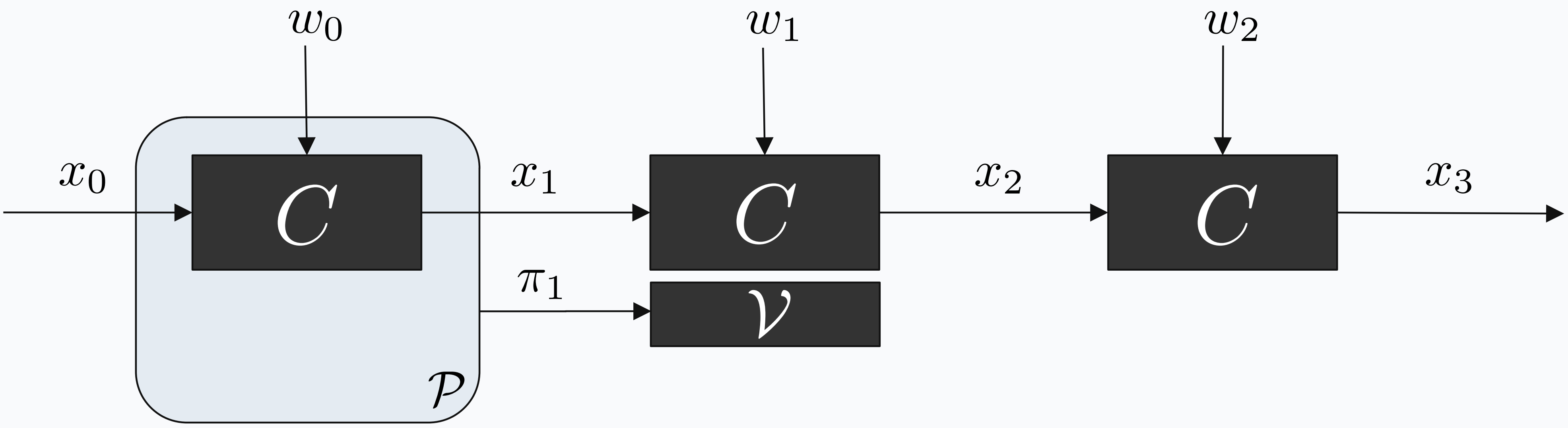
Recursive composition of proofs



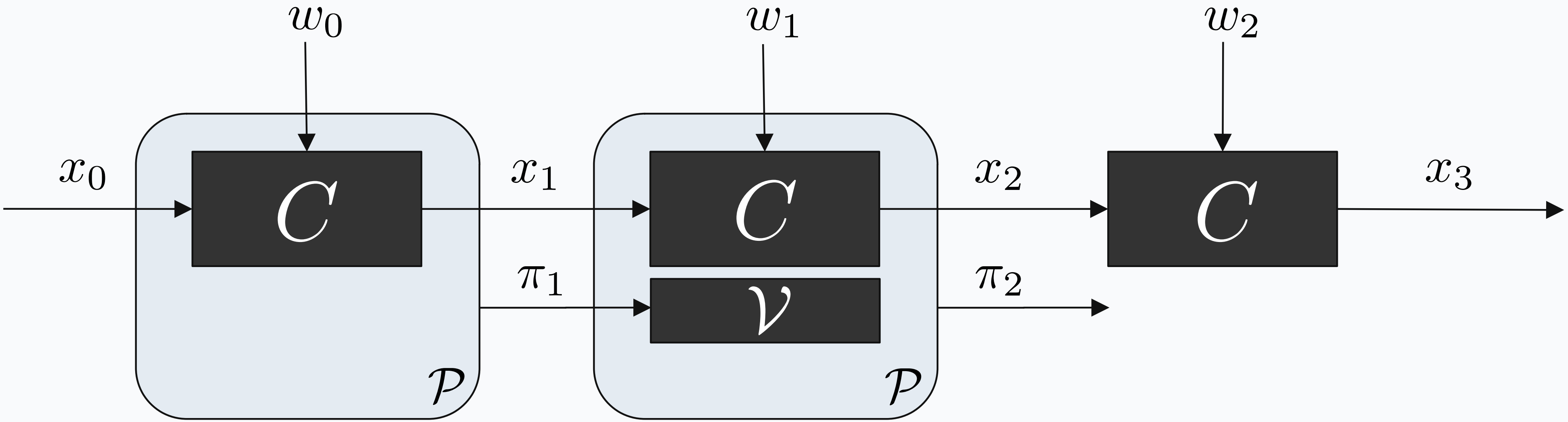
Recursive composition of proofs



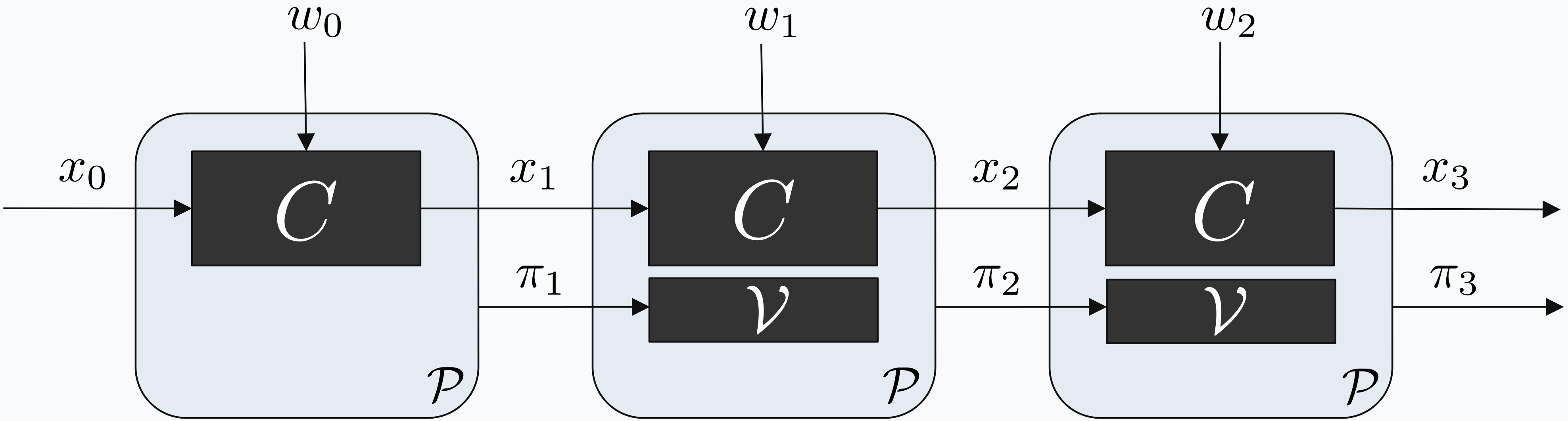
Recursive composition of proofs



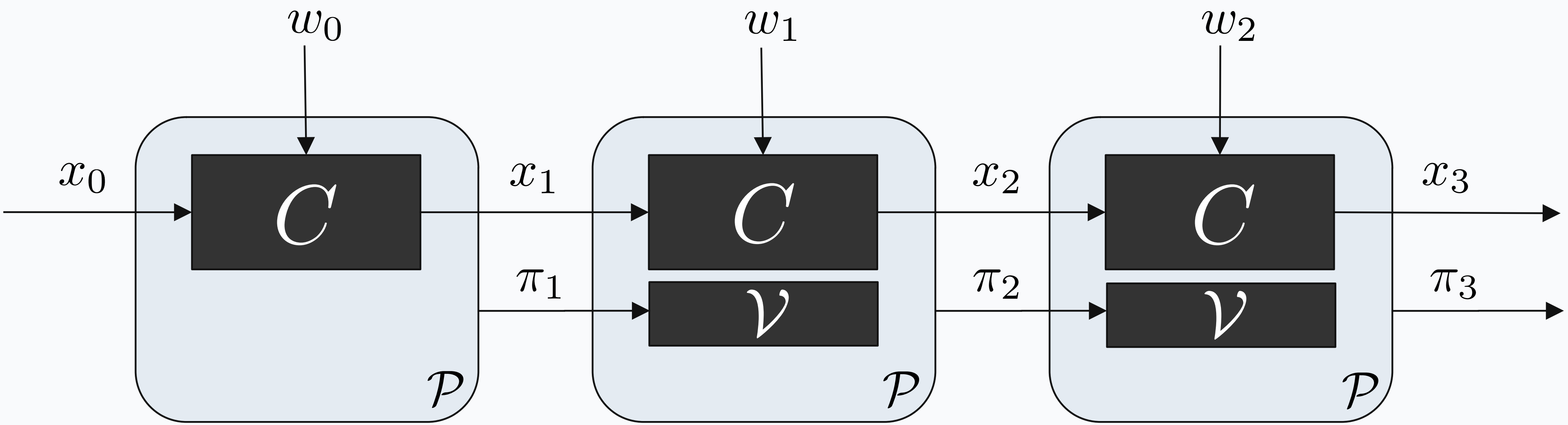
Recursive composition of proofs



Recursive composition of proofs

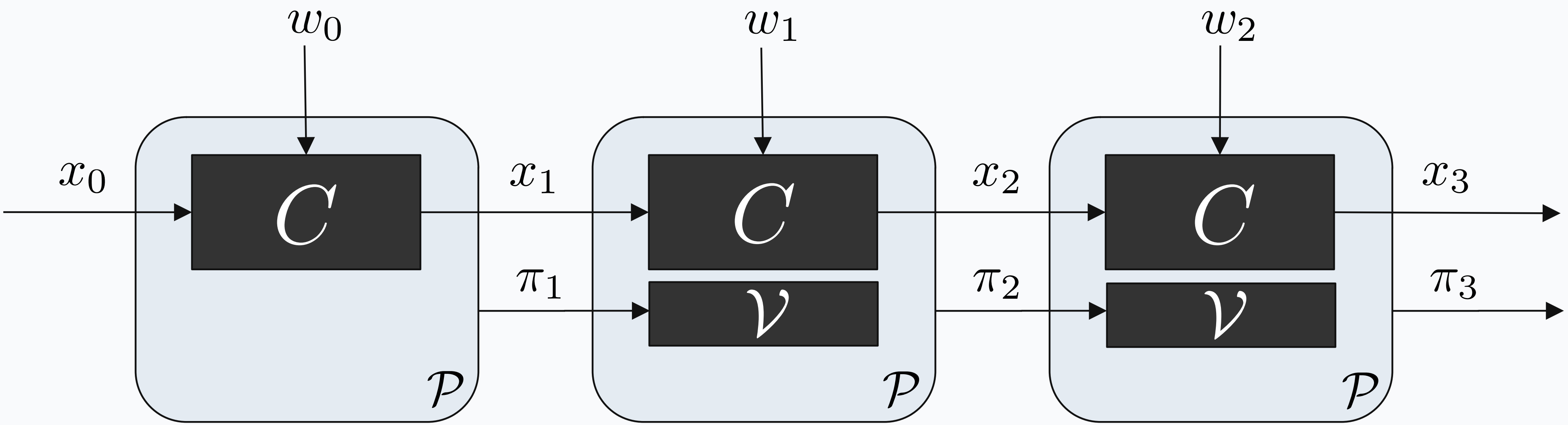


Recursive composition of proofs



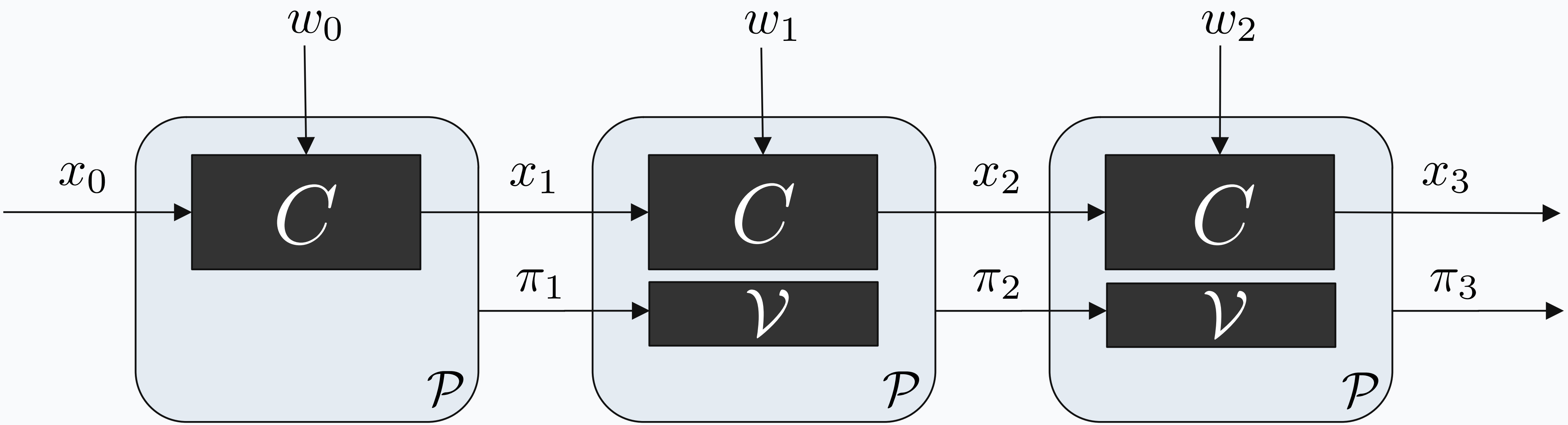
- The final verifier only needs to check **one proof**.

Recursive composition of proofs



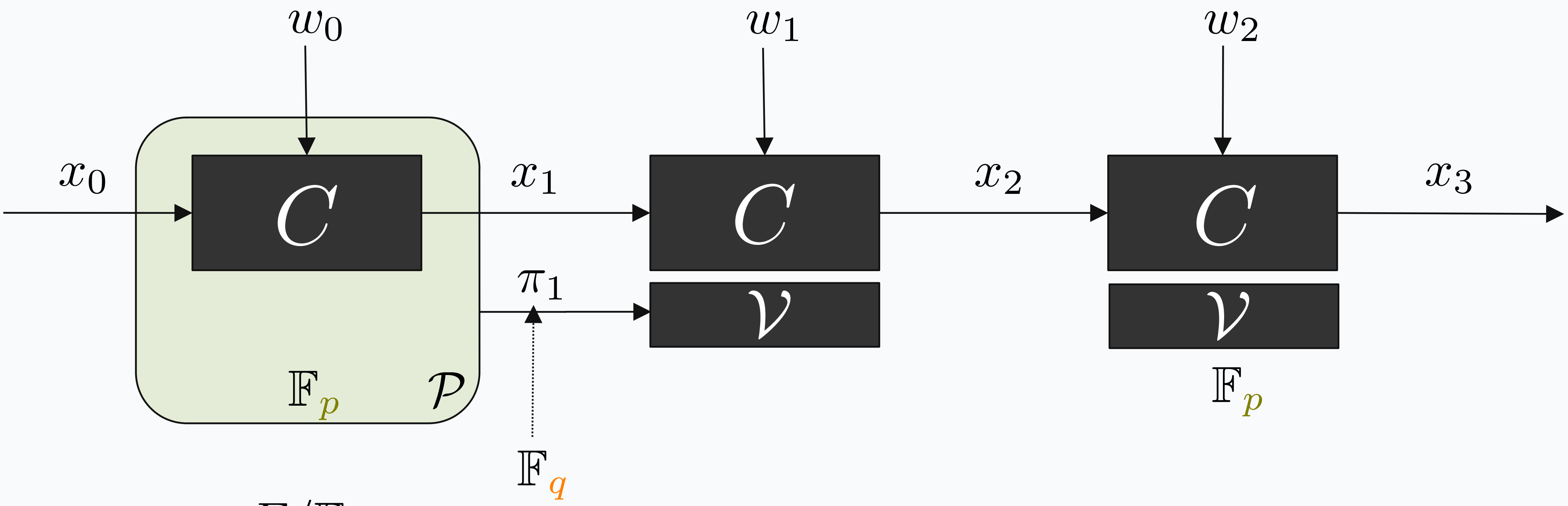
- The final verifier only needs to check **one proof**.
- We focus on the case of **pairing-friendly** SNARKs.

Recursive composition of proofs



- The final verifier only needs to check **one proof**.
- We focus on the case of **pairing-friendly** SNARKs.
- We need to be able to write $\mathcal{V}$ in the language of the SNARK.

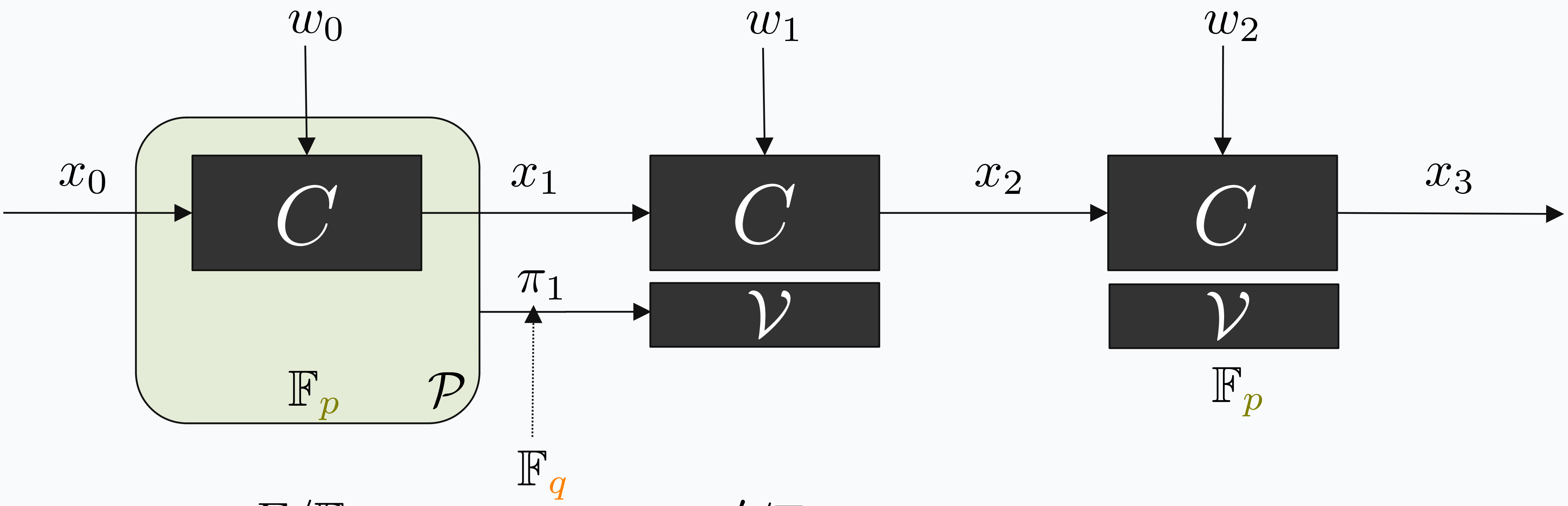
Recursion with cycles



$E/\mathbb{F}_q$

$\#E(\mathbb{F}_q) = p$

Recursion with cycles



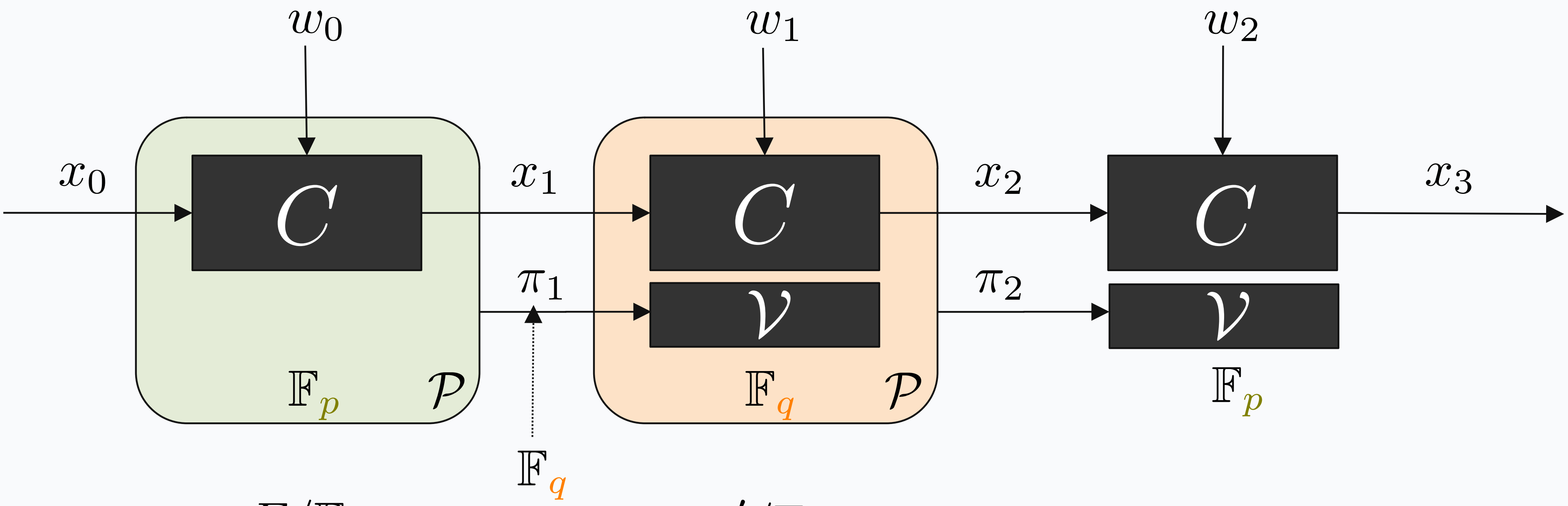
$E/\mathbb{F}_q$

$\#E(\mathbb{F}_q) = p$

$E'/\mathbb{F}_p$

$\#E'(\mathbb{F}_p) = q$

Recursion with cycles



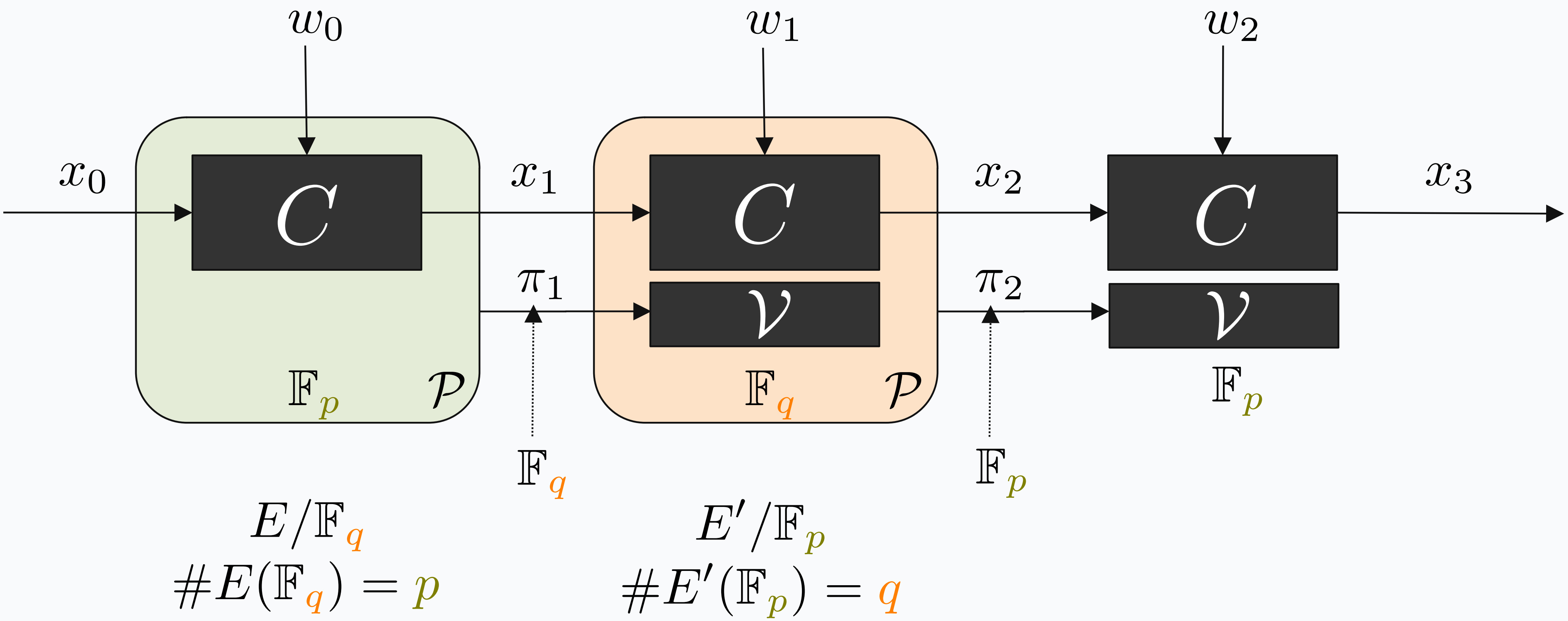
$E/\mathbb{F}_q$

$\#E(\mathbb{F}_q) = p$

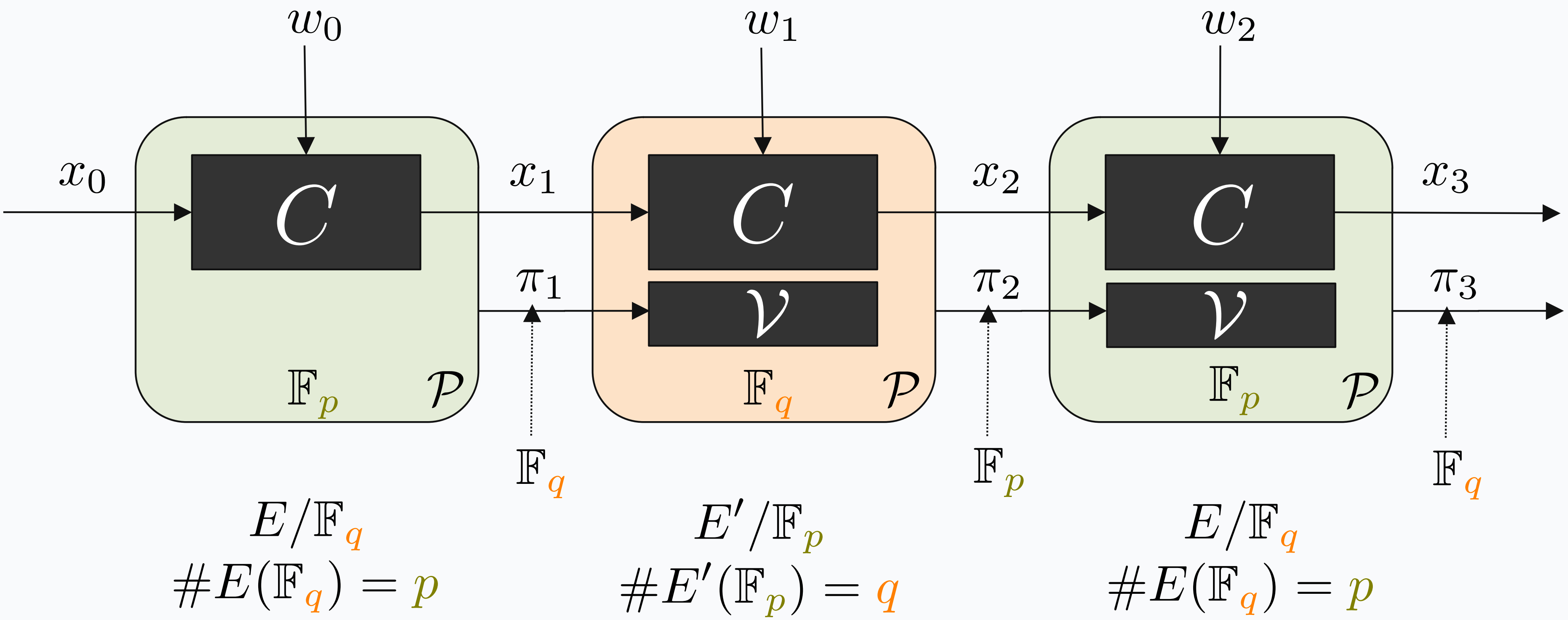
$E'/\mathbb{F}_p$

$\#E'(\mathbb{F}_p) = q$

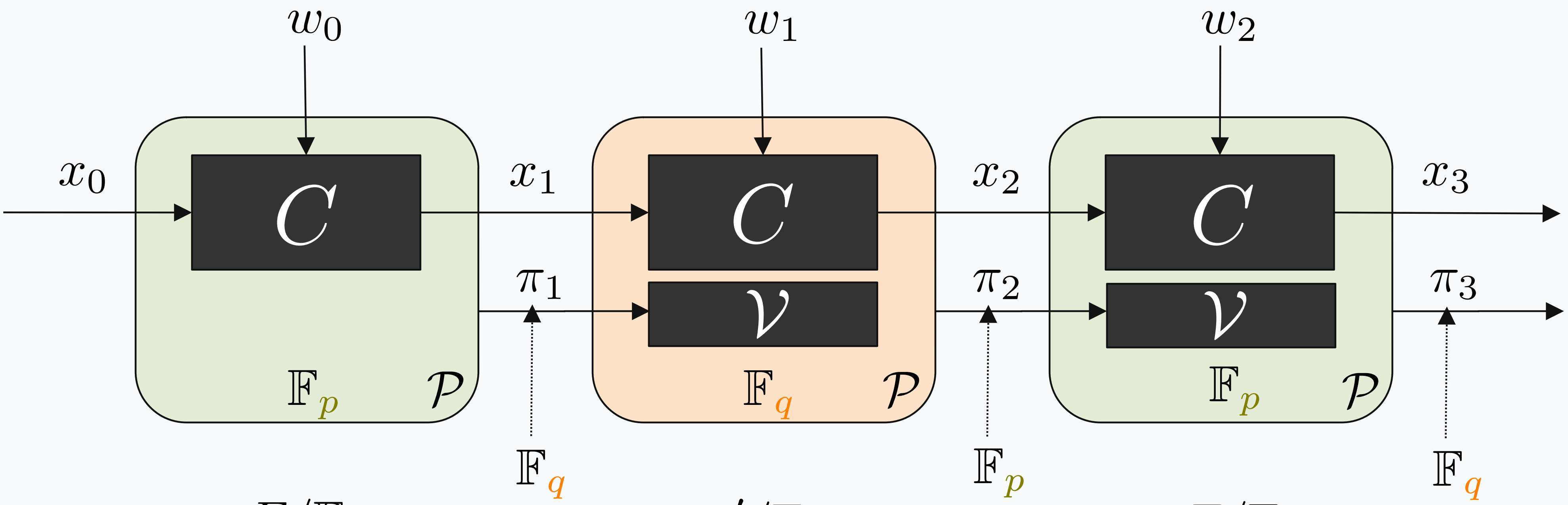
Recursion with cycles



Recursion with cycles



Recursion with cycles



$E/\mathbb{F}_q$ $\quad$ $E'/\mathbb{F}_p$ $\quad$ $E/\mathbb{F}_q$

$\#E(\mathbb{F}_q) = p$ $\quad$ $\#E'(\mathbb{F}_p) = q$ $\quad$ $\#E(\mathbb{F}_q) = p$

Only **two curves** needed in total.

A cycle of elliptic curves

The problem:

$E/\mathbb{F}_q$ such that $\#E(\mathbb{F}_q) = p$

$E'/\mathbb{F}_p$ such that $\#E'(\mathbb{F}_p) = q$

We want E, E' to be **pairing-friendly**.

$$e : E(\mathbb{F}_q) \times E(\mathbb{F}_q) \to \mu_p \subset \mathbb{F}_{q^k}$$

A cycle of elliptic curves

The problem:

$E/\mathbb{F}_q$ such that $\#E(\mathbb{F}_q) = p$

$E'/\mathbb{F}_p$ $\#E'(\mathbb{F}_p) = q$

We want E, E' to be **pairing-friendly**.

$$e : E(\mathbb{F}_q) \times E(\mathbb{F}_q) \to \mu_p \subset \mathbb{F}_{q^k}$$

← **embedding degree** of E

↑ roots of unity of order p

A cycle of elliptic curves

The problem:

$E/\mathbb{F}_q$ such that $\#E(\mathbb{F}_q) = p$

$E'/\mathbb{F}_p$ such that $\#E'(\mathbb{F}_p) = q$

We want E, E' to be **pairing-friendly**.

$$e : E(\mathbb{F}_q) \times E(\mathbb{F}_q) \to \mu_p \subset \mathbb{F}_{q^k}$$

k: **embedding degree** of E

μ_p: roots of unity of order p

$p \mid q^k - 1$ for **small** k, ℓ (to ensure efficient pairing computation),
$q \mid p^\ell - 1$ but **not too small** (to prevent security degradation).

Known facts about cycles

Known facts about cycles

- No cofactors allowed.

$$\#E(\mathbb{F}_q) = f \cdot p$$
$$\#E'(\mathbb{F}_p) = f' \cdot q$$

Known facts about cycles

- No cofactors allowed.

$$\#E(\mathbb{F}_q) = f \cdot p$$
$$\#E'(\mathbb{F}_p) = f' \cdot q$$

- What **prime-order pairing-friendly curves** are known?

Known facts about cycles

- No cofactors allowed.

$$\#E(\mathbb{F}_q) = f \cdot p$$
$$\#E'(\mathbb{F}_p) = f' \cdot q$$

- What **prime-order pairing-friendly curves** are known?

Polynomial families

- $p\ (X)$, $q\ (X)$ polynomials.

Known facts about cycles

- No cofactors allowed.

$$\#E(\mathbb{F}_q) = f \cdot p$$
$$\#E'(\mathbb{F}_p) = f' \cdot q$$

✗

- What **prime-order pairing-friendly curves** are known?

Polynomial families

- $p(X)$, $q(X)$ polynomials.
- For infinitely many x, the values $p(x)$, $q(x)$ correspond to the order of the scalar field and base field of a curve, and both values are prime.

Polynomial families with prime order

MNT3 $(k = 3)$

$$p(X) = 12X^2 - 6X + 1$$

$$q(X) = 12X^2 - 1$$

MNT4 $(k = 4)$

$$p(X) = X^2 + 2X + 2$$

$$q(X) = X^2 + X + 1$$

MNT6 $(k = 6)$

$$p(X) = 4X^2 - 2X + 1$$

$$q(X) = 4X^2 + 1$$

Freeman $(k = 10)$

$$p(X) = 25X^4 + 25X^3 + 15X^2 + 5X + 1$$

$$q(X) = 25X^4 + 25X^3 + 25X^2 + 10X + 3$$

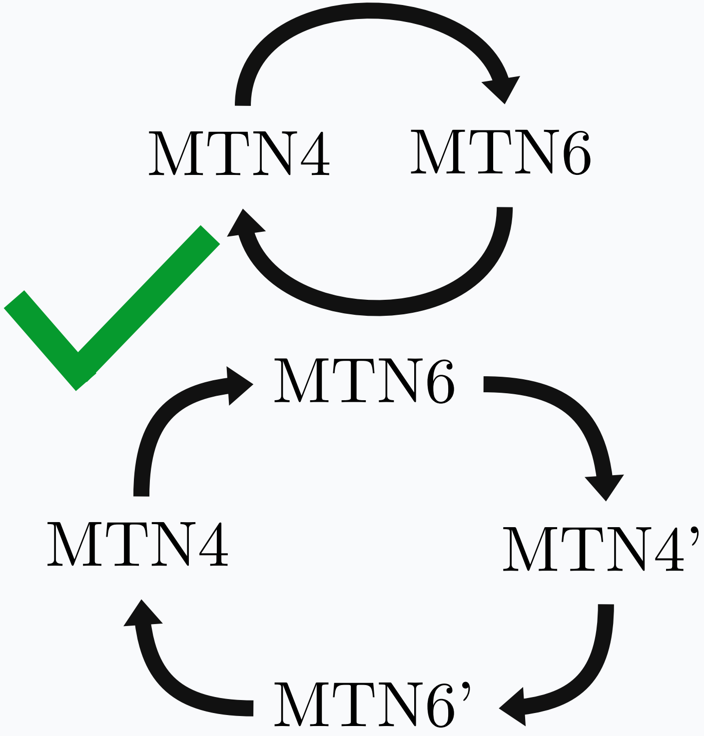
BN $(k = 12)$

$$p(X) = 36X^4 + 36X^3 + 18X^2 + 6X + 1$$

$$q(X) = 36X^4 + 46X^3 + 24X^2 + 6X + 1$$

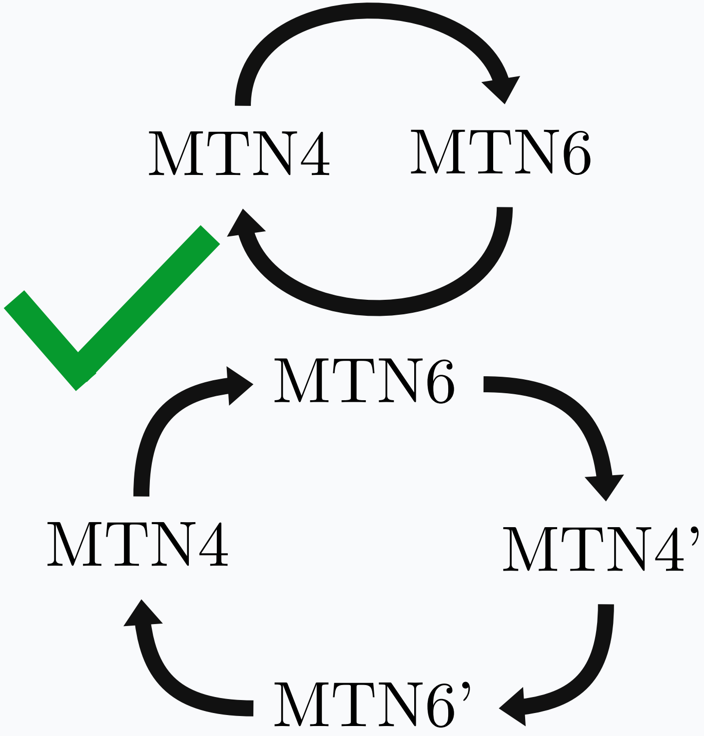
Cycles from known families

Cycles from known families

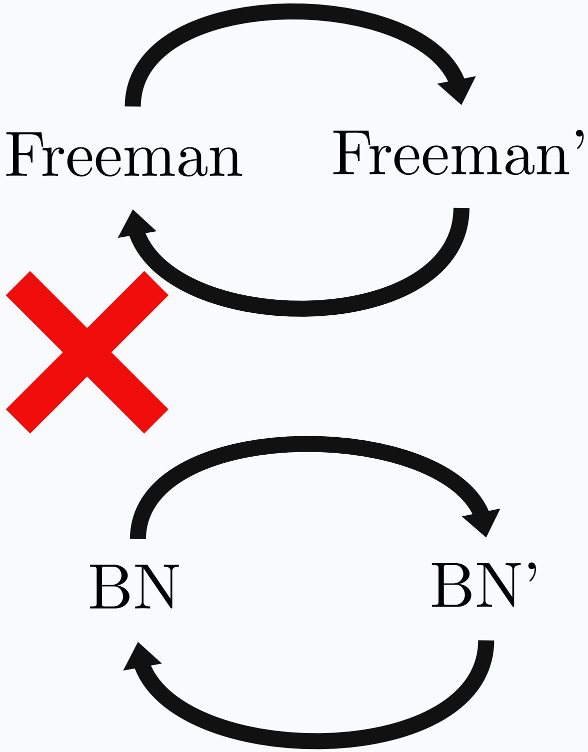


Embedding degree **too small for applications**.

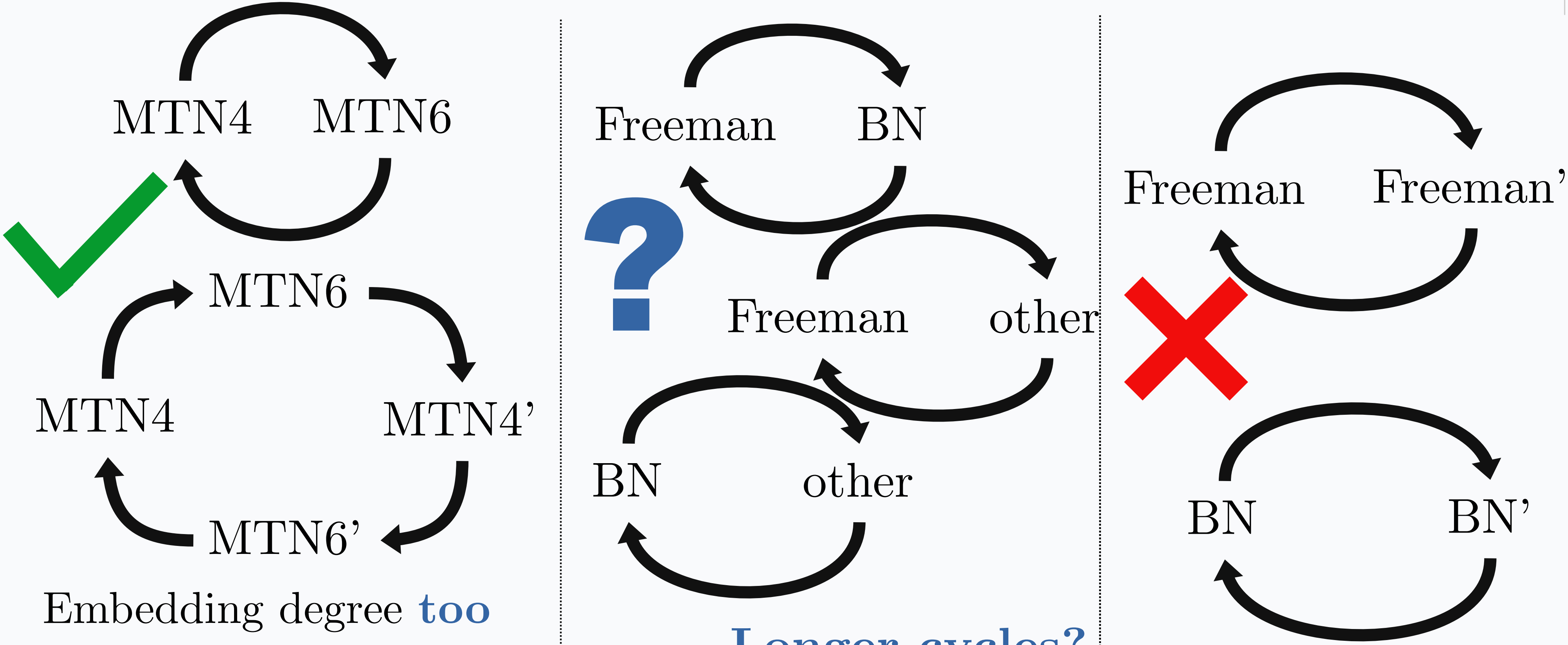
Cycles from known families



Embedding degree too small for applications.



Cycles from known families



Embedding degree too small for applications.

Longer cycles?

Our results

Considering polynomial families:

$$p(X) \mid q(X)^k - 1$$
$$q(X) \mid p(X)^\ell - 1$$

Our results

Considering polynomial families:

$$p(X) \mid q(X)^k - 1$$
$$q(X) \mid p(X)^\ell - 1$$

Given a known family $p(\mathrm{X})$, $q(\mathrm{X})$, the divisibility conditions are straightforward to check.

Our results

Considering polynomial families:

$$p(X) \mid q(X)^k - 1$$
$$q(X) \mid p(X)^\ell - 1$$

Given a known family $p(\mathrm{X})$, $q(\mathrm{X})$, the divisibility conditions are straightforward to check.

- If they hold, there is a polynomial family of cycles.

Our results

Considering polynomial families:

$$p(X) \mid q(X)^k - 1$$
$$q(X) \mid p(X)^\ell - 1$$

Given a known family $p(\mathrm{X})$, $q(\mathrm{X})$, the divisibility conditions are straightforward to check.

- If they hold, there is a polynomial family of cycles.
- If they fail, there are only **finitely many** x such that

$$p(x) \mid q(x)^k - 1$$
$$q(x) \mid p(x)^\ell - 1$$

Our results

Considering polynomial families:

$$p(X) \mid q(X)^k - 1$$
$$q(X) \mid p(X)^\ell - 1$$

Given a known family $p(\mathrm{X})$, $q(\mathrm{X})$, the divisibility conditions are straightforward to check.

- If they hold, there is a polynomial family of cycles.
- If they fail, there are only **finitely many** x such that

$$p(x) \mid q(x)^k - 1$$
$$q(x) \mid p(x)^\ell - 1$$

We can find explicit bounds on x for different embedding degrees.

Our results

Our results

- For embedding degree up to $\ell \leq 22$, we run an exhaustive search for 2-cycles for MNT3, Freeman and BN curves.

Our results

- For embedding degree up to $\ell \leq 22$, we run an exhaustive search for 2-cycles for MNT3, Freeman and BN curves.
- We show that no 2-cycles exist with the exception of a few toy examples.

Family	k	ℓ	x	t	p	q
MNT3	3	10	-1	-7	19	11
MNT3	3	10	1	5	7	11
BN	12	18	-1	7	13	19

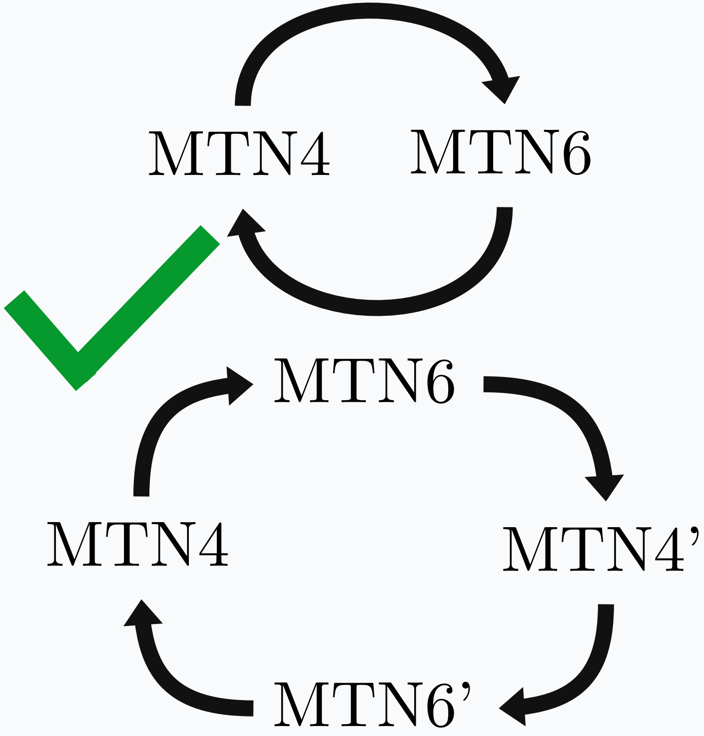
Our results

We could go higher, but the bounds on x grow quite fast.

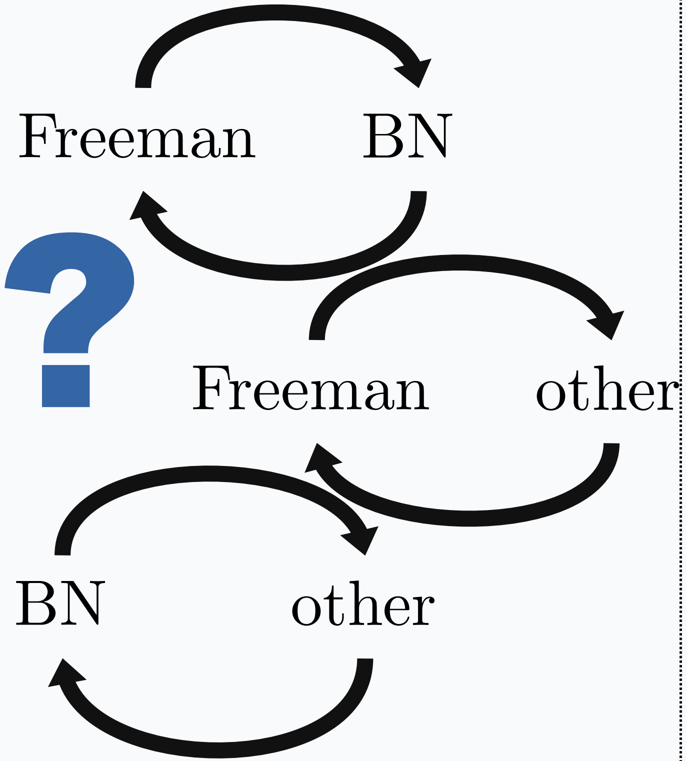
- For embedding degree up to $\ell \leq 22$, we run an exhaustive search for 2-cycles for MNT3, Freeman and BN curves.
- We show that no 2-cycles exist with the exception of a few toy examples.

Family	k	ℓ	x	t	p	q
MNT3	3	10	-1	-7	19	11
MNT3	3	10	1	5	7	11
BN	12	18	-1	7	13	19

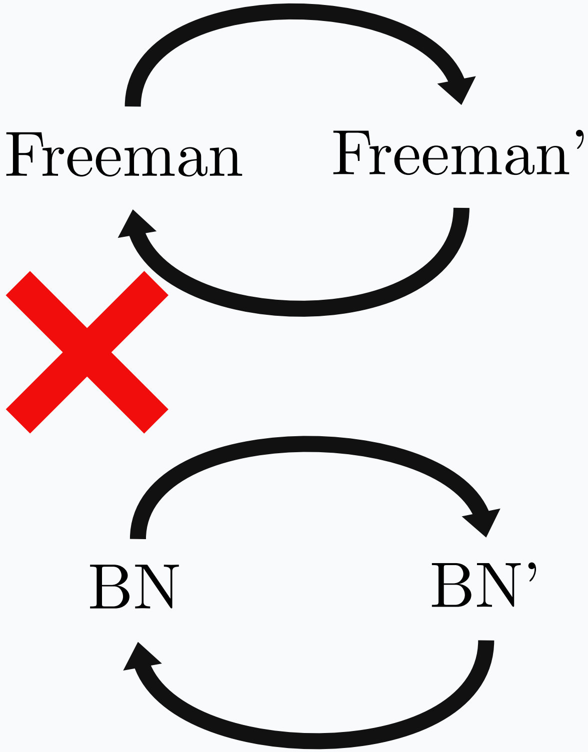
Cycles from known families



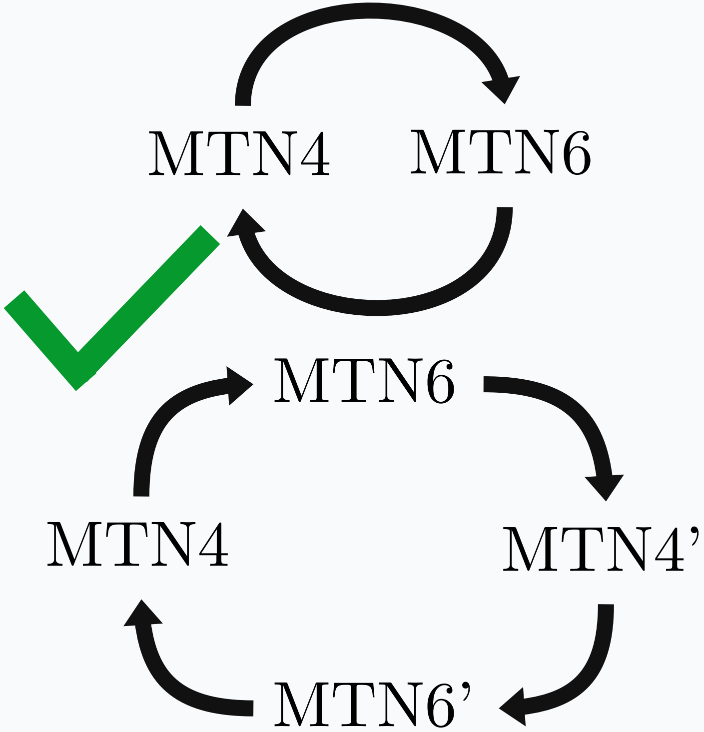
Embedding degree too small for applications.



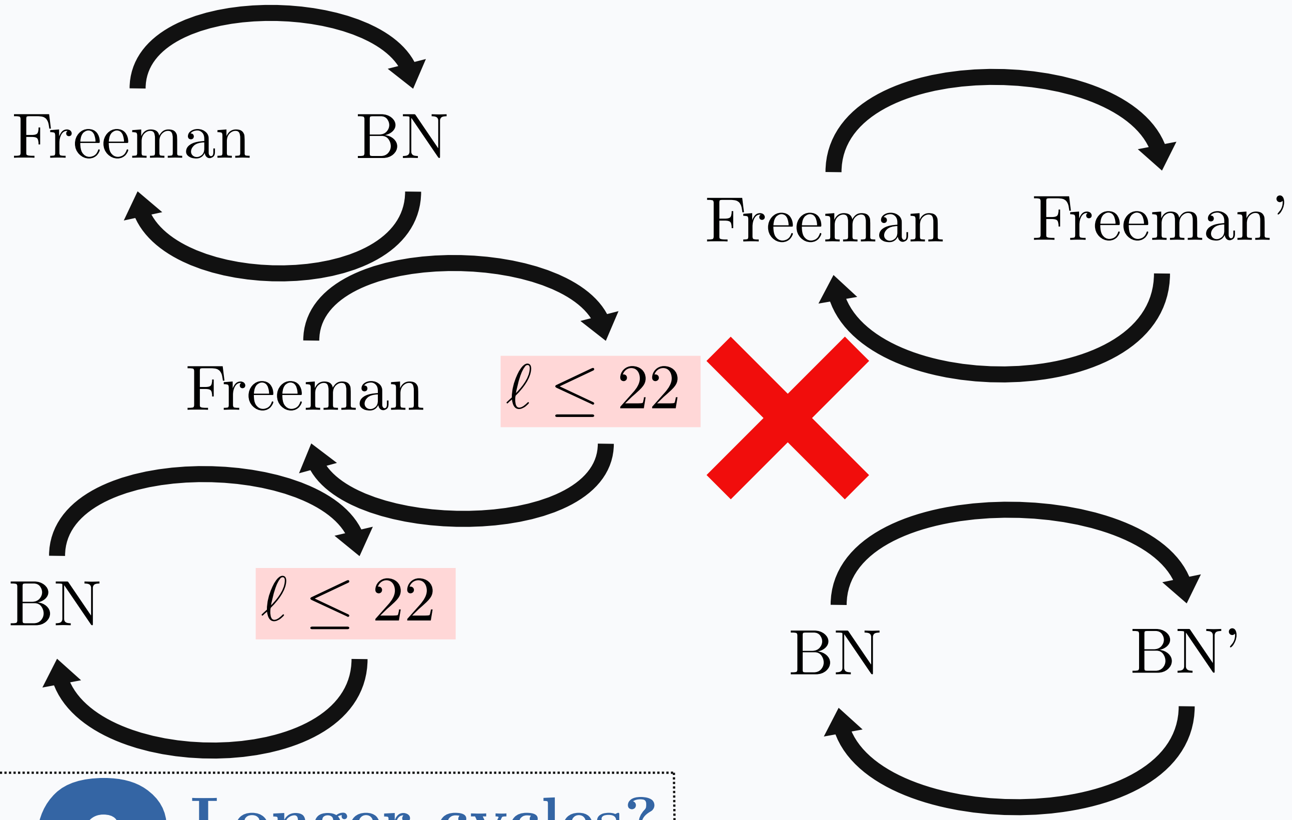
Longer cycles?



New impossibility results



Embedding degree too small for applications.



Longer cycles?

In conclusion

In conclusion

- Cycles are composed of prime-order curves.

In conclusion

- Cycles are composed of prime-order curves.
- The only known method to generate prime-order pairing-friendly curves is through **polynomial families**.

In conclusion

- Cycles are composed of prime-order curves.
- The only known method to generate prime-order pairing-friendly curves is through **polynomial families**.
- We show that **known families do not contain any 2-cycles that are pairing-friendly** (with the exception of the inefficient MNT4-MNT6 cycles.

In conclusion

- Cycles are composed of prime-order curves.
- The only known method to generate prime-order pairing-friendly curves is through **polynomial families**.
- We show that **known families do not contain any 2-cycles that are pairing-friendly** (with the exception of the inefficient MNT4-MNT6 cycles.
- The technique easily extends to any new polynomial family that might appear in the future.

In conclusion

- Cycles are composed of prime-order curves.
- The only known method to generate prime-order pairing-friendly curves is through **polynomial families**.
- We show that **known families do not contain any 2-cycles that are pairing-friendly** (with the exception of the inefficient MNT4-MNT6 cycles.
- The technique easily extends to any new polynomial family that might appear in the future.
- The code runs in a few hours for embedding degree up to 22, but there is a lot of room for optimization.

In conclusion

- Cycles are composed of prime-order curves.
- The only known method to generate prime-order pairing-friendly curves is through **polynomial families**.
- We show that **known families do not contain any 2-cycles that are pairing-friendly** (with the exception of the inefficient MNT4-MNT6 cycles.
- The technique easily extends to any new polynomial family that might appear in the future.
- The code runs in a few hours for embedding degree up to 22, but there is a lot of room for optimization.
- We also provide density estimates of pairing-friendly cycles among all cycles.

Thank you!

Paper

Code

Questions?