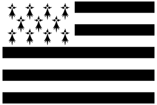
Elliptic curves for SNARK and proof systems

Diego F. Aranha[1], Youssef El Housni[2], Aurore Guillevic[3]

[1]Aarhus University, Denmark, [2]Consensys – Linea, NYC US, [3]Inria Rennes, France

Journées Numération, Arithmétique, Cryptographie
February 29 – March 1, 2024



https://webusers.imj-prg.fr/~jean-claude.bajard/NAC2024/

Outline

zk-SNARK

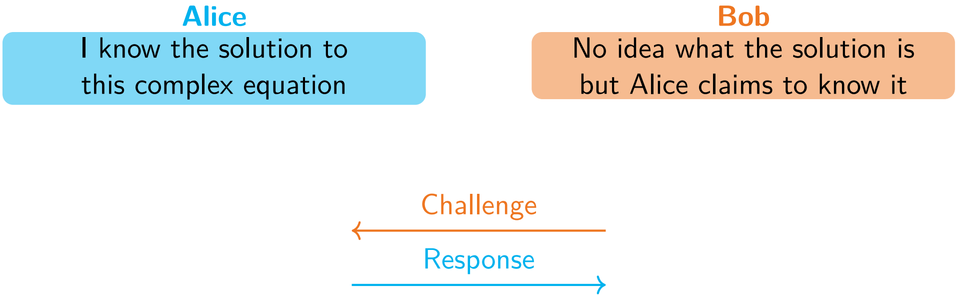
Elliptic Curves

Pairings

Pairing-friendly curves

SNARK-friendly curves

Zero-knowledge proofs (ZKP)



Alice

I know the solution to this complex equation

Bob

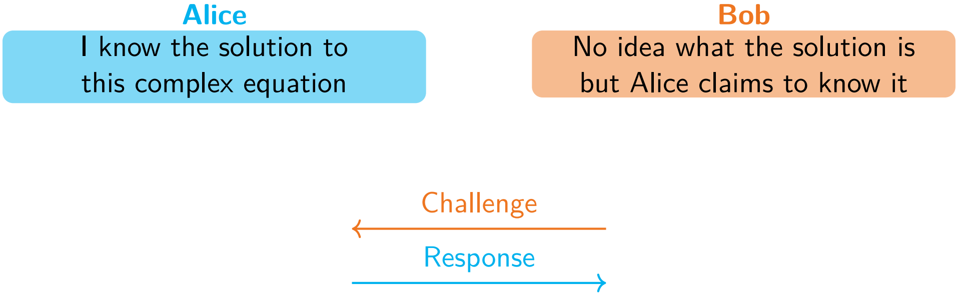
No idea what the solution is but Alice claims to know it

Challenge

Response

slide Y. El Housni

Zero-knowledge proofs (ZKP)



Alice

I know the solution to this complex equation

Bob

No idea what the solution is but Alice claims to know it

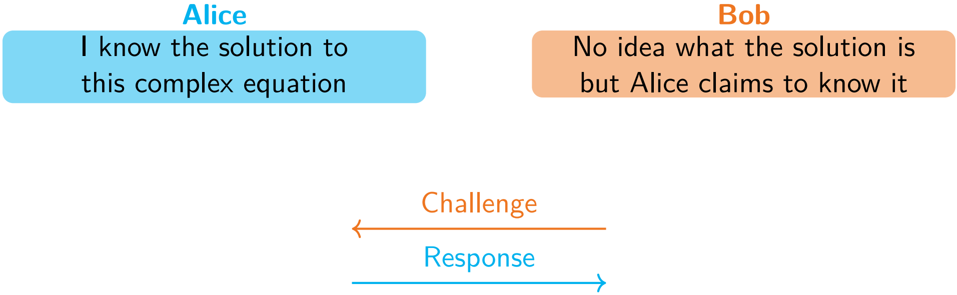
Challenge

Response

- **Sound**: **Alice** has a wrong solution $\implies$ **Bob** is not convinced.

slide Y. El Housni

Zero-knowledge proofs (ZKP)



Alice

I know the solution to this complex equation

Bob

No idea what the solution is but Alice claims to know it

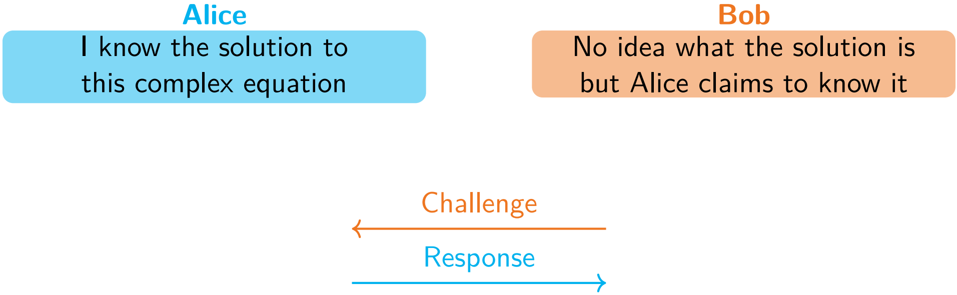
Challenge

Response

- **Sound**: **Alice** has a wrong solution $\implies$ **Bob** is not convinced.
- **Complete**: **Alice** has the solution $\implies$ **Bob** is convinced.

slide Y. El Housni

Zero-knowledge proofs (ZKP)



Alice

I know the solution to this complex equation

Bob

No idea what the solution is but Alice claims to know it

Challenge

Response

- **Sound**: **Alice** has a wrong solution $\implies$ **Bob** is not convinced.
- **Complete**: **Alice** has the solution $\implies$ **Bob** is convinced.
- **Zero-knowledge**: **Bob** does NOT learn the solution.

slide Y. El Housni

Example: Sigma protocol

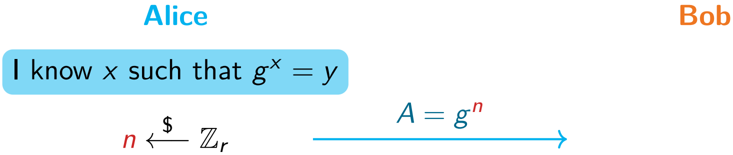
Alice

Bob

I know x such that $g^x = y$

slide Y. El Housni

Example: Sigma protocol



Alice

Bob

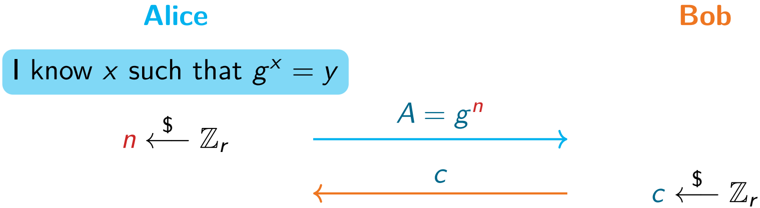
I know x such that $g^x = y$

$n \xleftarrow{\$} \mathbb{Z}_r$

$\xrightarrow{A = g^n}$

slide Y. El Housni

Example: Sigma protocol



Alice

Bob

I know x such that $g^x = y$

$n \xleftarrow{\$} \mathbb{Z}_r$

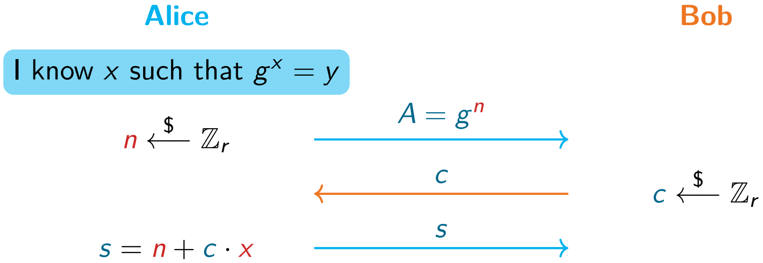
$A = g^n$

c

$c \xleftarrow{\$} \mathbb{Z}_r$

slide Y. El Housni

Example: Sigma protocol



Alice

Bob

I know x such that $g^x = y$

$n \xleftarrow{\$} \mathbb{Z}_r$

$A = g^n$

c

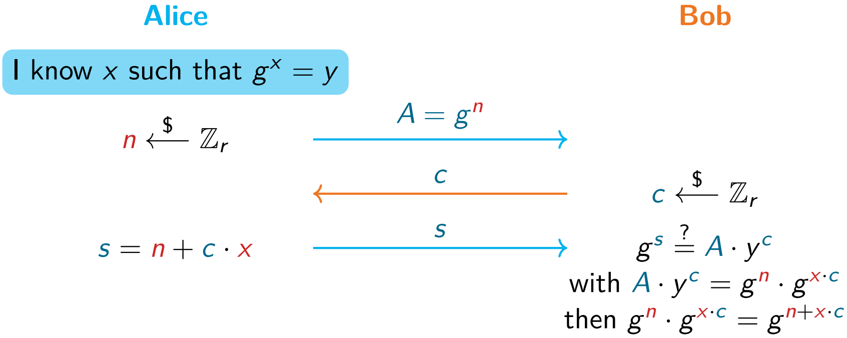
$c \xleftarrow{\$} \mathbb{Z}_r$

$s = n + c \cdot x$

s

slide Y. El Housni

Example: Sigma protocol



Alice

Bob

I know x such that $g^x = y$

$n \xleftarrow{\$} \mathbb{Z}_r$

$\xrightarrow{A = g^n}$

$\xleftarrow{c}$

$c \xleftarrow{\$} \mathbb{Z}_r$

$s = n + c \cdot x$

$\xrightarrow{s}$

$g^s \overset{?}{=} A \cdot y^c$

with $A \cdot y^c = g^n \cdot g^{x \cdot c}$

then $g^n \cdot g^{x \cdot c} = g^{n + x \cdot c}$

slide Y. El Housni

Non-Interactive Zero-Knowledge (NIZK) Sigma protocol

Alice **Bob**

I know x such that $g^{x} = y$

$$n \xleftarrow{\$} \mathbb{Z}_r$$

$$g \quad ; A = g^{n}$$

$$c = H(A, y)$$

$$s = n + c \cdot x$$

Non-Interactive Zero-Knowledge (NIZK) Sigma protocol

Alice **Bob**

I know x such that $g^x = y$

$$n \xleftarrow{\$} \mathbb{Z}_r$$

$$g \quad ; A = g^{n}$$

$$c = H(A, y)$$

$$s = n + c \cdot x \qquad \xrightarrow[\pi = (A, c, s)]{\hspace{4cm}}$$

Non-Interactive Zero-Knowledge (NIZK) Sigma protocol

Alice

Bob

I know x such that $g^x = y$

$$n \xleftarrow{\$} \mathbb{Z}_r$$

$$g \quad ; A = g^n$$

$$c = H(A, y)$$

$$s = n + c \cdot x$$

$$\xrightarrow[\pi = (A, c, s)]{}$$

$$g^s \stackrel{?}{=} A \cdot y^c$$

$$c \stackrel{?}{=} H(A, y)$$

slide Y. El Housni

Non-Interactive Zero-Knowledge (NIZK) Sigma protocol

Alice

Bob

I know x such that $g^x = y$

$$n \xleftarrow{\$} \mathbb{Z}_r$$

$$\underbrace{g}_{\text{Setup}} \; ; \; A = g^n$$

$$c = H(A, y)$$

$$\underbrace{s = n + c \cdot x}_{\text{Prove}}$$

$$\xrightarrow{\qquad\qquad} $$

$$\underbrace{\pi = (A, c, s)}_{\text{proof}}$$

$$g^s \overset{?}{=} A \cdot y^c$$

$$\underbrace{c \overset{?}{=} H(A, y)}_{\text{Verify}}$$

slide Y. El Housni

ZKP literature landmarks

- First ZKP work [GMR85]
- Non-Interactive ZKP [BFM88]
- Succinct ZKP [Kil92]
- Succinct Non-Interactive ZKP [Mic94]

slide Y. El Housni

ZKP literature landmarks

- First ZKP work [GMR85]
- Non-Interactive ZKP [BFM88]
- Succinct ZKP [Kil92]
- Succinct Non-Interactive ZKP [Mic94]
- Pairing-based succinct NIZK [Gro10]

slide Y. El Housni

ZKP literature landmarks

- First ZKP work [GMR85]
- Non-Interactive ZKP [BFM88]
- Succinct ZKP [Kil92]
- Succinct Non-Interactive ZKP [Mic94]
- Pairing-based succinct NIZK [Gro10]
- “SNARK” terminology and characterization of existence [BCCT12]
- Pairing-based SNARK in quasi-linear prover time [GGPR13]
- Pairing-based SNARK with shortest proof and verifier time [Gro16]

slide Y. El Housni

ZKP literature landmarks

- First ZKP work [GMR85]
- Non-Interactive ZKP [BFM88]
- Succinct ZKP [Kil92]
- Succinct Non-Interactive ZKP [Mic94]
- Pairing-based succinct NIZK [Gro10]
- “SNARK” terminology and characterization of existence [BCCT12]
- Pairing-based SNARK in quasi-linear prover time [GGPR13]
- Pairing-based SNARK with shortest proof and verifier time [Gro16]
- SNARK with universal and updatable setup [GKM$^+$18], [MBKM19] (Sonic), [GWC19] (PlonK), [CHM$^+$20] (Marlin), ...

slide Y. El Housni

What is a zero-knowledge proof?

“I have a *sound*, *complete* and *zero-knowledge* proof that a statement is true”. [GMR85]

Sound

False statement $\implies$ cheating prover cannot convince honest verifier.

Complete

True statement $\implies$ honest prover convinces honest verifier.

Zero-knowledge

True statement $\implies$ verifier learns nothing other than statement is true.

slide Y. El Housni

zk-SNARK: Zero-Knowledge Succinct Non-interactive ARgument of Knowledge

"I have a *computationally sound*, *complete*, *zero-knowledge*, **succinct**, **non-interactive** proof that a statement is true and that I know a related secret".

Succinct

A proof is very short and easy to verify.

Non-interactive

No interaction between the prover and verifier for proof generation and verification (except the proof message).

ARgument of Knowledge

Honest verifier is convinced that a computationally bounded prover knows a secret information.

slide Y. El Housni

Preprocessing zk-SNARK for NP language

F: public `NP` program, x, z: public inputs, w: private input (witness)

$z := F(x, w)$

Preprocessing zk-SNARK for NP language

F: public $\mathtt{NP}$ program, x, z: public inputs, w: private input (witness)

$z := F(x, w)$

A zk-SNARK consists of algorithms S, P, V s.t. for a security parameter λ:

$$\begin{array}{llll} \mathit{Setup}: & (pk, vk) & \leftarrow & S(F, 1^{\lambda}) \\ \mathit{Prove}: & \pi & \leftarrow & P(x, z, w, pk) \\ \mathit{Verify}: & \mathtt{false/true} & \leftarrow & V(x, z, \pi, vk) \end{array}$$

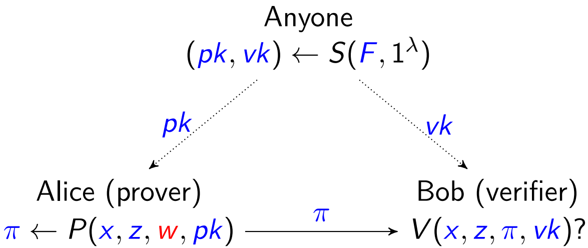
Preprocessing zk-SNARK for NP language

F: public $\texttt{NP}$ program, x, z: public inputs, w: private input (witness)

$z := F(x, w)$

A zk-SNARK consists of algorithms S, P, V s.t. for a security parameter λ:

$$\begin{array}{llll}
\textit{Setup}: & (pk, vk) & \leftarrow & S(F, 1^{\lambda}) \\
\textit{Prove}: & \pi & \leftarrow & P(x, z, w, pk) \\
\textit{Verify}: & \texttt{false/true} & \leftarrow & V(x, z, \pi, vk)
\end{array}$$



Anyone

$(pk, vk) \leftarrow S(F, 1^{\lambda})$

Alice (prover)

$\pi \leftarrow P(x, z, w, pk)$

Bob (verifier)

$V(x, z, \pi, vk)?$

slide Y. El Housni

zk-SNARKs in a nutshell

Main ideas:

zk-SNARKs in a nutshell

Main ideas:

1. Reduce a general statement satisfiability to a polynomial equation satisfiability.

zk-SNARKs in a nutshell

Main ideas:

1. Reduce a general statement satisfiability to a polynomial equation satisfiability.
2. Use Schwartz–Zippel lemma to succinctly verify the polynomial equation with high probability.

zk-SNARKs in a nutshell

Main ideas:

1. Reduce a general statement satisfiability to a polynomial equation satisfiability.
2. Use Schwartz–Zippel lemma to succinctly verify the polynomial equation with high probability.
3. Use homomorphic hiding cryptography to blindly verify the polynomial equation.

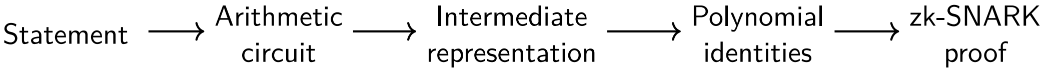
zk-SNARKs in a nutshell

Main ideas:

1. Reduce a general statement satisfiability to a polynomial equation satisfiability.
2. Use Schwartz–Zippel lemma to succinctly verify the polynomial equation with high probability.
3. Use homomorphic hiding cryptography to blindly verify the polynomial equation.
4. Make the protocol non-interactive.

slide Y. El Housni

Data flow

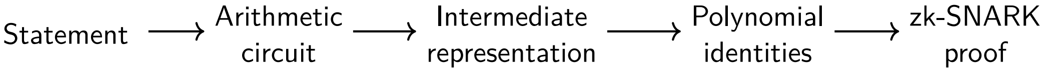


Statement → Arithmetic circuit → Intermediate representation → Polynomial identities → zk-SNARK proof

(knowledge of a
hash function preimage,
a discrete log...)

(QAP: Quadratic
Arithmetic Program)

Data flow



Statement → Arithmetic circuit → Intermediate representation → Polynomial identities → zk-SNARK proof

(knowledge of a
hash function preimage,
a discrete log...)

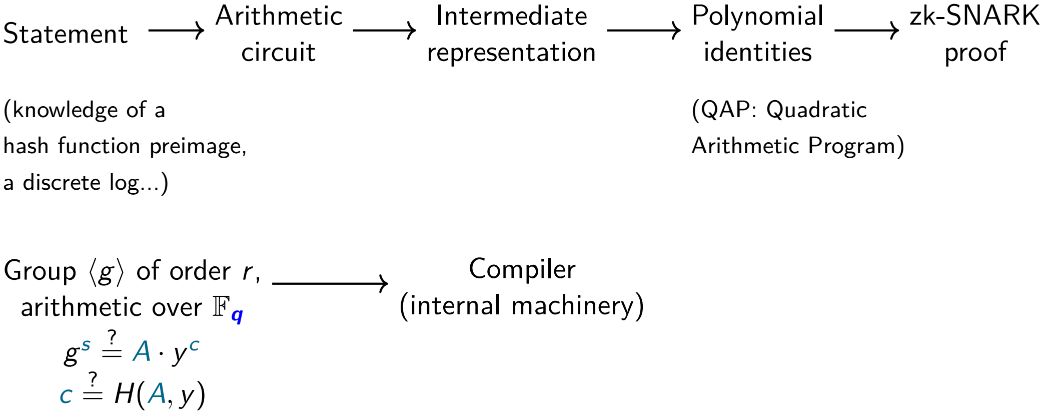
(QAP: Quadratic
Arithmetic Program)

Group $\langle g \rangle$ of order r,
arithmetic over $\mathbb{F}_q$

$$g^s \overset{?}{=} A \cdot y^c$$

$$c \overset{?}{=} H(A, y)$$

Data flow



Statement ⟶ Arithmetic circuit ⟶ Intermediate representation ⟶ Polynomial identities ⟶ zk-SNARK proof

(knowledge of a hash function preimage, a discrete log...)

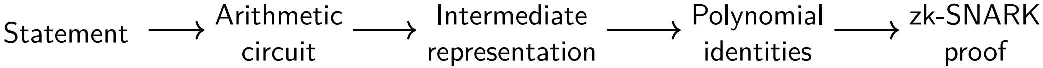
(QAP: Quadratic Arithmetic Program)

Group $\langle g \rangle$ of order r, arithmetic over $\mathbb{F}_{q}$ ⟶ Compiler (internal machinery)

$$g^{s} \overset{?}{=} A \cdot y^{c}$$

$$c \overset{?}{=} H(A, y)$$

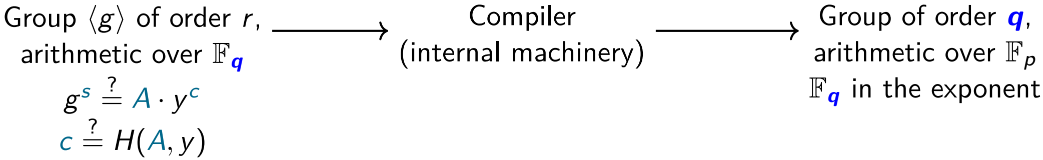
Data flow



Statement → Arithmetic circuit → Intermediate representation → Polynomial identities → zk-SNARK proof

(knowledge of a hash function preimage, a discrete log...)

(QAP: Quadratic Arithmetic Program)



Group $\langle g \rangle$ of order r, arithmetic over $\mathbb{F}_{\boldsymbol{q}}$

$$g^s \stackrel{?}{=} A \cdot y^c$$

$$c \stackrel{?}{=} H(A, y)$$

→ Compiler (internal machinery) →

Group of order $\boldsymbol{q}$, arithmetic over $\mathbb{F}_p$, $\mathbb{F}_{\boldsymbol{q}}$ in the exponent

Outline

zk-SNARK

Elliptic Curves

Pairings

Pairing-friendly curves

SNARK-friendly curves

Elliptic curves in cryptography

- 1985 (published in 1987) Hendrik Lenstra Jr., Elliptic Curve Method (ECM) for integer factoring
- 1985, Koblitz, Miller: Elliptic Curves over a finite field form a group suitable for Diffie–Hellman key exchange
- 1985, Certicom: company owning patents on ECC

Elliptic curves in cryptography

- 1985 (published in 1987) Hendrik Lenstra Jr., Elliptic Curve Method (ECM) for integer factoring
- 1985, Koblitz, Miller: Elliptic Curves over a finite field form a group suitable for Diffie–Hellman key exchange
- 1985, Certicom: company owning patents on ECC
- 2000 Elliptic curves in IEEE P1363 standard
- 2000 Bilinear pairings over elliptic curves
- NSA cipher suite B, elliptic curves for public-key crypto

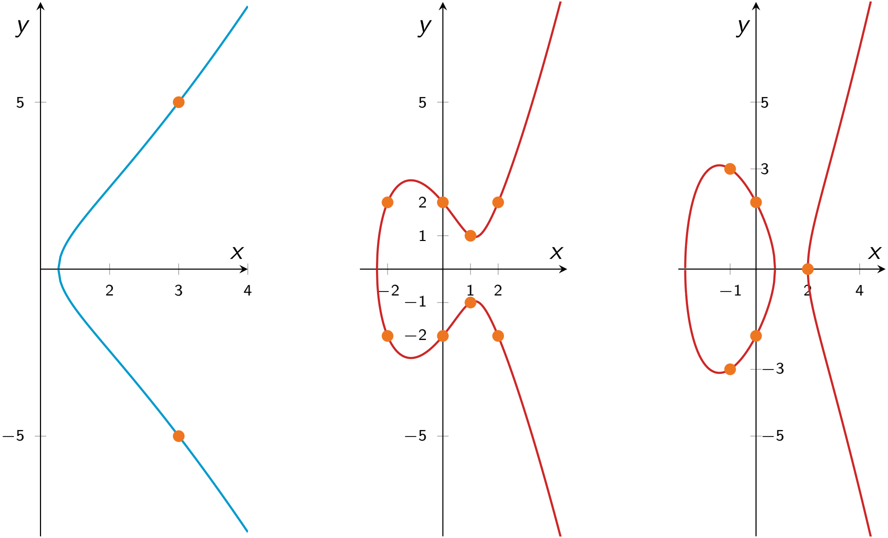
Elliptic curves in cryptography

- 1985 (published in 1987) Hendrik Lenstra Jr., Elliptic Curve Method (ECM) for integer factoring
- 1985, Koblitz, Miller: Elliptic Curves over a finite field form a group suitable for Diffie–Hellman key exchange
- 1985, Certicom: company owning patents on ECC
- 2000 Elliptic curves in IEEE P1363 standard
- 2000 Bilinear pairings over elliptic curves
- NSA cipher suite B, elliptic curves for public-key crypto
- 2014: Quasi-polynomial-time algorithm
  for discrete log computation in GF(2^n), GF(3^m)
  No more pairings on elliptic curves over these fields
- 2015: Tower Number Field Sieve in GF(p^n)
  Pairing-friendly curves should have larger key sizes

Elliptic curves in cryptography

- 1985 (published in 1987) Hendrik Lenstra Jr., Elliptic Curve Method (ECM) for integer factoring
- 1985, Koblitz, Miller: Elliptic Curves over a finite field form a group suitable for Diffie–Hellman key exchange
- 1985, Certicom: company owning patents on ECC
- 2000 Elliptic curves in IEEE P1363 standard
- 2000 Bilinear pairings over elliptic curves
- NSA cipher suite B, elliptic curves for public-key crypto
- 2014: Quasi-polynomial-time algorithm
  for discrete log computation in GF(2^n), GF(3^m)
  No more pairings on elliptic curves over these fields
- 2015: Tower Number Field Sieve in GF(p^n)
  Pairing-friendly curves should have larger key sizes
- 2016: NIST Post-Quantum competition
  Isogenies on elliptic curves

Examples of elliptic curves

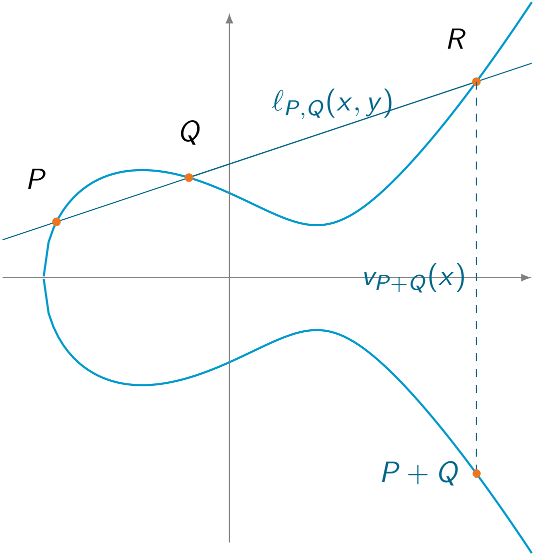


$y^2 = x^3 - 2$

$y^2 = x^3 - 4x + 4$

$y^2 = x^3 - 6x + 4$

Chord and tangent rule



$P(x_1, y_1),\ Q(x_2, y_2),\ x_1 \neq x_2$

slope $\lambda = \dfrac{\Delta y}{\Delta x} = \dfrac{y_2 - y_1}{x_2 - x_1}$

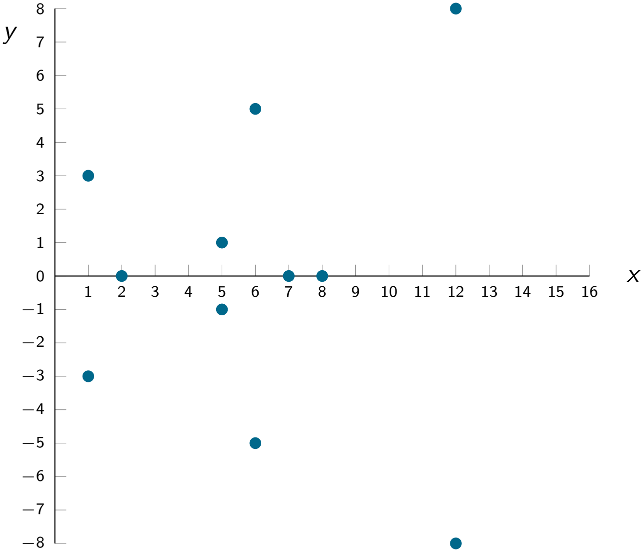
line L through P and Q has equation

$L\colon y = \lambda(x - x_1) + y_1$

$P, Q, R \in L \cap E$

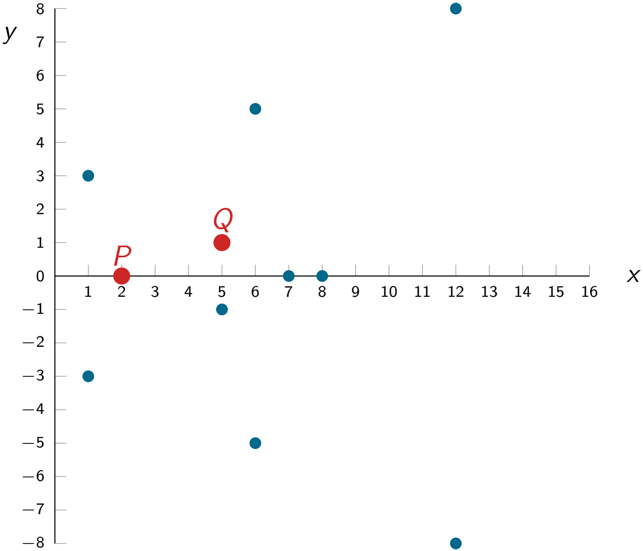
Elliptic curves over finite fields

$$E/\mathbb{F}_{17} : y^2 = x^3 + x + 7$$



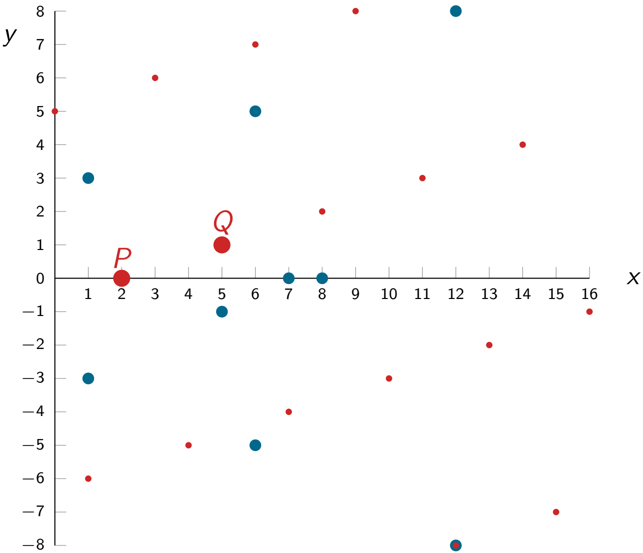
Elliptic curves over finite fields

$$E/\mathbb{F}_{17}: y^2 = x^3 + x + 7$$



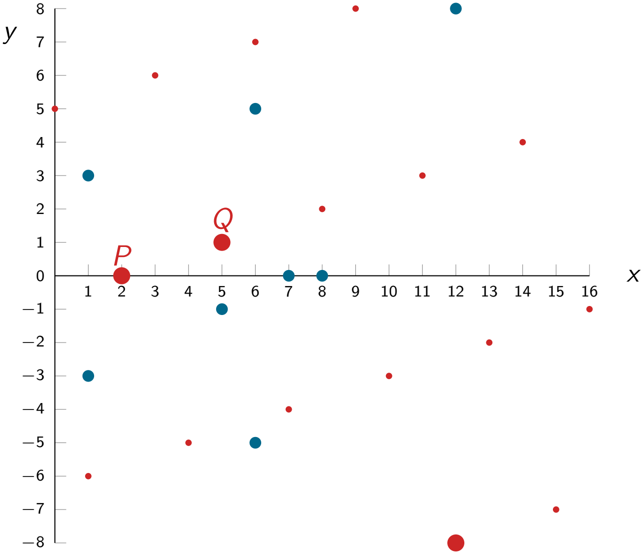
Elliptic curves over finite fields

$$E/\mathbb{F}_{17}\colon y^2 = x^3 + x + 7$$



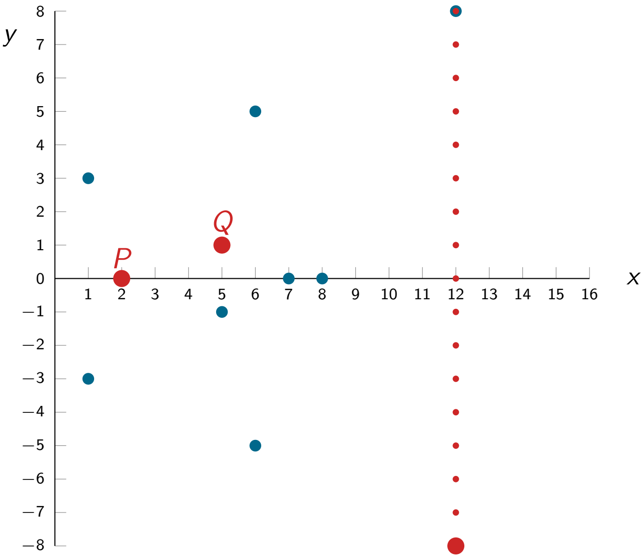
Elliptic curves over finite fields

$$E/\mathbb{F}_{17}\colon y^2 = x^3 + x + 7$$



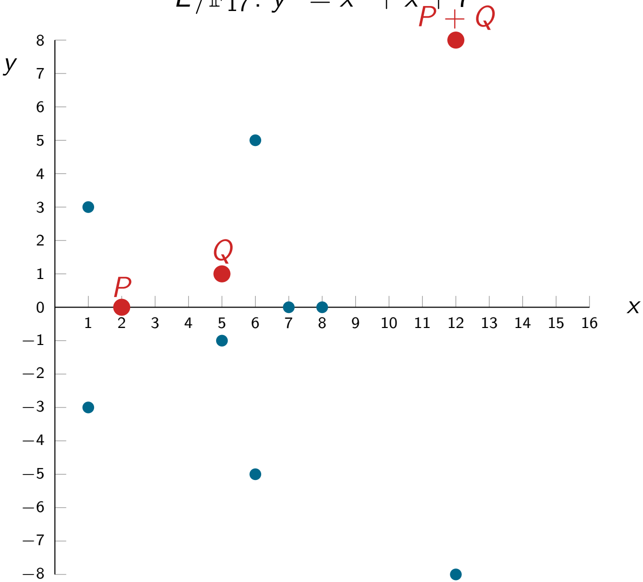
Elliptic curves over finite fields

$$E/\mathbb{F}_{17}\colon y^2 = x^3 + x + 7$$



Elliptic curves over finite fields

$$E/\mathbb{F}_{17}\colon y^2 = x^3 + x + 7$$



Outline

zk-SNARK

Elliptic Curves

Pairings

Pairing-friendly curves

SNARK-friendly curves

What is a pairing?

$(\mathbf{G}_1,+),(\mathbf{G}_2,+),(\mathbf{G}_T,\cdot)$ three cyclic groups of large prime order n
Pairing: map $e : \mathbf{G}_1 \times \mathbf{G}_2 \to \mathbf{G}_T$

1. bilinear: $e(P_1+P_2,Q)=e(P_1,Q)\cdot e(P_2,Q)$, $e(P,Q_1+Q_2)=e(P,Q_1)\cdot e(P,Q_2)$
2. non-degenerate: $e(G_1,G_2)\neq 1$ for $\langle G_1\rangle=\mathbf{G}_1$, $\langle G_2\rangle=\mathbf{G}_2$
3. efficiently computable.

Most often used in practice:

$$e([a]P,[b]Q)=e([b]P,[a]Q)=e(P,Q)^{ab}\ .$$

$\rightsquigarrow$ Many applications in asymmetric cryptography.

Pairing setting: elliptic curves

$$E/\mathbb{F}_p : \ y^2 = x^3 + ax + b, \ a, b \in \mathbb{F}_p, \ p \geq 5$$

- proposed in 1985 by Koblitz, Miller
- $E(\mathbb{F}_p)$ has an efficient group law (chord an tangent rule) → $\mathbf{G}_1$
- $\#E(\mathbb{F}_p) = p + 1 - t$, trace t: $|t| \leq 2\sqrt{p}$
- efficient group order computation (*point counting*)

Pairing setting: elliptic curves

$$E/\mathbb{F}_p : \; y^2 = x^3 + ax + b, \; a, b \in \mathbb{F}_p, \; p \geq 5$$

- proposed in 1985 by Koblitz, Miller
- $E(\mathbb{F}_p)$ has an efficient group law (chord an tangent rule) $\rightarrow \; \mathbf{G}_1$
- $\#E(\mathbb{F}_p) = p + 1 - t$, trace t: $|t| \leq 2\sqrt{p}$
- efficient group order computation (*point counting*)
- large subgroup of prime order n s.t. $n \mid p + 1 - t$ and n coprime to p
- $E(\mathbb{F}_p)[n] = \{P \in E(\mathbb{F}_p) \colon [n]P = \mathcal{O}\}$ has order n
- $E[n] \simeq \mathbb{Z}/n\mathbb{Z} \times \mathbb{Z}/n\mathbb{Z}$ (for crypto)
- only generic attacks against DLP on well-chosen genus 1 and genus 2 curves
- optimal parameter sizes

Tate pairing

From its definition to its efficient implementation

- John Tate, 1958
- Stephen Lichtenbaum, 1969
- Victor Miller, 1986, Miller algorithm for f_P
- Frey–Rück, 1994: the MOV attack with the Tate pairing instead of the Weil pairing
- Harasawa, Shikata, Suzuki, Imai, 1999, 161467 s (112 days)
  163-bit supersingular curve, $\mathbf{G}_T \subset \mathbb{F}_{p^2}$ of 326 bits.
- Antoine Joux, 2000: how to compute Miller algorithm more efficiently
  1 s on a supersingular 528-bit curve, $\mathbf{G}_T \subset \mathbb{F}_{p^2}$ of 1055 bits

Cryptographic pairing

Modified Weil or Tate pairing on an elliptic curve

Discrete logarithm problem with one more dimension.

$$e \; : \; E(\mathbb{F}_p)[n] \times E(\mathbb{F}_{p^k})[n] \longrightarrow \mathbb{F}_{p^k}^*, \quad e([a]P, [b]Q) = e(P, Q)^{ab}$$

Cryptographic pairing

Modified Weil or Tate pairing on an elliptic curve

Discrete logarithm problem with one more dimension.

$$e \ : \ E(\mathbb{F}_p)[n] \times E(\mathbb{F}_{p^k})[n] \longrightarrow \mathbb{F}_{p^k}^*, \quad e([a]P, [b]Q) = e(P, Q)^{ab}$$

Attacks

Cryptographic pairing

Modified Weil or Tate pairing on an elliptic curve

Discrete logarithm problem with one more dimension.

$$e \ : \ E(\mathbb{F}_p)[n] \times E(\mathbb{F}_{p^k})[n] \longrightarrow \mathbb{F}_{p^k}^*, \quad e([a]P, [b]Q) = e(P, Q)^{ab}$$

Attacks

- inversion of e : hard problem (exponential)

Cryptographic pairing

Modified Weil or Tate pairing on an elliptic curve

Discrete logarithm problem with one more dimension.

$$e \;:\; E(\mathbb{F}_p)[n] \times E(\mathbb{F}_{p^k})[n] \longrightarrow \mathbb{F}^*_{p^k}, \quad e([a]P, [b]Q) = e(P, Q)^{ab}$$

Attacks

- inversion of e : hard problem (exponential)
- discrete logarithm computation in $E(\mathbb{F}_p)$: hard problem (exponential, in $O(\sqrt{n})$)

Cryptographic pairing

Modified Weil or Tate pairing on an elliptic curve

Discrete logarithm problem with one more dimension.

$$e \ : \ E(\mathbb{F}_p)[n] \times E(\mathbb{F}_{p^k})[n] \longrightarrow \mathbb{F}_{p^k}^*, \quad e([a]P, [b]Q) = e(P, Q)^{ab}$$

Attacks

- inversion of e : hard problem (exponential)
- discrete logarithm computation in $E(\mathbb{F}_p)$: hard problem (exponential, in $O(\sqrt{n})$)
- discrete logarithm computation in $\mathbb{F}_{p^k}^*$: **easier, subexponential** → take a large enough field

Jens Groth's proof composition [Gro16]

Given an instance $\Phi = (a_0, \ldots, a_\ell) \in \mathbb{F}_r^\ell$ of a public $\texttt{NP}$ program F

- $(pk, vk) \leftarrow S(F, \tau, 1^\lambda)$ where

$$vk = (vk_{\alpha,\beta}, \{vk_{\pi_i}\}_{i=0}^{\ell}, vk_\gamma, vk_\delta) \in \mathbf{G}_T \times \mathbf{G}_1^{\ell+1} \times \mathbf{G}_2 \times \mathbf{G}_2$$

- $\pi \leftarrow P(\Phi, w, pk)$ where

$$\pi = (A, B, C) \in \mathbf{G}_1 \times \mathbf{G}_2 \times \mathbf{G}_1 \qquad (O_\lambda(1))$$

- $0/1 \leftarrow V(\Phi, \pi, vk)$ where V is

$$e(A, B) = vk_{\alpha,\beta} \cdot e(vk_x, vk_\gamma) \cdot e(C, vk_\delta) \qquad (O_\lambda(|\Phi|)) \tag{1}$$

and $vk_x = \sum_{i=0}^{\ell} [a_i] vk_{\pi_i}$ depends only on the instance Φ and $vk_{\alpha,\beta} = e(vk_\alpha, vk_\beta)$ can be computed in the trusted setup for $(vk_\alpha, vk_\beta) \in \mathbf{G}_1 \times \mathbf{G}_2$.

slide Y. El Housni

Applications not in cryptocurrencies

ZK Microphone: Trusted audio in the age of deepfakes

https://ethglobal.com/showcase/zk-microphone-8161v

Proving sound authenticity

Using ZK Proofs to Fight Disinformation

https://iacr.org/submit/files/slides/2023/rwc/rwc2023/13/slides.pdf

Proving image authenticity

A Tool for Proving Software Vulnerabilities in Zero Knowledge

https://galois.com/blog/2024/02/introducing-cheesecloth-a-tool-for-proving-software-vulnerabilities-in-zero-knowledge/

Outline

zk-SNARK

Elliptic Curves

Pairings

Pairing-friendly curves

SNARK-friendly curves

First ordinary pairing-friendly curves: MNT

Miyaji, Nakabayashi, Takano, $\#E(\mathbb{F}_p) = p(u) + 1 - t(u) = q(u)$

$$k = 3 \begin{cases} t(u) = -1 \pm 6u \\ q(u) = 12u^2 \mp 6u + 1 \\ p(u) = 12u^2 - 1 \\ Dy^2 = 12u^2 \pm 12u - 5 \end{cases}$$

$$k = 4 \begin{cases} t(u) = -u,\ \ u + 1 \\ q(u) = u^2 + 2u + 2,\ \ u^2 + 1 \\ p(u) = u^2 + u + 1 \\ Dy^2 = 3u^2 + 4u + 4 \end{cases} \qquad k = 6 \begin{cases} t(u) = 1 \pm 2u \\ q(u) = 4u^2 \mp 2u + 1 \\ p(u) = 4u^2 + 1 \\ Dy^2 = 12u^2 - 4u + 3 \end{cases}$$

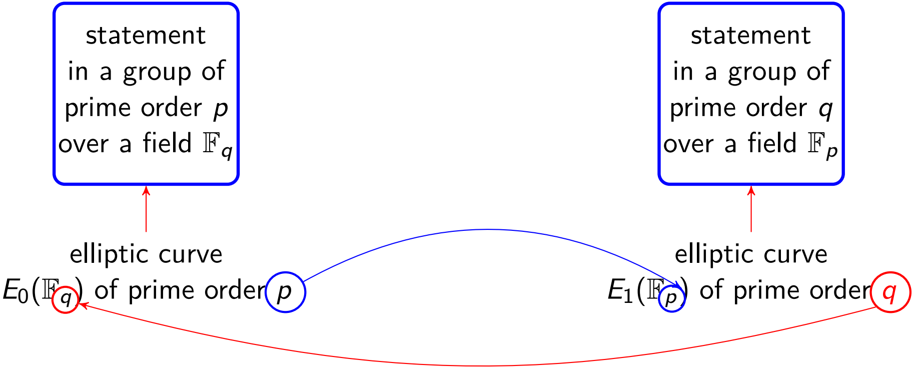
CODA [MS18]:

$k = 6$, 753 bits, $E_6 \approx 137$ bits of security, $D = -241873351932854907$, seed $u =$

`0xaa3a58eb20d1fec36e5e772ee6d3ff28c296465f137300399db8a5521e18d33581a262716214583d3b89820dd0c000`

$k = 4$, 753 bits, $E_4 \approx 113$ bits of security

Cycle of curves: unlimited chains of SNARKs [BCTV14]



statement in a group of prime order p over a field $\mathbb{F}_q$

statement in a group of prime order q over a field $\mathbb{F}_p$

elliptic curve $E_0(\mathbb{F}_q)$ of prime order p

elliptic curve $E_1(\mathbb{F}_p)$ of prime order q

MNT-4 and MNT-6 curves form a cycle

$k = 4$, MNT-4 parameters $\quad t_4 = -v, \quad q_4 = v^2 + 1, \quad p_4 = v^2 + v + 1$
$k = 6$, MNT-6 parameters $\quad t_6 = 1 - 2u, \quad q_6 = 4u^2 + 2u + 1, \quad p_6 = 4u^2 + 1$

$$\begin{array}{c} q_4 = p_6 \\ \text{and} \\ p_4 = q_6 \end{array} \iff \begin{array}{c} v = 2u \\ \text{and} \\ q_4,\ q_6 \text{ are primes} \end{array}$$

Unique known cycle of pairing-friendly curves.
Impossibility results:

Alessandro Chiesa, Lynn Chua, and Matthew Weidner.
On cycles of pairing-friendly elliptic curves.
SIAM Journal on Applied Algebra and Geometry, 3(2):175–192, 2019.

Marta Bellés-Muñoz, Jorge Jiménez Urroz, and Javier Silva.
Revisiting cycles of pairing-friendly elliptic curves.
In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part II*, volume 14082 of *LNCS*, pages 3–37. Springer, Heidelberg, August 2023.

Very popular pairing-friendly curves: Barreto-Naehrig (BN)

$$E_{BN} : \ y^2 = x^3 + b, \ p \equiv 1 \bmod 3, \ D = -3 \text{ (ordinary)}$$

$$\begin{aligned} p &= 36x^4 + 36x^3 + 24x^2 + 6x + 1 \\ t &= 6x^2 + 1 \\ q &= p + 1 - t = 36x^4 + 36x^3 + 18x^2 + 6x + 1 \\ t^2 - 4p &= -3(6x^2 + 4x + 1)^2 \ \rightarrow \text{ no CM method needed} \end{aligned}$$

Comes from the Aurifeuillean factorization of Φ_{12} :

$\Phi_{12}(6x^2) = q(x)q(-x)$

Security level	$\log_2 q$	finite field	k	$\log_2 p$	$\deg P, \ p = P(u)$	ρ
102	256	3072	12	256	4	1
123	384	4608	12	384	4	1
132	448	5376	12	448	4	1

Formerly BN-254 in Euthereum with seed `0x44e992b44a6909f1`

BLS12

Barreto, Lynn, Scott method.
Becomes more and more popular, replacing BN curves

$$E_{\mathrm{BLS}}: \; y^2 = x^3 + b, \; p \equiv 1 \bmod 3, \; D = -3 \text{ (ordinary)}$$

$$\begin{aligned} p &= (u-1)^2/3(u^4 - u^2 + 1) + u \\ t &= u + 1 \\ q &= (u^4 - u^2 + 1) = \Phi_{12}(u) \\ p + 1 - t &= \underbrace{(u-1)^2/3}_{\text{cofactor}}(u^4 - u^2 + 1) \end{aligned}$$

$$t^2 - 4p = -3y(u)^2 \rightarrow \text{no CM method needed}$$

BLS12-381 (Zcash [Bow17]) with seed `-0xd201000000010000`
BLS12-377 (Zexe [BCG+18]) with seed `0x8508c00000000001`

Outline

zk-SNARK

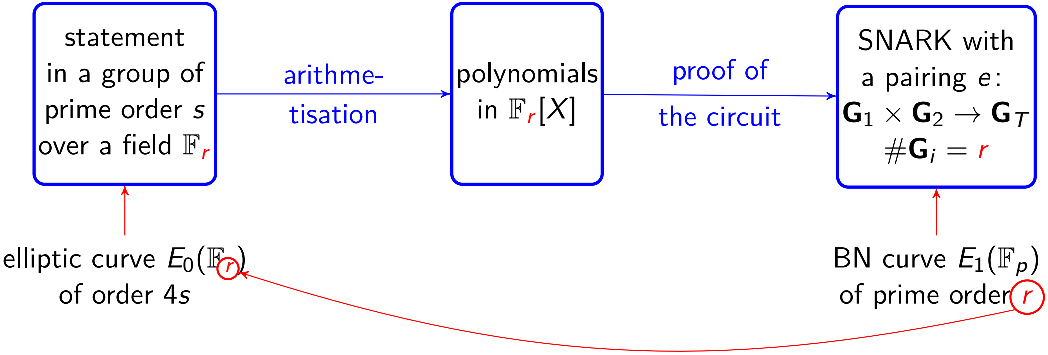
Elliptic Curves

Pairings

Pairing-friendly curves

SNARK-friendly curves

CØCØ embedded curve: Kosba et al. construction [KZM+15]



statement in a group of prime order s over a field $\mathbb{F}_r$

arithme-tisation

polynomials in $\mathbb{F}_r[X]$

proof of the circuit

SNARK with a pairing $e \colon \mathbf{G}_1 \times \mathbf{G}_2 \to \mathbf{G}_T$ $\#\mathbf{G}_i = r$

elliptic curve $E_0(\mathbb{F}_r)$ of order $4s$

BN curve $E_1(\mathbb{F}_p)$ of prime order r

CØCØ: given r, search for a curve E_0 over $\mathbb{F}_r$ of order 4 times a prime

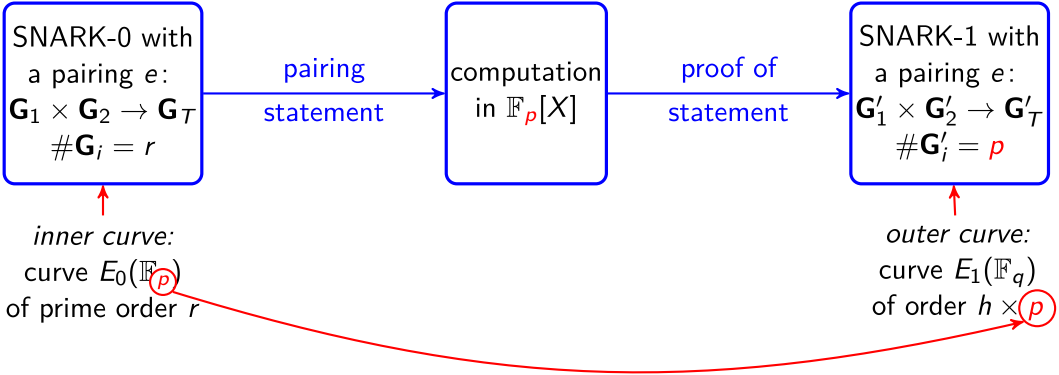
Embedded SNARK-friendly curves

Usually a twist-secure elliptic curve in Montgomery or (twisted) Edwards form

Input: field $\mathbb{F}_p$
Output: an embedded curve of order $4s$ or $8s$ with prime s
Procedure: Increment the curve coefficient(s) until a suitable curve is found

CØCØ [KZM+15] with BN-254a,
JubJub [ZCa21] or Bandersnatch [MSZ21] with BLS12-381, ...

2-chains of elliptic curves



SNARK-0 with a pairing e: $\mathbf{G}_1 \times \mathbf{G}_2 \to \mathbf{G}_T$ $\#\mathbf{G}_i = r$

pairing statement

computation in $\mathbb{F}_p[X]$

proof of statement

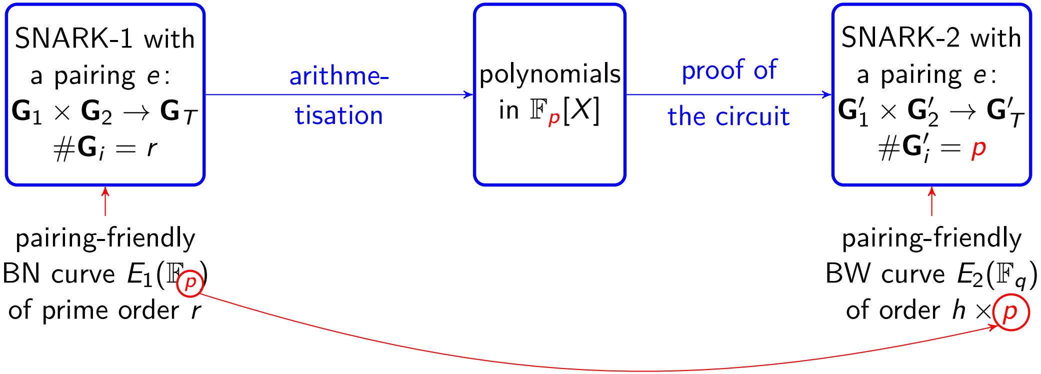
SNARK-1 with a pairing e: $\mathbf{G}'_1 \times \mathbf{G}'_2 \to \mathbf{G}'_T$ $\#\mathbf{G}'_i = p$

inner curve: curve $E_0(\mathbb{F}_p)$ of prime order r

outer curve: curve $E_1(\mathbb{F}_q)$ of order $h \times p$

Given p, search for a pairing-friendly curve E_1 of order $h \cdot p$ over a field $\mathbb{F}_q$

Geppetto construction [CFH$^+$15]



SNARK-1 with a pairing e: $\mathbf{G}_1 \times \mathbf{G}_2 \to \mathbf{G}_T$ $\#\mathbf{G}_i = r$

arithmetisation

polynomials in $\mathbb{F}_p[X]$

proof of the circuit

SNARK-2 with a pairing e: $\mathbf{G}'_1 \times \mathbf{G}'_2 \to \mathbf{G}'_T$ $\#\mathbf{G}'_i = p$

pairing-friendly BN curve $E_1(\mathbb{F}_p)$ of prime order r

pairing-friendly BW curve $E_2(\mathbb{F}_q)$ of order $h \times p$

Geppetto: given p, search for a pairing-friendly curve BW6 (Brezing–Weng) of order $h \cdot p$ over a field $\mathbb{F}_q$

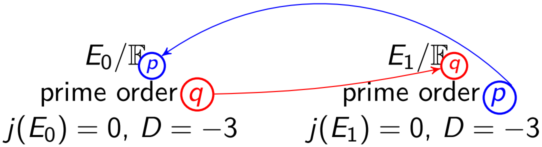
2-chains of pairing-friendly curves

- Geppetto [CFH+15]: BN254b + BW6-509
- Zexe [BCG+18]: BLS12-377 + CP6-782
- BLS12-377 + BW6-761 [EHG20] for Gorth16
- BLS24-315 + BW6-633 [EHG22] For KZG / universal SNARK

More plain/hybrid cycles of curves

Plain cycles: 2 plain prime-order elliptic curves (no pairing)

secp256k1/secq256k1 https://moderncrypto.org/mail-archive/curves/2018/000992.html
HALO: Tweedledum/tweedledee curves https://github.com/daira/tweedle
HALO2: Pallas-Vesta – Pasta curves https://github.com/zcash/pasta_curves



$E_0/\mathbb{F}_p$
prime order q
$j(E_0) = 0,\ D = -3$

$E_1/\mathbb{F}_q$
prime order p
$j(E_1) = 0,\ D = -3$

Hybrid cycles: a plain curve and a BN pairing-friendly curve, both prime order

BN254-Grumpkin https://hackmd.io/@aztec-network/ByzgNxBfd
BN382-plain https://github.com/o1-labs/zexe/tree/master/algebra/src/bn_382
Pluto (BN446) - Eris https://github.com/daira/pluto-eris/

Conclusion

Statement	SNARK 1	SNARK 2
embedded curve	inner curve	outer curve
CØCØ [KZM+15]	BN254a Ethereum	
E_0	BN254b	BW6-509 Geppetto [CFH+15]
Jubjub [ZCa21] Bandersnatch [MSZ21]	BLS12-381 [Bow17]	
E_0'	BLS12-377 [BCG+18]	CP6-782 [BCG+18] BW6-761 [EHG20]
E_0''	BLS24-315	BW6-633 [EHG22]

Survey paper [AEHG23]

Diego F. Aranha, Youssef El Housni, and Aurore Guillevic.
A survey of elliptic curves for proof systems.
Des. Codes Cryptogr., Special Issue: Mathematics of Zero-Knowledge:1–46, December 2022. ePrint 2022/586

Félicitations Jean-Claude et bonne retraite bientôt en Bretagne !

References I

Diego F. Aranha, Youssef El Housni, and Aurore Guillevic.
A survey of elliptic curves for proof systems.
DCC, 91(11):3333–3378, 2023.
Special Issue: Mathematics of Zero-Knowledge.

Nir Bitansky, Ran Canetti, Alessandro Chiesa, and Eran Tromer.
From extractable collision resistance to succinct non-interactive arguments of knowledge, and back again.
In Shafi Goldwasser, editor, *ITCS 2012*, pages 326–349. ACM, January 2012.

Sean Bowe, Alessandro Chiesa, Matthew Green, Ian Miers, Pratyush Mishra, and Howard Wu.
Zexe: Enabling decentralized private computation.
Cryptology ePrint Archive, Report 2018/962, 2018.
https://eprint.iacr.org/2018/962.

Eli Ben-Sasson, Alessandro Chiesa, Eran Tromer, and Madars Virza.
Scalable zero knowledge via cycles of elliptic curves.
In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part II*, volume 8617 of *LNCS*, pages 276–294. Springer, Heidelberg, August 2014.

Manuel Blum, Paul Feldman, and Silvio Micali.
Non-interactive zero-knowledge and its applications (extended abstract).
In *20th ACM STOC*, pages 103–112. ACM Press, May 1988.

Marta Bellés-Muñoz, Jorge Jiménez Urroz, and Javier Silva.
Revisiting cycles of pairing-friendly elliptic curves.
In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part II*, volume 14082 of *LNCS*, pages 3–37. Springer, Heidelberg, August 2023.

References II

Sean Bowe.
BLS12-381: New zk-SNARK elliptic curve construction.
Zcash blog, March 11 2017.
https://electriccoin.co/blog/new-snark-curve/.

Alessandro Chiesa, Lynn Chua, and Matthew Weidner.
On cycles of pairing-friendly elliptic curves.
SIAM Journal on Applied Algebra and Geometry, 3(2):175–192, 2019.

Craig Costello, Cédric Fournet, Jon Howell, Markulf Kohlweiss, Benjamin Kreuter, Michael Naehrig, Bryan Parno, and Samee Zahur.
Geppetto: Versatile verifiable computation.
In *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17-21, 2015*, pages 253–270. IEEE Computer Society, 2015.
ePrint 2014/976.

Alessandro Chiesa, Yuncong Hu, Mary Maller, Pratyush Mishra, Psi Vesely, and Nicholas P. Ward.
Marlin: Preprocessing zkSNARKs with universal and updatable SRS.
In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part I*, volume 12105 of *LNCS*, pages 738–768. Springer, Heidelberg, May 2020.

Youssef El Housni and Aurore Guillevic.
Optimized and secure pairing-friendly elliptic curves suitable for one layer proof composition.
In Stephan Krenn, Haya Shulman, and Serge Vaudenay, editors, *CANS 20*, volume 12579 of *LNCS*, pages 259–279. Springer, Heidelberg, December 2020.

References III

Youssef El Housni and Aurore Guillevic.
Families of SNARK-friendly 2-chains of elliptic curves.
In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022, Part II*, volume 13276 of *LNCS*, pages 367–396. Springer, Heidelberg, May / June 2022.

Rosario Gennaro, Craig Gentry, Bryan Parno, and Mariana Raykova.
Quadratic span programs and succinct NIZKs without PCPs.
In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 626–645. Springer, Heidelberg, May 2013.

Jens Groth, Markulf Kohlweiss, Mary Maller, Sarah Meiklejohn, and Ian Miers.
Updatable and universal common reference strings with applications to zk-SNARKs.
In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 698–728. Springer, Heidelberg, August 2018.

Shafi Goldwasser, Silvio Micali, and Charles Rackoff.
The knowledge complexity of interactive proof-systems (extended abstract).
In *17th ACM STOC*, pages 291–304. ACM Press, May 1985.

Jens Groth.
Short pairing-based non-interactive zero-knowledge arguments.
In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 321–340. Springer, Heidelberg, December 2010.

Jens Groth.
On the size of pairing-based non-interactive arguments.
In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 305–326. Springer, Heidelberg, May 2016.

References IV

Ariel Gabizon, Zachary J. Williamson, and Oana Ciobotaru.
PLONK: Permutations over lagrange-bases for oecumenical noninteractive arguments of knowledge.
Cryptology ePrint Archive, Report 2019/953, 2019.
https://eprint.iacr.org/2019/953.

Joe Kilian.
A note on efficient zero-knowledge proofs and arguments (extended abstract).
In *24th ACM STOC*, pages 723–732. ACM Press, May 1992.

Ahmed Kosba, Zhichao Zhao, Andrew Miller, Yi Qian, Hubert Chan, Charalampos Papamanthou, Rafael Pass, abhi shelat, and Elaine Shi.
C∅c∅: A framework for building composable zero-knowledge proofs.
Cryptology ePrint Archive, Report 2015/1093, 2015.
https://eprint.iacr.org/2015/1093.

Mary Maller, Sean Bowe, Markulf Kohlweiss, and Sarah Meiklejohn.
Sonic: Zero-knowledge SNARKs from linear-size universal and updatable structured reference strings.
In Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz, editors, *ACM CCS 2019*, pages 2111–2128. ACM Press, November 2019.

Silvio Micali.
CS proofs (extended abstracts).
In *35th FOCS*, pages 436–453. IEEE Computer Society Press, November 1994.

Izaak Meckler and Evan Shapiro.
Coda: Decentralized cryptocurrency at scale.
O(1) Labs whitepaper, 2018.
https://cdn.codaprotocol.com/v2/static/coda-whitepaper-05-10-2018-0.pdf.

References V

Simon Masson, Antonio Sanso, and Zhenfei Zhang.
Bandersnatch: a fast elliptic curve built over the BLS12-381 scalar field.
Cryptology ePrint Archive, Report 2021/1152, 2021.
https://eprint.iacr.org/2021/1152.

ZCash.
What is jubjub?
https://z.cash/technology/jubjub/, 2021.