Code-based cryptography and AG codes

Alex Pellegrini

Eindhoven University of Technology

COGNAC 2023

including joint work with Daniel J. Bernstein, Tanja Lange and Alberto Ravagnani

Cryptography

A cipher is a tool that enables *confidential communication* over an *insecure channel*.

Cryptography

A cipher is a tool that enables *confidential communication* over an *insecure channel*.

$$\textbf{Alice} : m \longrightarrow \text{as87t21gd1qw} \longrightarrow m : \textbf{Bob}$$

$$\downarrow$$

$$as87t21gd1qw : \textbf{Eve}$$

Cryptography

A cipher is a tool that enables *confidential communication* over an *insecure channel*.

$$\textbf{Alice} : m \longrightarrow \text{as87t21gd1qw} \longrightarrow m : \textbf{Bob}$$

$$\downarrow$$

$$as87t21gd1qw : \textbf{Eve}$$

Relies on the *hardness* of solving certain math problems.

Cryptography

A cipher is a tool that enables *confidential communication* over an *insecure channel*.

$$\textbf{Alice} : m \longrightarrow \text{as87t21gd1qw} \longrightarrow m : \textbf{Bob}$$

$$\downarrow$$

$$as87t21gd1qw : \textbf{Eve}$$

Relies on the *hardness* of solving certain math problems.
The hard problem can be solved knowing some extra information, called the *key*.

Cryptography

A cipher is a tool that enables *confidential communication* over an *insecure channel*.

$$\textbf{Alice} : m \longrightarrow \text{as87t21gd1qw} \longrightarrow m : \textbf{Bob}$$

$$\downarrow$$

$$as87t21gd1qw : \textbf{Eve}$$

Relies on the *hardness* of solving certain math problems.
The hard problem can be solved knowing some extra information, called the *key*.

- Symmetric : Alice and Bob use the same key.

Cryptography

A cipher is a tool that enables *confidential communication* over an *insecure channel*.

$$\textbf{Alice} : m \longrightarrow \text{as87t21gd1qw} \longrightarrow m : \textbf{Bob}$$

$$\downarrow$$

$$as87t21gd1qw : \textbf{Eve}$$

Relies on the *hardness* of solving certain math problems.
The hard problem can be solved knowing some extra information, called the *key*.

- Symmetric : Alice and Bob use the same key.
- Asymmetric:
  - Alice uses Bob's **public key** to encrypt.
  - Bob uses his **private key** to decrypt.
  - Computationally infeasible to recover private key from public key.

Cryptography - cont.

Currently used asymmetric cryptography:

- Factoring : given product of primes $p_1 \cdot p_2$, find p_1
- DLP: given a group $\mathbb{G}$, $g \in \mathbb{G}$ and g^k with $k \in \mathbb{N}$, find k
- ECDLP: given the group $\mathbb{G}(E)$ of rational points of an elliptic curve E, $P \in \mathbb{G}(E)$ and $kP \in \mathbb{G}(E)$, find k.

Cryptography - cont.

Currently used asymmetric cryptography:

- Factoring : given product of primes $p_1 \cdot p_2$, find p_1
- DLP: given a group $\mathbb{G}$, $g \in \mathbb{G}$ and g^k with $k \in \mathbb{N}$, find k
- ECDLP: given the group $\mathbb{G}(E)$ of rational points of an elliptic curve E, $P \in \mathbb{G}(E)$ and $kP \in \mathbb{G}(E)$, find k.

These can be efficiently solved with a quantum computer (Shor 1994).

Cryptography - cont.

Currently used asymmetric cryptography:

- Factoring : given product of primes $p_1 \cdot p_2$, find p_1
- DLP: given a group $\mathbb{G}$, $g \in \mathbb{G}$ and g^k with $k \in \mathbb{N}$, find k
- ECDLP: given the group $\mathbb{G}(E)$ of rational points of an elliptic curve E, $P \in \mathbb{G}(E)$ and $kP \in \mathbb{G}(E)$, find k.

These can be efficiently solved with a quantum computer (Shor 1994).
Eve can **store today** $as87t21gd1qw$

Cryptography - cont.

Currently used asymmetric cryptography:

- Factoring : given product of primes $p_1 \cdot p_2$, find p_1
- DLP: given a group $\mathbb{G}$, $g \in \mathbb{G}$ and g^k with $k \in \mathbb{N}$, find k
- ECDLP: given the group $\mathbb{G}(E)$ of rational points of an elliptic curve E, $P \in \mathbb{G}(E)$ and $kP \in \mathbb{G}(E)$, find k.

These can be efficiently solved with a quantum computer (Shor 1994).
Eve can **store today** $as87t21gd1qw$ and **decrypt tomorrow**!

Cryptography - cont.

Currently used asymmetric cryptography:

- Factoring : given product of primes $p_1 \cdot p_2$, find p_1
- DLP: given a group $\mathbb{G}$, $g \in \mathbb{G}$ and g^k with $k \in \mathbb{N}$, find k
- ECDLP: given the group $\mathbb{G}(E)$ of rational points of an elliptic curve E, $P \in \mathbb{G}(E)$ and $kP \in \mathbb{G}(E)$, find k.

These can be efficiently solved with a quantum computer (Shor 1994).
Eve can **store today** $as87t21gd1qw$ and **decrypt tomorrow**!

Post-quantum cryptography:

- Codes (this talk)
- Hash based
- Isogeny based
- Lattices
- Multivariate quadratic systems

Coding Theory

Definition

A (linear) **code** C is a k-dimensional subspace of $\mathbb{F}_q^n$. The **minimum distance** of C is defined as $\min_{c \neq 0 \in C} \mathrm{wt}(c)$ where

$$\mathrm{wt}(c) = \#\{i \mid c_i \neq 0\}.$$

Coding Theory

Definition

A (linear) **code** C is a k-dimensional subspace of $\mathbb{F}_q^n$. The **minimum distance** of C is defined as $\min_{c \neq 0 \in C} \mathrm{wt}(c)$ where

$$\mathrm{wt}(c) = \#\{i \mid c_i \neq 0\}.$$

The **correction capability** of a code C having min. distance d is $\left\lfloor \frac{d-1}{2} \right\rfloor$.

Coding Theory

Definition

A (linear) **code** C is a k-dimensional subspace of $\mathbb{F}_q^n$. The **minimum distance** of C is defined as $\min_{c \neq 0 \in C} \mathrm{wt}(c)$ where

$$\mathrm{wt}(c) = \#\{i \mid c_i \neq 0\}.$$

The **correction capability** of a code C having min. distance d is $\left\lfloor \frac{d-1}{2} \right\rfloor$.

We say that a $k \times n$ matrix $\mathcal{G}$ is a **generator matrix** of a code C over $\mathbb{F}_q$ if it has full rank and its rows span C.

Coding Theory

Definition

A (linear) **code** C is a k-dimensional subspace of $\mathbb{F}_q^n$. The **minimum distance** of C is defined as $\min_{c \neq 0 \in C} \mathrm{wt}(c)$ where

$$\mathrm{wt}(c) = \#\{i \mid c_i \neq 0\}.$$

The **correction capability** of a code C having min. distance d is $\left\lfloor \frac{d-1}{2} \right\rfloor$.

We say that a $k \times n$ matrix $\mathcal{G}$ is a **generator matrix** of a code C over $\mathbb{F}_q$ if it has full rank and its rows span C.

Encode a message $m \in \mathbb{F}_q^k$ as $c = m\mathcal{G}$.

Coding Theory

Definition

A (linear) **code** C is a k-dimensional subspace of $\mathbb{F}_q^n$. The **minimum distance** of C is defined as $\min_{c \neq 0 \in C} \mathrm{wt}(c)$ where

$$\mathrm{wt}(c) = \#\{i \mid c_i \neq 0\}.$$

The **correction capability** of a code C having min. distance d is $\left\lfloor \frac{d-1}{2} \right\rfloor$.

We say that a $k \times n$ matrix $\mathcal{G}$ is a **generator matrix** of a code C over $\mathbb{F}_q$ if it has full rank and its rows span C.

Encode a message $m \in \mathbb{F}_q^k$ as $c = m\mathcal{G}$.

Regard noise as a vector addition during transmission. If an error occurs the received data is $r = c + e$ with $e \in \mathbb{F}_q^n$.

Coding Theory

Definition

A (linear) **code** C is a k-dimensional subspace of $\mathbb{F}_q^n$. The **minimum distance** of C is defined as $\min_{c \neq 0 \in C} \mathrm{wt}(c)$ where

$$\mathrm{wt}(c) = \#\{i \mid c_i \neq 0\}.$$

The **correction capability** of a code C having min. distance d is $\left\lfloor \frac{d-1}{2} \right\rfloor$.

We say that a $k \times n$ matrix $\mathcal{G}$ is a **generator matrix** of a code C over $\mathbb{F}_q$ if it has full rank and its rows span C.

Encode a message $m \in \mathbb{F}_q^k$ as $c = m\mathcal{G}$.

Regard noise as a vector addition during transmission. If an error occurs the received data is $r = c + e$ with $e \in \mathbb{F}_q^n$.

A *decoder* $\mathcal{D}$ takes r as input and attempts to find e and return $c' \in C$, where hopefully $c' = c = r - e$.

Code-based cryptography

Hard problem: decode a random linear code.

Code-based cryptography

Hard problem: decode a random linear code.

Idea : Asymmetric ciphers use a code as a secret key and publish a disguised version of the code as a public key.

Code-based cryptography

Hard problem: decode a random linear code.

Idea : Asymmetric ciphers use a code as a secret key and publish a disguised version of the code as a public key.

McEliece (1978):

Code-based cryptography

Hard problem: decode a random linear code.

Idea : Asymmetric ciphers use a code as a secret key and publish a disguised version of the code as a public key.

McEliece (1978):

- Key Generation (Bob) :
  - Pick $\mathcal{G}$ a $k \times n$ matrix over $\mathbb{F}_q$ generating a code C which admits a decoder $\mathcal{D}$ that *efficiently* corrects t errors.
  - Pick random S a $k \times k$ invertible matrix and P an $n \times n$ permutation matrix over $\mathbb{F}_q$.
  - Set $\mathcal{G}' = S\mathcal{G}P$.
  - Private key : $(S, \mathcal{D}, P)$, Public key : $(\mathcal{G}', t)$

Code-based cryptography

Hard problem: decode a random linear code.

Idea : Asymmetric ciphers use a code as a secret key and publish a disguised version of the code as a public key.

McEliece (1978):

- Key Generation (Bob) :
  - Pick $\mathcal{G}$ a $k \times n$ matrix over $\mathbb{F}_q$ generating a code C which admits a decoder $\mathcal{D}$ that *efficiently* corrects t errors.
  - Pick random S a $k \times k$ invertible matrix and P an $n \times n$ permutation matrix over $\mathbb{F}_q$.
  - Set $\mathcal{G}' = S\mathcal{G}P$.
  - Private key : $(S, \mathcal{D}, P)$, Public key : $(\mathcal{G}', t)$

- Encryption (Alice) : $c = m\mathcal{G}' + e$ with e randomly chosen in $\mathbb{F}_q^n$ with $\mathrm{wt}(e) = t$.

Code-based cryptography

Hard problem: decode a random linear code.

Idea : Asymmetric ciphers use a code as a secret key and publish a disguised version of the code as a public key.

McEliece (1978):

- Key Generation (Bob) :
  - Pick $\mathcal{G}$ a $k \times n$ matrix over $\mathbb{F}_q$ generating a code C which admits a decoder $\mathcal{D}$ that *efficiently* corrects t errors.
  - Pick random S a $k \times k$ invertible matrix and P an $n \times n$ permutation matrix over $\mathbb{F}_q$.
  - Set $\mathcal{G}' = S\mathcal{G}P$.
  - Private key : $(S, \mathcal{D}, P)$, Public key : $(\mathcal{G}', t)$
- Encryption (Alice) : $c = m\mathcal{G}' + e$ with e randomly chosen in $\mathbb{F}_q^n$ with $\mathrm{wt}(e) = t$.
- Decryption (Bob) :
  - Compute $cP^{-1} = m\mathcal{G}'P^{-1} + eP^{-1} = mS\mathcal{G} + eP^{-1}$
  - Decode and get $mS = \mathcal{D}(cP^{-1})$.
  - $m = mSS^{-1}$

Code-based crypto - cont.

Pro : Short ciphertext.
Con : Large public key.

Code-based crypto - cont.

Pro : Short ciphertext.
Con : Large public key.

Attacks:

- *Structural attack*: Find $\mathcal{G}$ given $\mathcal{G}'$.
- *Generic attack*: Try to decode in $\mathcal{G}'$, i.e. decrypt a given ciphertext in a random-looking code. The best attacks rely on *information-set decoding*. They have **exponential** computational cost.

Code-based crypto - cont.

Pro : Short ciphertext.
Con : Large public key.

Attacks:

- *Structural attack*: Find $\mathcal{G}$ given $\mathcal{G}'$.
- *Generic attack*: Try to decode in $\mathcal{G}'$, i.e. decrypt a given ciphertext in a random-looking code. The best attacks rely on *information-set decoding*. They have **exponential** computational cost.

Aim: Reduce the key size for a given security parameter (computational cost of best algorithm), or equivalently obtain higher security for a given key size (while avoiding attacks).

Goppa codes

Need a code with an algebraic structure that admits efficient decoders and avoids structural attacks.

Goppa codes

Need a code with an algebraic structure that admits efficient decoders and avoids structural attacks.

Definition

Let $\alpha = \{\alpha_1, \ldots, \alpha_n\} \subseteq \mathbb{F}_{q^\ell}$. A **Reed-Solomon** (RS) code is a code over $\mathbb{F}_{q^\ell}$ defined as

$$RS(\alpha, k) = \left\{(f(\alpha_1), \ldots, f(\alpha_n)) \mid f \in \mathbb{F}_{q^\ell}[x]_{<k}\right\}$$

Goppa codes

Need a code with an algebraic structure that admits efficient decoders and avoids structural attacks.

Definition

Let $\alpha = \{\alpha_1, \ldots, \alpha_n\} \subseteq \mathbb{F}_{q^\ell}$. A **Reed-Solomon** (RS) code is a code over $\mathbb{F}_{q^\ell}$ defined as

$$RS(\alpha, k) = \left\{(f(\alpha_1), \ldots, f(\alpha_n)) \mid f \in \mathbb{F}_{q^\ell}[x]_{<k}\right\}$$

We have fast decoders for RS codes.

Goppa codes

Need a code with an algebraic structure that admits efficient decoders and avoids structural attacks.

Definition

Let $\alpha = \{\alpha_1, \ldots, \alpha_n\} \subseteq \mathbb{F}_{q^\ell}$. A **Reed-Solomon** (RS) code is a code over $\mathbb{F}_{q^\ell}$ defined as

$$RS(\alpha, k) = \left\{(f(\alpha_1), \ldots, f(\alpha_n)) \mid f \in \mathbb{F}_{q^\ell}[x]_{<k}\right\}$$

We have fast decoders for RS codes. There are effective structural attacks.

Goppa codes

Need a code with an algebraic structure that admits efficient decoders and avoids structural attacks.

Definition

Let $\alpha = \{\alpha_1, \ldots, \alpha_n\} \subseteq \mathbb{F}_{q^\ell}$. A **Reed-Solomon** (RS) code is a code over $\mathbb{F}_{q^\ell}$ defined as

$$RS(\alpha, k) = \left\{(f(\alpha_1), \ldots, f(\alpha_n)) \mid f \in \mathbb{F}_{q^\ell}[x]_{<k}\right\}$$

We have fast decoders for RS codes.There are effective structural attacks.

Definition

Let $g(x) \in \mathbb{F}_{q^\ell}[x]$ irreducible with $\deg g = t$ and $g(\alpha_i) \neq 0$. A **Goppa code** over $\mathbb{F}_q$ is defined as

$$\Gamma(\alpha, g) = \left\{(c_1, \ldots, c_n) \in \mathbb{F}_q^n \mid \sum \frac{c_i}{x - \alpha_i} \equiv 0 \mod g(x)\right\}$$

Goppa codes in McEliece

Theorem

Set $h = \prod(x - \alpha_i)$ *and let* $\beta = (g/h'(\alpha_1), \ldots, g/h'(\alpha_n))$. *Then* $\Gamma(\alpha, g)$ *is the subfield-subcode:*

$$\Gamma(\alpha, g) = \beta RS(\alpha, n - t) \cap \mathbb{F}_q^n.$$

Goppa codes in McEliece

Theorem

Set $h = \prod(x - \alpha_i)$ *and let* $\beta = (g/h'(\alpha_1), \ldots, g/h'(\alpha_n))$. *Then* $\Gamma(\alpha, g)$ *is the subfield-subcode:*

$$\Gamma(\alpha, g) = \beta RS(\alpha, n - t) \cap \mathbb{F}_q^n.$$

RS decoders can be used to decode their subfield-subcodes

Goppa codes in McEliece

Theorem

Set $h = \prod(x - \alpha_i)$ *and let* $\beta = (g/h'(\alpha_1), \ldots, g/h'(\alpha_n))$. *Then* $\Gamma(\alpha, g)$ *is the subfield-subcode:*

$$\Gamma(\alpha, g) = \beta RS(\alpha, n - t) \cap \mathbb{F}_q^n.$$

RS decoders can be used to decode their subfield-subcodes $\Rightarrow$ Goppa codes have fast decoders.

Goppa codes in McEliece

Theorem

Set $h = \prod(x - \alpha_i)$ and let $\beta = (g/h'(\alpha_1), \ldots, g/h'(\alpha_n))$. Then $\Gamma(\alpha, g)$ is the subfield-subcode:

$$\Gamma(\alpha, g) = \beta RS(\alpha, n - t) \cap \mathbb{F}_q^n.$$

RS decoders can be used to decode their subfield-subcodes $\Rightarrow$ Goppa codes have fast decoders.
There are no known structural attacks against McEliece using $\Gamma(\alpha, g)$.

Original McEliece proposal uses $\Gamma(\alpha, g)$ over $\mathbb{F}_2$.

Goppa codes in McEliece

Theorem

Set $h = \prod(x - \alpha_i)$ and let $\beta = (g/h'(\alpha_1), \ldots, g/h'(\alpha_n))$. Then $\Gamma(\alpha, g)$ is the subfield-subcode:

$$\Gamma(\alpha, g) = \beta RS(\alpha, n - t) \cap \mathbb{F}_q^n.$$

RS decoders can be used to decode their subfield-subcodes $\Rightarrow$ Goppa codes have fast decoders.
There are no known structural attacks against McEliece using $\Gamma(\alpha, g)$.

Original McEliece proposal uses $\Gamma(\alpha, g)$ over $\mathbb{F}_2$.

Pros:

- small ciphertext
- fast decoder
- good bound on min. distance
- resist more than 40 years of cryptanalysys!

Goppa codes in McEliece

Theorem

Set $h = \prod(x - \alpha_i)$ *and let* $\beta = (g/h'(\alpha_1), \ldots, g/h'(\alpha_n))$. *Then* $\Gamma(\alpha, g)$ *is the subfield-subcode:*

$$\Gamma(\alpha, g) = \beta RS(\alpha, n - t) \cap \mathbb{F}_q^n.$$

RS decoders can be used to decode their subfield-subcodes $\Rightarrow$ Goppa codes have fast decoders.
There are no known structural attacks against McEliece using $\Gamma(\alpha, g)$.

Original McEliece proposal uses $\Gamma(\alpha, g)$ over $\mathbb{F}_2$.

Pros:

- small ciphertext
- fast decoder
- good bound on min. distance
- resist more than 40 years of cryptanalysys!

Cons:

- large public keys

Algebraic geometry (AG) codes

Definition

Let $\mathcal{X}$ be an algebraic curve over $\mathbb{F}_{q^\ell}$, $P_1, \ldots, P_n \in \mathcal{X}$ a set of $\mathbb{F}_{q^\ell}$-points. Let $D = P_1 + \cdots + P_n$ be a divisor on $\mathcal{X}$ and G be another divisor s.t. $supp(G) \cap supp(D) = \emptyset$. The AG code on $\mathcal{X}$ defined by D and G is

$$C_{\mathcal{X}}(D, G) = \{(f(P_1), \ldots, f(P_n)) \mid f \in \mathcal{L}(G)\}$$

where $\mathcal{L}(G)$ is the Riemann-Roch space of G.

Algebraic geometry (AG) codes

Definition

Let $\mathcal{X}$ be an algebraic curve over $\mathbb{F}_{q^\ell}$, $P_1, \ldots, P_n \in \mathcal{X}$ a set of $\mathbb{F}_{q^\ell}$-points. Let $D = P_1 + \cdots + P_n$ be a divisor on $\mathcal{X}$ and G be another divisor s.t. $supp(G) \cap supp(D) = \emptyset$. The AG code on $\mathcal{X}$ defined by D and G is

$$C_{\mathcal{X}}(D, G) = \{(f(P_1), \ldots, f(P_n)) \mid f \in \mathcal{L}(G)\}$$

where $\mathcal{L}(G)$ is the Riemann-Roch space of G.

We have decoders for AG codes.

Algebraic geometry (AG) codes

Definition

Let $\mathcal{X}$ be an algebraic curve over $\mathbb{F}_{q^\ell}$, $P_1, \ldots, P_n \in \mathcal{X}$ a set of $\mathbb{F}_{q^\ell}$-points. Let $D = P_1 + \cdots + P_n$ be a divisor on $\mathcal{X}$ and G be another divisor s.t. $supp(G) \cap supp(D) = \emptyset$. The AG code on $\mathcal{X}$ defined by D and G is

$$C_{\mathcal{X}}(D, G) = \{(f(P_1), \ldots, f(P_n)) \mid f \in \mathcal{L}(G)\}$$

where $\mathcal{L}(G)$ is the Riemann-Roch space of G.

We have decoders for AG codes. Fast decoders for certain AG codes.

Algebraic geometry (AG) codes

Definition

Let $\mathcal{X}$ be an algebraic curve over $\mathbb{F}_{q^\ell}$, $P_1, \ldots, P_n \in \mathcal{X}$ a set of $\mathbb{F}_{q^\ell}$-points. Let $D = P_1 + \cdots + P_n$ be a divisor on $\mathcal{X}$ and G be another divisor s.t. $supp(G) \cap supp(D) = \emptyset$. The AG code on $\mathcal{X}$ defined by D and G is

$$C_{\mathcal{X}}(D, G) = \{(f(P_1), \ldots, f(P_n)) \mid f \in \mathcal{L}(G)\}$$

where $\mathcal{L}(G)$ is the Riemann-Roch space of G.

We have decoders for AG codes. Fast decoders for certain AG codes.

There exist polynomial time structural attacks against McEliece using $C_{\mathcal{X}}(D, G)$.

Algebraic geometry (AG) codes

Definition

Let $\mathcal{X}$ be an algebraic curve over $\mathbb{F}_{q^\ell}$, $P_1, \ldots, P_n \in \mathcal{X}$ a set of $\mathbb{F}_{q^\ell}$-points. Let $D = P_1 + \cdots + P_n$ be a divisor on $\mathcal{X}$ and G be another divisor s.t. $supp(G) \cap supp(D) = \emptyset$. The AG code on $\mathcal{X}$ defined by D and G is

$$C_{\mathcal{X}}(D, G) = \{(f(P_1), \ldots, f(P_n)) \mid f \in \mathcal{L}(G)\}$$

where $\mathcal{L}(G)$ is the Riemann-Roch space of G.

We have decoders for AG codes. Fast decoders for certain AG codes.

There exist polynomial time structural attacks against McEliece using $C_{\mathcal{X}}(D, G)$.

None of these attacks affects *subfield-subcodes of AG codes*.

Algebraic geometry (AG) codes

Definition

Let $\mathcal{X}$ be an algebraic curve over $\mathbb{F}_{q^\ell}$, $P_1, \ldots, P_n \in \mathcal{X}$ a set of $\mathbb{F}_{q^\ell}$-points. Let $D = P_1 + \cdots + P_n$ be a divisor on $\mathcal{X}$ and G be another divisor s.t. $supp(G) \cap supp(D) = \emptyset$. The AG code on $\mathcal{X}$ defined by D and G is

$$C_{\mathcal{X}}(D, G) = \{(f(P_1), \ldots, f(P_n)) \mid f \in \mathcal{L}(G)\}$$

where $\mathcal{L}(G)$ is the Riemann-Roch space of G.

We have decoders for AG codes. Fast decoders for certain AG codes.

There exist polynomial time structural attacks against McEliece using $C_{\mathcal{X}}(D, G)$.

None of these attacks affects *subfield-subcodes of AG codes*.

Janwa & Moreno proposed (1996) McEliece cryptosystem using subfield-subcodes of AG codes, achieving smaller public keys than classic McEliece.

Jacobian Goppa codes

We define a code $\Gamma_{\mathcal{X}}(D, G)$ which is always a subcode of a subfield-subcode of an AG code.

Jacobian Goppa codes

We define a code $\Gamma_{\mathcal{X}}(D, G)$ which is always a subcode of a subfield-subcode of an AG code.
For certain parameters it is a subfield-subcode of an AG code. ($\Rightarrow$ no known structural attacks)

Jacobian Goppa codes

We define a code $\Gamma_{\mathcal{X}}(D, G)$ which is always a subcode of a subfield-subcode of an AG code.
For certain parameters it is a subfield-subcode of an AG code. ($\Rightarrow$ no known structural attacks)

Definition

Let $\mathcal{X}$ be an algebraic curve. For a certain shape of G, a **Jacobian Goppa code** is defined as

$$\Gamma_{\mathcal{X}}(D, G) = \left\{ (c_1, \ldots, c_n) \in \mathbb{F}_q^n \mid \sum c_i \varepsilon_i \in \Omega(G - D) \right\}$$

where $\Omega(G - D)$ is the space of differentials of $G - D$ and ε_i are differentials on $\mathcal{X}$ generating $\Omega(-D)$.

For $\mathcal{X} = \mathbb{P}^1(\mathbb{F}_{q^\ell})$ we have that $\Gamma_{\mathcal{X}}(D, G)$ is a Goppa code of the form $\Gamma(\alpha, g)$.

Jacobian Goppa codes

We define a code $\Gamma_{\mathcal{X}}(D, G)$ which is always a subcode of a subfield-subcode of an AG code.
For certain parameters it is a subfield-subcode of an AG code. ($\Rightarrow$ no known structural attacks)

Definition

Let $\mathcal{X}$ be an algebraic curve. For a certain shape of G, a **Jacobian Goppa code** is defined as

$$\Gamma_{\mathcal{X}}(D, G) = \left\{ (c_1, \ldots, c_n) \in \mathbb{F}_q^n \mid \sum c_i \varepsilon_i \in \Omega(G - D) \right\}$$

where $\Omega(G - D)$ is the space of differentials of $G - D$ and ε_i are differentials on $\mathcal{X}$ generating $\Omega(-D)$.

For $\mathcal{X} = \mathbb{P}^1(\mathbb{F}_{q^\ell})$ we have that $\Gamma_{\mathcal{X}}(D, G)$ is a Goppa code of the form $\Gamma(\alpha, g)$.

Improved bounds on the minimum distance $\Rightarrow$ smaller public keys than Janwa & Moreno

Jacobian Goppa codes

We define a code $\Gamma_{\mathcal{X}}(D, G)$ which is always a subcode of a subfield-subcode of an AG code.
For certain parameters it is a subfield-subcode of an AG code. ($\Rightarrow$ no known structural attacks)

Definition

Let $\mathcal{X}$ be an algebraic curve. For a certain shape of G, a **Jacobian Goppa code** is defined as

$$\Gamma_{\mathcal{X}}(D, G) = \left\{ (c_1, \ldots, c_n) \in \mathbb{F}_q^n \mid \sum c_i \varepsilon_i \in \Omega(G - D) \right\}$$

where $\Omega(G - D)$ is the space of differentials of $G - D$ and ε_i are differentials on $\mathcal{X}$ generating $\Omega(-D)$.

For $\mathcal{X} = \mathbb{P}^1(\mathbb{F}_{q^\ell})$ we have that $\Gamma_{\mathcal{X}}(D, G)$ is a Goppa code of the form $\Gamma(\alpha, g)$.

Improved bounds on the minimum distance $\Rightarrow$ smaller public keys than Janwa & Moreno

Theorem

The code $\Gamma_{\mathcal{X}}(D, G)$ can be uniquely represented by two rational functions $h, \gamma \in \mathcal{O}$, where $\mathcal{O}$ is a subring of the fuction field of $\mathcal{X}$ defined by D and the Jacobian $Jac(\mathcal{X})$. Denote $\Gamma_{\mathcal{X}}(h, \gamma)$ the code $\Gamma_{\mathcal{X}}(D, G)$.

McEliece + Jacobian Goppa codes : Key generation

Definition

Given a finite field $\mathbb{F}_{q^\ell}$, an integer $g \geq 0$ and a curve $\mathcal{X}$ of genus g, $\Gamma_{\mathcal{X}}$ is the **family of Jacobian Goppa codes** on $\mathcal{X}$ parametrized by $h, \gamma \in \mathcal{O}$.

[1] For applications the Niederreiter version of McEliece is used along with a permutation of $P_1, \ldots, P_n$ instead of matrices S and P.

McEliece + Jacobian Goppa codes : Key generation

Definition

Given a finite field $\mathbb{F}_{q^\ell}$, an integer $g \geq 0$ and a curve $\mathcal{X}$ of genus g, $\Gamma_{\mathcal{X}}$ is the **family of Jacobian Goppa codes** on $\mathcal{X}$ parametrized by $h, \gamma \in \mathcal{O}$.

Pick $P_1, \ldots, P_n$ rational points on $\mathcal{X}$ and compute h and secret γ.

[1] For applications the Niederreiter version of McEliece is used along with a permutation of $P_1, \ldots, P_n$ instead of matrices S and P.

McEliece + Jacobian Goppa codes : Key generation

Definition

Given a finite field $\mathbb{F}_{q^\ell}$, an integer $g \geq 0$ and a curve $\mathcal{X}$ of genus g, $\Gamma_{\mathcal{X}}$ is the **family of Jacobian Goppa codes** on $\mathcal{X}$ parametrized by $h, \gamma \in \mathcal{O}$.

Pick $P_1, \ldots, P_n$ rational points on $\mathcal{X}$ and compute h and secret γ.
Compute the generator matrix $\mathcal{G}$ of the code $\Gamma_{\mathcal{X}}(h, \gamma) \in \Gamma_{\mathcal{X}}$.

[1] For applications the Niederreiter version of McEliece is used along with a permutation of $P_1, \ldots, P_n$ instead of matrices S and P.

McEliece + Jacobian Goppa codes : Key generation

Definition

Given a finite field $\mathbb{F}_{q^\ell}$, an integer $g \geq 0$ and a curve $\mathcal{X}$ of genus g, $\Gamma_{\mathcal{X}}$ is the **family of Jacobian Goppa codes** on $\mathcal{X}$ parametrized by $h, \gamma \in \mathcal{O}$.

Pick $P_1, \ldots, P_n$ rational points on $\mathcal{X}$ and compute h and secret γ.
Compute the generator matrix $\mathcal{G}$ of the code $\Gamma_{\mathcal{X}}(h, \gamma) \in \Gamma_{\mathcal{X}}$.
Randomly pick S and P and publish $\mathcal{G}' = S\mathcal{G}P$.[1]

[1] For applications the Niederreiter version of McEliece is used along with a permutation of $P_1, \ldots, P_n$ instead of matrices S and P.

McEliece + Jacobian Goppa codes : Key generation

Definition

Given a finite field $\mathbb{F}_{q^\ell}$, an integer $g \geq 0$ and a curve $\mathcal{X}$ of genus g, $\Gamma_{\mathcal{X}}$ is the **family of Jacobian Goppa codes** on $\mathcal{X}$ parametrized by $h, \gamma \in \mathcal{O}$.

Pick $P_1, \ldots, P_n$ rational points on $\mathcal{X}$ and compute h and secret γ.
Compute the generator matrix $\mathcal{G}$ of the code $\Gamma_{\mathcal{X}}(h, \gamma) \in \Gamma_{\mathcal{X}}$.
Randomly pick S and P and publish $\mathcal{G}' = S\mathcal{G}P$.[1]

ALGORITHMS:

- We gave algorithms to compute h by simple arithmetic in the Jacobian $Jac(\mathcal{X})$ for a large family of curves. Work In Progress: Develop efficient software.
- Work In Progress: Devise an efficient algorithm to randomly draw γ.

[1] For applications the Niederreiter version of McEliece is used along with a permutation of $P_1, \ldots, P_n$ instead of matrices S and P.

McEliece + Jacobian Goppa codes : Decryption

Theorem

There exists $\beta = (\beta_1, \ldots, \beta_n) \in \mathbb{F}_{q^\ell}^n$, *depending only on* h *and* γ, *such that*

$$\Gamma_{\mathcal{X}}(h, \gamma) = \Gamma_{\mathcal{X}}(D, G) = \beta C_{\mathcal{X}}(D, E) \cap \mathbb{F}_q^n$$

where E *is a divisor on* $\mathcal{X}$ *related to* h *and* γ.

McEliece + Jacobian Goppa codes : Decryption

Theorem

There exists $\beta = (\beta_1, \ldots, \beta_n) \in \mathbb{F}_{q^\ell}^n$, *depending only on* h *and* γ, *such that*

$$\Gamma_{\mathcal{X}}(h, \gamma) = \Gamma_{\mathcal{X}}(D, G) = \beta C_{\mathcal{X}}(D, E) \cap \mathbb{F}_q^n$$

where E *is a divisor on* $\mathcal{X}$ *related to* h *and* γ.

The code $\Gamma_{\mathcal{X}}(h, \gamma)$ can be decoded using any AG decoder.

McEliece + Jacobian Goppa codes : Decryption

Theorem

There exists $\beta = (\beta_1, \ldots, \beta_n) \in \mathbb{F}_{q^\ell}^n$, *depending only on* h *and* γ, *such that*

$$\Gamma_{\mathcal{X}}(h, \gamma) = \Gamma_{\mathcal{X}}(D, G) = \beta C_{\mathcal{X}}(D, E) \cap \mathbb{F}_q^n$$

where E *is a divisor on* $\mathcal{X}$ *related to* h *and* γ.

The code $\Gamma_{\mathcal{X}}(h, \gamma)$ can be decoded using any AG decoder.

Theorem (Decoding Theorem)

Let $C_{\mathcal{X}}(D, E)$ *be an AG code with correction capability* t. *Let* c *be the sent codeword and* $r \in \mathbb{F}_{q^\ell}^n$ *be such that* $wt(r - c) \leq t$, $R \in \mathcal{O}$ *be such that* $R(P_i) = r_i$ *and* $I := P_1 \cdots P_n \subseteq \mathcal{O}$ *is a proper ideal of* $\mathcal{O}$. *We can decode* r *by solving the modular equation*

$$\sigma R \equiv \varphi \mod I,$$

with $\varphi/\sigma \in \mathcal{L}(E)$. *We can do this in polynomial time.*

McEliece + Jacobian Goppa codes : Decryption

Theorem

There exists $\beta = (\beta_1, \ldots, \beta_n) \in \mathbb{F}_{q^\ell}^n$, *depending only on* h *and* γ, *such that*

$$\Gamma_{\mathcal{X}}(h, \gamma) = \Gamma_{\mathcal{X}}(D, G) = \beta C_{\mathcal{X}}(D, E) \cap \mathbb{F}_q^n$$

where E *is a divisor on* $\mathcal{X}$ *related to* h *and* γ.

The code $\Gamma_{\mathcal{X}}(h, \gamma)$ can be decoded using any AG decoder.

Theorem (Decoding Theorem)

Let $C_{\mathcal{X}}(D, E)$ *be an AG code with correction capability* t. *Let* c *be the sent codeword and* $r \in \mathbb{F}_{q^\ell}^n$ *be such that* $wt(r - c) \leq t$, $R \in \mathcal{O}$ *be such that* $R(P_i) = r_i$ *and* $I := P_1 \cdots P_n \subseteq \mathcal{O}$ *is a proper ideal of* $\mathcal{O}$. *We can decode* r *by solving the modular equation*

$$\sigma R \equiv \varphi \mod I,$$

with $\varphi/\sigma \in \mathcal{L}(E)$. *We can do this in polynomial time.*

ALGORITHMS:

- The Decoding Theorem generalizes to AG codes an efficient decoder for RS code (Gao decoder). Work In Progress: Improve complexity of this decoder.

Wrap-up

Completed Work:

- extend the algebraic structure of Goppa codes to varieties of genus ≥ 0.
- allow either one-point and multi-point Jacobian Goppa codes.
- give improved bounds and efficient methods for computing objects.

Wrap-up

Completed Work:

- extend the algebraic structure of Goppa codes to varieties of genus ≥ 0.
- allow either one-point and multi-point Jacobian Goppa codes.
- give improved bounds and efficient methods for computing objects.

Ongoing Work:

- Improve/devise efficient algorithms.
- Use list-decoding to increase the number of errors that can be introduced in the ciphertext.

Wrap-up

Completed Work:

- extend the algebraic structure of Goppa codes to varieties of genus ≥ 0.
- allow either one-point and multi-point Jacobian Goppa codes.
- give improved bounds and efficient methods for computing objects.

Ongoing Work:

- Improve/devise efficient algorithms.
- Use list-decoding to increase the number of errors that can be introduced in the ciphertext.

Thanks for your attention!