SNARKs over any field

Thomas Piellard, Consensys

TABLE OF CONTENTS

1. **SNARKs and FFTs**

2. **FFT, geometric point of view**
   - geometric point of view
   - main issue

3. **Elliptic curves**
   - isogenies
   - issue of the doubling map
   - low degree isogenies

State of the art

In the most compact proof systems, like Groth16 or polynomials IOP like PLONK, the circuits reason on variables that live in $\mathbb{F}_p$.

- In Groth16, $\mathbb{F}_p^*$ must contain a large power of two subgroup, and must correspond to a subgroup of a pairing-friendly elliptic curve
- In Plonk, instantiated with FRI for instance, $\mathbb{F}_p^*$ must contain a large power of two subgroup

The condition that $\mathbb{F}_p^*$ contains a large power of two subgroup is necessary to perform FFTs. In the second example, it's the only constraint on p.

Problem to solve

Circuits reason on variables that live in $\mathbb{F}_p$

It means that writing a circuit where the variables naturally live in $\mathbb{F}_p$ is efficient constraint-wise. But writing a circuit where the variables live elsewhere (in another finite field $\mathbb{F}_r$ for instance) will be quite inefficient constraint wise, because we need to emulate the field $\mathbb{F}_r$ in $\mathbb{F}_p$.

Example: Writing a $secp256k1$ verification circuit in on $BN254$.

Can we instantiate a SNARK directly on $secp256k1$'s field of definition? More generally, can we get rid of the constraint of p to instantiate a SNARK over any field?

Problem to solve

Without $\mathbb{F}_p^*$ containing a large power of two subgroup, we cannot do FFTs...

The ECFFT seems to be the answer. One could instantiate a plonk-like proof system, using FRI (which can be instantiated similarly to the ECFFT), and the FFT could be replaced by the ECFFT.

The goal of this talk is to list the problems of this approach.

FFT, geometric point of view

- The FFT is merely a change of basis (from canonical basis to the dual basis of a certain basis $\mathcal{B}$ of $\mathbb{F}_p[X]^*_{<n}$, namely the Lagrange basis)
- If $P \in \mathbb{F}_p[X]_{<n}$, such an operation can be done in $O(n\log(n))$, provided that $n = 2^r$ and $2^r | p-1$ (instead of the $O(n^2)$ in a naive implementation)

FFT, geometric point of view

Start with a separable rational map of degree k

$$f : \mathbb{P}_1(\mathbb{F}_p) \longmapsto \mathbb{P}_1(\mathbb{F}_p)$$

and 2 sets $S_1, S_2 \subsetneq \mathbb{P}_1(\mathbb{F}_p)$ with

- $f^{-1}(S_2) = S_1$
- f does not ramify on any point on S_1

Note: since $S_1, S_2 \neq \mathbb{P}_1(\mathbb{F}_p)$, we can assume $\infty \notin S_1, S_2$, and we can interpret f as a function defined on S_1.

FFT, geometric point of view

Radix 2 FFT: we work on $\mathbb{F}_p$, where $2^r | p - 1$.

- $\mathbb{F}_p^*$ contains a large subgroup of order $n = 2^r$
- $f : X \longmapsto X^2$
- $S_1 = V(< X^n - 1 >), S_2 = V(< X^{n/2} - 1 >)$.

We see that f does not ramify on S_1 and $f^{-1}(S_2) = S_1$.

© Thomas Piellard, Consensys

FFT, geometric point of view

Write $\mathbb{F}_p[S_i]$ the ring of functions defined over S_i.

Concretely, if for instance $S_1 = \{x_1, \ldots, x_n\}$, we have

$$\mathbb{F}_p[S_1] = \mathbb{F}_p[X]/(X - x_1)\ldots(X - x_n) = \mathbb{F}_p[X]/P_1$$

Since f does not ramify on S_1 and $f^{-1}(S_2) = S_1$, we have

$$|S_1| = k|S_2|$$

Similarly $\mathbb{F}_p[S_2] = \mathbb{F}_p[Y]/(Y - y_1)\ldots(Y - y_{n/k}) = \mathbb{F}_p[Y]/P_2$

Example: In the radix-2 FFT:

- $\mathbb{F}_p[S_1] = \mathbb{F}_p[X]/ < X^n - 1 >$
- $\mathbb{F}_p[S_2] = \mathbb{F}_p[Y]/ < Y^{n/2} - 1 >$

FFT, geometric point of view

If $f = U(X)/V(X)$, the hypothesis ensures that

- $\deg(U) = k$
- $\deg(V) \leq k$
- V does not vanish on S_1

f induces an injection

$$\begin{aligned} \tilde{f} : \mathbb{F}_p[S_2] &\hookrightarrow \mathbb{F}_p[S_1] \\ Q &\longmapsto \tilde{Q} = V^{|S_2|-1}(Q \circ f) \end{aligned}$$

Example: In the radix-2 FFT, $f(X) = X^2$. If $Q \in \mathbb{F}_p[S_2]$, $\tilde{f}(Q) = Q(X^2)$

FFT, geometric point of view

Indeed, if $Q \in \mathbb{F}_p[S_2]$, composing with f defines a function

$$Q \circ f = Q(U/V)$$

defined on S_1 since V does not vanish on S_1.

Moreover, if $Q \circ f = 0$ on S_1, then $Q(y) = 0$ for all $y \in S_2$ so it means that $Q = 0$.

Since V is invertible in $\mathbb{F}_p[S_1]$, the multiplication by $V^{|S_2|-1}$ gives an isomorphism on $\mathbb{F}_p[S_1]$.

Note: the multiplication by $V^{|S_2|-1}$ avoids to compute V^{-1}. Mathematically it's not necessary to multiply by $V^{|S_2|-1}$.

FFT, geometric point of view

$\mathbb{F}_p[S_1]$ is a finite module of rank k over $\mathbb{F}_p[S_2]$ with basis $\{1, X, \dots, X^{k-1}\}$.

Indeed

$$\mathbb{F}_p[S_1] \cong \mathbb{F}_p[S_2][X]$$

where X satisfies $U(X) - V(X)Y = 0$. There are 2 cases:

- If $\deg(V) < k$, X satisfies the integral equation $U(X) - V(X)Y = 0$ (the leading coefficient in X is in $\mathbb{F}_p$)
- If $\deg(V) = k$, the leading coefficient in X in $U - VY = 0$ contains Y. If it vanishes on S_2, for some $y_i \in S_2$, the term in X^k in $U - Vy_i$ is zero, but this equation has k solutions and is is non zero. So the leading term does not vanish on S_2. So X is also the solution of an integral equation of degree k.

FFT, geometric point of view

$\mathbb{F}_p[S_1]$ is a finite module of rank k over $\mathbb{F}_p[S_2]$ with basis $\{1, X, \ldots, X^{k-1}\}$.

Using the injection

$$\begin{aligned} \tilde{f} : \mathbb{F}_p[S_2] &\hookrightarrow \mathbb{F}_p[S_1] \\ Q &\longmapsto \tilde{Q} = V^{|S_2|-1}(Q \circ f) \end{aligned}$$

it means from $Q_1, \ldots, Q_k \in \mathbb{F}_p[S_2]$ we can map back to an element $Q \in F_p[S_1]$ where

$$Q = \sum_{i \in [1,k]} \tilde{f}(Q_i) X^{i-1} = V^{|S_2|-1} \sum_{i \in [1,k]} (Q_i(U/V) X^{i-1}$$

and vice versa.

FFT, geometric point of view

$\mathbb{F}_p[S_1]$ is a finite module of rank k over $\mathbb{F}_p[S_2]$ with basis $\{1, X, \ldots, X^{k-1}\}$.

Example: For the radix-2 FFT, $f(X) = U(X)/V(X) = X^2$,

$$\begin{aligned} \tilde{f} : \mathbb{F}_p[S_2] &\longmapsto \mathbb{F}_p[S_1] \\ Q &\longmapsto Q \circ f = Q(X^2) \end{aligned}$$

If $Q_1, Q_2 \in \mathbb{F}_p[S_2]$, $Q := \tilde{f}(Q_1) + \tilde{f}(Q_2)X = Q_1(X^2) + XQ_2(X^2) \in \mathbb{F}_p[S_1]$

© Thomas Piellard, Consensys

FFT, geometric point of view

What we have so far:

- A separable, degree k map $f : \mathbb{P}_1(\mathbb{F}_p) \longmapsto \mathbb{P}_1(\mathbb{F}_p)$
- 2 rings, $R_1 := \mathbb{F}_p[S_1]$ and $R_2 := \mathbb{F}_p[S_2]$
- R_1 is module finite of rank k over R_2 so $R_1 \cong R_2^k$

Where we go:

- How to convert a polynomial Q to a certain Lagrange basis?
- How to convert back a polynomial Q in canonical basis?

If the chosen evaluation set is S_1, with $|S_1|$ of size n (so the polynomial must be of degree $|S_1| - 1$) we can use the rings R_1 and R_2.

FFT, geometric point of view

The FFT relies on this decomposition of modules, starting from a polynomial in $\mathbb{F}_p[S_1]$. It relies on the following assumptions:

- the module decomposition $DEC : R_1 \longmapsto R_2^k$, and its inverse DEC^{-1}, are efficient to compute when the polynomial and its components in R_2 are in **canonical basis**
- the module decomposition $DEC : R_1 \longmapsto R_2^k$, and its inverse DEC^{-1}, are efficient to compute when the polynomial and its components in R_2 are in **Lagrange basis**

From now on we put a "tilde" on DEC or DEC^{-1} when the operation is done in Lagrange basis.

Ex: $D\tilde{E}C(Q) = Q_1, \ldots, Q_k$ means that $Q, Q_1, \ldots, Q_k$ are in Lagrange basis.

FFT, geometric point of view

General FFT algorithm:

```
Require $Q \in \mathbb{F}_p[S_1]$ in canonical form
Ensure $\tilde{Q}$, i.e. $Q$ in Lagrange basis $(Q(x))_{x \in S_1}$
$(Q_1, \ldots, Q_k) \leftarrow$ DEC(Q)
$(\tilde{Q}_1, \ldots, \tilde{Q}_k) \leftarrow (FFT(Q_i))_{i<k}$
$\tilde{Q} \leftarrow D\tilde{E}C^{-1}(\tilde{Q}_1, \ldots, \tilde{Q}_k)$
```

FFT, geometric point of view

General FFT inverse algorithm:

```
**Require** $\tilde{Q} \in \mathbb{F}_p[S_1]$ in Lagrange basis
**Ensure** $Q$, i.e. $\tilde{Q}$ in canonical basis
$(\tilde{Q}_1, \ldots, \tilde{Q}_k) \leftarrow \tilde{DEC}(\tilde{Q})$
$(Q_1, \ldots, Q_k) \leftarrow (FFT^{-1}(\tilde{Q}_i))_{i<k}$
$Q \leftarrow DEC^{-1}(Q_1, \ldots, Q_k)$
```

FFT, geometric point of view

The operation $\tilde{DEC} : R_1 \longmapsto R_2^k$ is efficient to compute.

- $Q \in \mathbb{F}_p[S_1]$ is in Lagrange basis $(Q(x_1), \ldots, Q(x_n))$
- let $y \in S_2$, with $f^{-1}(y) = \{x_1, \ldots, x_k\}$

$\tilde{DEC} : R_1 \longmapsto R_2^k$ is computed by solving the systems $\mathcal{S}_y$:

$$V(x_1)^{|S_2|-1}(\sum_{j<k} Q_j(y)x_1^j) = Q(x_1)$$

$$\ldots$$

$$V(x_k)^{|S_2|-1}(\sum_{j<k} Q_j(y)x_k^j) = Q(x_k)$$

(In $Q = V^{|S_2|-1} \sum_{i\in[1,k]}(Q_i(U/V)X^{i-1}$, replace X by x_i)

FFT, geometric point of view

In fact, if we stack those systems matricially, the resulting matrix is block diagonal, with n/k blocks of size k corresponding to the sub systems $\mathcal{S}_y$:

$$\underbrace{\begin{pmatrix} V(x_1)^{|S_2|-1} & \dots & V(x_1)^{|S_2|-1}x_1^{k-1} \\ & \dots & \\ V(x_k)^{|S_2|-1} & \dots & V(x_k)^{|S_2|-1}x_k^{k-1} \end{pmatrix}}_{\mathcal{S}_y \in GL_k(F_p)} \underbrace{\begin{pmatrix} Q_1(y) \\ \dots \\ Q_k(y) \end{pmatrix}}_{l_y} = \underbrace{\begin{pmatrix} Q(x_1) \\ \dots \\ Q(x_k) \end{pmatrix}}_{r_y}$$

FFT, geometric point of view

Overall the system looks like this:

$$\begin{pmatrix} \mathcal{S}_1 & \dots & 0 \\ \dots & \mathcal{S}_2 & \dots \\ 0 & \dots & \mathcal{S}_{n/k} \end{pmatrix} \begin{pmatrix} l_1 \\ \dots \\ l_{n/k} \end{pmatrix} = \begin{pmatrix} r_1 \\ \dots \\ r_{n/k} \end{pmatrix}$$

The matrices $(\mathcal{S}_i)_{i=1\dots n/k}$ and their inverse can be precomputed.

So similarly to $\tilde{DEC}$, $\tilde{DEC}^{-1}$ is computationally efficient.

Main issue

When polynomials are in canonical basis, the operations DEC and DEC^{-1} are **not** efficient to compute, except when f is extremely simple.

It is the case for the radix-2 FFT, where $f : X \longmapsto X^2$, the operations DEC and DEC^{-1} are extremely simple:

- $DEC(Q) = Q_1, Q_2$ where Q_1 and Q_2 rep. contain the even and odd powers
- $DEC^{-1}(Q_1, Q_2) = Q_1(X^2) + XQ_2(X^2)$

But in general, DEC and DEC^{-1} are not easy to compute. It requires $O(n^2)$ operations, which is the cost of doing the FFT naively.

Note: For ECFRI, only $\widetilde{DEC}$ and $\widetilde{DEC}^{-1}$ are necessary.

Curves

What do curves have to do with all this?

$\rightarrow$ Finding separable degree k functions on $\mathcal{P}_1(\mathbb{F}_p)$ is easy. What is less easy is to define the domains S_1 and S_2.

Elliptic curves

In the radix 2 FFT this problem is solved by the group structure of S_1. It is a cyclic group C_r of size 2^r, and $f : X \longmapsto X^2$ maps it onto C_{r-1}.

In the "geometric" FFT, S_1 and S_2 are no longer groups. The get some structure, the idea is to pick S_1 and S_2 as the x coordinates of two isogenous elliptic curves **in Weierstrass form**, and f as the x-coordinate of the isogeny.

Elliptic curves

Let $E_1(\mathbb{F}_p)$, $E_2(\mathbb{F}_p)$ two $\mathbb{F}_p$-isogenous elliptic curves in Weierstrass form, let Φ be the isogeny, suppose that ϕ is separable and of degree k. There is a rational function $R = A/B$ with $k = \deg(A) = \deg(B) + 1$ and $\alpha \in \mathbb{F}_p$ such that

$$\Phi(x, y) = (R(x), \alpha y R'(x))$$

It relies on 2 facts:

- The pullback of a holomorphic 1-form is holomorphic
- The $\bar{\mathbb{F}_p}$-vector space of holomorphic 1-form is of dimension 1, of basis dx/y (because we are in Weierstrass form)

Example of a technical issue

Recall: $\Phi(x,y) = (R(x), \alpha y R'(x))$, and $\ker(\Phi) \subset E_1(\mathbb{F}_p)$

R ramifies on the x-coordinates of points P such that $[2]P \in \ker(\Phi)$.

Indeed $[2]P \in \ker(\Phi)$ means that P and $-P$ belong to the same coset. But R ramifies exactly on the x-coordinates of opposite points, so R ramifies on the x-coordinate of P.

Elliptic curves

To perform the full FFT, we actually need:

- a **sequence** of sets $S_1, S_2, \ldots, S_{\log(n)}$
- a **sequence** of functions $f_i : S_i \longmapsto S_{i+1}$

Example: in the radix-2 FFT, for $i = 1 \ldots \log(n)$:

- $S_i = V(< X^{n/2^i} - 1 >)$
- $f_i : X \longmapsto X^2$

The corresponding elliptic curve version of this is the doubling map. It's a special case where the isogeny is an endomorphism. It is a map of degree 4, so $R = A/B$ with $\deg(A) = 4$.

The doubling map

If the doubling map on the group $\mathbb{Z}/2^r\mathbb{Z} \times \mathbb{Z}/2^r\mathbb{Z}$ can be used, then the radix-2 FFT can be used.

Indeed, to use the doubling map we must have $E(\bar{\mathbb{F}_p})[2^r] \subset E(\mathbb{F}_p)$ otherwise the fibers of the doubling map are not defined completely on $E(\mathbb{F}_p)$.

A necessary condition for $E(\bar{\mathbb{F}_p})[n] \subset E(\mathbb{F}_p)$ is that

$$n|p-1$$

Here it means that $2^r|p-1$ so $\mathbb{F}_p$ contains a 2^r cyclic subgroup, so the radix 2-FFT can used.

Low degree isogenies

We want isogenies of small degree, ideally 2 (so $\deg(R) = 2$).

In order to do the ECFFT for polynomials of degree 2^r, it is enough to pick a curve whose number of points is divisible by 2^r.

Indeed if $2^r | \#E_1(\mathbb{F}_p)$, there is a subgroup $G_1 \subset E_1(\mathbb{F}_p)$ of size 2^r, and we can pick $S_1 = x(P + G_1)$. We can also find easily a curve $E_2(\mathbb{F}_p)$ and $\mathbb{F}_p$-isogeny of degree 2 $f_1 : E_1 \longmapsto E_2$. Then we take $G_2 = f_1(G_1)$ and $S_2 = x(f_1(P) + G_2)$. $G_2 \subset E_2(\mathbb{F}_p)$ of size 2^{r-1}, etc.

Low degree isogenies

For a fixed prime p and a large r, it's very difficult to generate efficiently a curve whose group points over $\mathbb{F}_p$ is divisible by 2^r: we cannot control the discriminant for the CM.

By random sampling, there is $1/2^r$ chances to find a curve whose group of points is divisible by 2^r.

There are p isomorphism classes of elliptic curves over $\mathbb{F}_p$, and the number of curves whose group of points over $\mathbb{F}_p$ is divisible by 2^r is

- $\frac{1}{2^r-1}p + O(2^r\sqrt{p})$ if $p \neq 1 \mod 2^r$
- $\frac{2^r}{2^{2r}-1}p + O(2^r\sqrt{p})$ if $p = 1 \mod 2^r$

Conclusion

List of open questions:

- How to compute DEC, DEC^{-1} efficiently?
- If DEC, DEC^{-1} cannot be computed efficiently, could we instantiate snark circuits small enough, using ECFRI as commitment scheme, and for polynomial operations use standard algorithms (not FFT)?
- Can we instantiate the functions $f_i : S_i \subset \mathcal{P}_1(\mathbb{F}_p) \longmapsto \subset \mathcal{P}_1(\mathbb{F}_p)$ otherwise than taking the x-coordinates of an isogeny between elliptic curves?

Questions

Questions ?

Check out our repo...

https://github.com/consensys/gnark
https://github.com/consensys/gnark-crypto

© Thomas Piellard, Consensys