EECS 498/CSE 598: Zero-Knowledge Proofs
Winter 2026
Lecture 4

Paul Grubbs

paulgrub@umich.edu

Beyster 4709

Agenda for this lecture

- Announcements
- Elliptic curve groups
- Computing with elliptic curves
- Polynomials
- The Schwartz-Zippel lemma
- A randomized protocol for computing string equality
- Efficient algorithms for polynomials

(Skipping pairings background until later)

Agenda for this lecture

- Announcements
- Elliptic curve groups
- Computing with elliptic curves
- Polynomials
- The Schwartz-Zippel lemma
- A randomized protocol for computing string equality
- Efficient algorithms for polynomials

Announcements

- Project 1 is online
- Autograder is working

football?!

Agenda for this lecture

- Announcements
- Elliptic curve groups
- Computing with elliptic curves
- Polynomials
- The Schwartz-Zippel lemma
- A randomized protocol for computing string equality
- Efficient algorithms for polynomials

Elliptic curve basics

$$y^2 = x^3 + ax + b \qquad a, b \in \mathbb{F}_p$$

$$E_0 = \{(x,y) \mid y^2 = x^3 + ax + b\}$$

$$E := E_0 \cup \{\mathcal{O}\}$$

$$|E/\mathbb{F}_p|$$

$\boxed{\text{Theorem:}}$

$$\big|\,|E/\mathbb{F}_p| - p\,\big| \approx \sqrt{p}$$

Fact: Can compute $|E/\mathbb{F}_p|$ efficiently (Schoof)

Scalar mult:

Add a point to itself

$aP \quad a \in \mathbb{Z} \quad P \in E/\mathbb{F}_p$

$\underbrace{P + \dots + P}_{a}$

$\overset{a}{\rightarrow} |E/\mathbb{F}_p| P = \mathcal{O} \qquad a = bq + r$

$aP = bq + rP$

$\cancel{bqP} + rP$

Group size:

Compute generators

EC groups we care about are either cyclic or have generated w/ large order

Multi-Scalar multiplication

n points $P_1 \ldots P_n$
scalars $a_1 \ldots a_n$

Compute $\sum a_i P_i$

Bottleneck

Bucket method

(pippenger)

MSM is additively homomorphic:

$$MSM(\vec{a}) + MSM(\vec{b}) = MSM(\vec{a} + \vec{b})$$

Computational issues

Secure ECC impls

- side channels
  Montgomery ladder
- Invalid curve attack
  Adversary gives you
  $(x, y) \notin E/\mathbb{F}_p$

P256:
both p and $E/\mathbb{F}_p \approx 256$ bits

36.5 μSec
PQ crypto: faster, but bigger

Agenda for this lecture

- Announcements
- Elliptic curve groups
- Computing with elliptic curves
- Polynomials
- The Schwartz-Zippel lemma
- A randomized protocol for computing string equality
- Efficient algorithms for polynomials

Computational questions

See prev

Agenda for this lecture

- Announcements
- Elliptic curve groups
- Computing with elliptic curves
- Polynomials
- The Schwartz-Zippel lemma
- A randomized protocol for computing string equality
- Efficient algorithms for polynomials

What is a polynomial?

- Sum of vars of different powers
- list of coefficients
- many variables ?
- encoding of data

Terminology

Degree	(univariate) maximum power of x with nonzero coefficient
“total” degree	maximum of sum of exponents in any single term
“maximal” degree in a variable	(univariate) degree in that variable, viewing other vars as fixed
root	for p(x), root r is point so that p(r) = 0
*-variate (eg univariate, multivariate)	how many variables. Uni=1, multi=more than 1.
multilinear	Multivariate polynomial with max degree 1 in each variable.
monomial/term	either a single variable with exponent, or product of vars with exponents
monic	leading coefficient is 1
evaluation	compute result of “plugging in” value for variables and reducing using add/mul
partial evaluation	Substituting values for only some variables. gives polynomial in fewer variables.

$$f(x) \coloneqq x_1x_2x_3 + 4x_1x_2 + 3x_2x_3 + 6x_1 + 1$$

over $\mathbb{F}_7$

~~Low-degree and multilinear extensions~~

Divisibility

Let $a, b \in \mathbb{F}[X]$ $\deg(a) > \deg(b)$

$\exists q, r \in \mathbb{F}[X]$ s.t. $\deg(r) < \deg(b)$

$$a(X) = q(X)\,b(X) + r(X)$$

If $r = 0$ $b \mid a$

Polynomial long division computes q, r

Polynomial GCD a, b $\exists d$

$d \mid a$ and $d \mid b$

EEA gives x, y s.t.

$$ax + by = d = \gcd(a, b)$$

Modular arith. of polynomials

$a(X) \bmod m(X)$

remainder of a div m

Let $b(X) = (X-a_1) \cdots (X-a_k)$

Then a/b :

$$a(X) = q(X)b(X) + r(X)$$

$$a(a_i) = r(a_i) \quad \forall i \in [k]$$

$$a(a_1) = \cancel{q(a_1)\, b(a_1)} + r(a_1)$$

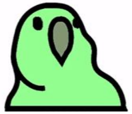
$$(((\mathbb{F}[X_1])[X_2])[X_3])[X_4]$$

Tower

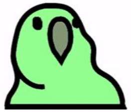
Agenda for this lecture

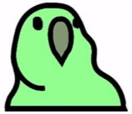
- Announcements
- Elliptic curve groups
- Computing with elliptic curves
- Polynomials
- The Schwartz-Zippel lemma
- A randomized protocol for computing string equality
- Efficient algorithms for polynomials

Schwartz-Zippel

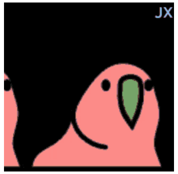



Schwartz-Zippel



It’s Schwartz-Zippel time folks!



Schwartz-Zippel

Lemma 3.3 (Schwartz-Zippel Lemma). *Let $\mathbb{F}$ be any field, and let $g : \mathbb{F}^m \to \mathbb{F}$ be a nonzero m-variate polynomial of total degree at most d. Then on any finite set $S \subseteq \mathbb{F}$,*

$$\Pr_{x \leftarrow S^m}[g(x) = 0] \leq d/|S|.$$

Base case $m=1$: by FTA

Induction: Take $g(X_1, \dots, X_m)$

Assume for $m-1$

write as poly $\sum_{i=0}^{d} X_1^i \, g_i(X_2, \dots X_m)$

Sample $r_2, \dots r_m$ apply g_i

Ind. Hyp.

(Statement from Thaler)

Agenda for this lecture

- Announcements
- Elliptic curve groups
- Computing with elliptic curves
- Polynomials
- The Schwartz-Zippel lemma
- A randomized protocol for computing string equality
- Efficient algorithms for polynomials

An interactive protocol for string equality

Alice $s_A \in \{0,1\}^n$

Bob $s_{\mathrm{B}} \in \{0,1\}^n$

An interactive protocol for string equality

Alice $s_A \in \{0,1\}^n$

Bob $s_B \in \{0,1\}^n$

Sample $r \leftarrow_\$ \mathbb{F}$

$r \in \mathbb{F}$

An interactive protocol for string equality

Alice $s_A \in \{0,1\}^n$

Bob $s_{\mathrm{B}} \in \{0,1\}^n$

Sample $r \leftarrow_{\$} \mathbb{F}$

$r \in \mathbb{F}$ →

- $s_B(x) \coloneqq s_{B,0} + s_{B,1}x + \cdots + s_{B,n-1}x^{n-1}$
- Compute $t \leftarrow s_B(r)$

← $\mathrm{t} \in \mathbb{F}$

An interactive protocol for string equality

Alice $s_A \in \{0,1\}^n$

Bob $s_{\mathrm{B}} \in \{0,1\}^n$

Sample $r \leftarrow_{\$} \mathbb{F}$

$r \in \mathbb{F}$ →

- $s_B(x) \coloneqq s_{B,0} + s_{B,1}x + \cdots + s_{B,n-1}x^{n-1}$
- Compute $t \leftarrow s_B(r)$

← $\mathrm{t} \in \mathbb{F}$

- $s_{\mathrm{A}}(x) \coloneqq s_{\mathrm{A},0} + s_{\mathrm{A},1}x + \cdots + s_{\mathrm{A},n-1}x^{n-1}$
- Compute $u \leftarrow s_{\mathrm{A}}(r)$
- If $t = u$ return “equal”, else “not equal”

An interactive protocol for string equality

Alice $s_A \in \{0,1\}^n$

Bob $s_B \in \{0,1\}^n$

Sample $r \leftarrow_\$ \mathbb{F}$

$r \in \mathbb{F}$ →

- $s_B(x) \coloneqq s_{B,0} + s_{B,1}x + \cdots + s_{B,n-1}x^{n-1}$
- Compute $t \leftarrow s_B(r)$

← $\mathrm{t} \in \mathbb{F}$

- $s_A(x) \coloneqq s_{A,0} + s_{A,1}x + \cdots + s_{A,n-1}x^{n-1}$
- Compute $u \leftarrow s_A(r)$
- If $t = u$ return "equal", else "not equal"

If we choose the field to be $poly(n)$, S-Z lemma says we get inv-poly error with only $O(\log n)$ communication
=> exponential improvement over deterministic case!

Agenda for this lecture

- Announcements
- Elliptic curve groups
- Computing with elliptic curves
- Polynomials
- The Schwartz-Zippel lemma
- A randomized protocol for computing string equality
- Efficient algorithms for polynomials

Computational questions