p-adic Numbers in Cryptography

Kaio Deeter

June 8, 2026

1 Cryptography

Cryptography is the study of techniques for secure communication. The goal is to allow two parties (usually called Alice and Bob) to exchange messages over an insecure channel in such a way that an eavesdropper (Eve) learns nothing about the contents. The original message Alice wishes to send is called the *plaintext*, and the scrambled version is called the *ciphertext*. The process of transforming plaintext into ciphertext is called *encryption*, and the reverse process is called *decryption*.

Definition 1.1 (Cipher)**.** A *cipher* is a pair of algorithms (E, D), together with a set of *keys* $\mathcal{K}$, such that for every key $k \in \mathcal{K}$ and every plaintext m,

$$D_k(E_k(m)) = m.$$

The algorithm E is called the *encryption function* and D the *decryption function*.

The security of a cipher rests on the principle that the algorithms E and D are public while the key k is secret. A cipher is secure if recovering m from $E_k(m)$ without knowledge of k is computationally infeasible. Historically, ciphers were *symmetric* meaning Alice and Bob shared a single secret key k used for both encryption and decryption. The difficulty is how Alice and Bob could agree on a key in the first place without a secret channel. This problem was resolved in the 1970s with the introduction of *public-key cryptography*, in which each participant holds a pair of keys: a *public key* and a *private key*. Anyone can encrypt a message to Bob using his public key, but only Bob can decrypt it with his private key. This type of cryptography uses a *one-way function*.

Definition 1.2 (One-Way Function)**.** A function $f : X \to Y$ is *one-way* if $f(x)$ can be computed quickly for any $x \in X$, but given $y = f(x)$ for a randomly chosen x, no efficient algorithm can find a preimage $x' \in X$ with $f(x') = y$.

Efficient usually means running in time polynomial to the size of the input, and hard means that no known algorithm does better than brute-force search. While the existence of one-way functions has never been proven (P=NP), cryptography rests on the fact that we believe certain functions are one-way because no one has found a way to invert them quickly. An example is the *discrete logarithm*. Let G be a finite cyclic group with generator g and order n. Given $a \in \{0, 1, \ldots, n-1\}$, the element $g^a \in G$ can be computed in roughly $\log_2 n$ multiplications using repeated squaring. The reverse problem (given $h = g^a \in G$, find a) is the discrete logarithm problem, and for well-chosen groups it is believed to require exponential time. This single hardness assumption is enough to let Alice and Bob agree on a shared secret over a public channel. They publicly fix a group G and a generator g. Alice chooses a secret integer a and sends g^a; Bob chooses a secret integer b and sends g^b. Each then computes

$$(g^b)^a = g^{ab} = (g^a)^b,$$

which becomes their shared secret. Eve, observing only g^a and g^b, must solve a discrete-logarithm-type problem to recover g^{ab}. A closely related construction is the RSA cryptosystem which is based on on the difficulty of factoring large integers. The discrete logarithm and integer factorization are the two classical of public-key cryptography, and most deployed systems today reduce to one of them, while often also using *elliptic curve cryptography*, where the group G is taken to be the points of an elliptic curve over a finite field.

2 Random-Looking Sequences from p-adic Maps

The best-developed part of p-adic cryptography produces random-looking bits by iterating a single function. A 2-adic integer is essentially an infinite string of bits, so we pick a function $f : \mathbb{Z}_2 \to \mathbb{Z}_2$, start at some x_0, and apply f over and over,

$$x_0, \quad x_1 = f(x_0), \quad x_2 = f(x_1), \quad \dots$$

Reading off the lowest bit of each x_i gives a stream of bits, which a stream cipher adds to the plaintext. We want that stream to look random.

We only allow functions f for which the first k bits of $f(x)$ depend only on the first k bits of x. This is called *compatible*. Equivalently, if $x \equiv y \pmod{2^k}$ then $f(x) \equiv f(y) \pmod{2^k}$. The reason this matters is that a compatible f then makes sense as a function on the finite set $\{0, 1, \dots, 2^k - 1\}$ for each k, so we can split it up by k's.

Fix a k, so f acts on the 2^k values $\{0, 1, \dots, 2^k - 1\}$. We then need f to run through *all* 2^k of these values in one big loop before repeating:

$$x_0 \to x_1 \to \cdots \to x_{2^k-1} \to x_0,$$

hitting each value exactly once. When this happens for every window size k, f is called *transitive*. Transitivity is important because the stream then has the longest possible period, and every short bit-pattern shows up equally often, making it more random.

Anashin's test

To state the test, write f in its *Mahler expansion*, which is the p-adic analogue of a Taylor series:

$$f(x) = \sum_{m=0}^{\infty} a_m \binom{x}{m}, \qquad \binom{x}{m} = \frac{x(x-1)\cdots(x-m+1)}{m!},$$

with coefficients $a_m \in \mathbb{Z}_2$. Kurt Mahler proved this representation in 1958.

Theorem 2.1 (Anashin's criterion for $p = 2$)**.** The function $f(x) = \sum_m a_m \binom{x}{m}$ is transitive on $\mathbb{Z}_2$ if and only if

$$a_0 \text{ is odd}, \qquad a_1 \equiv 1 \pmod 4, \qquad a_m \equiv 0 \pmod{2^{\lfloor \log_2(m+1) \rfloor + 1}} \text{ for all } m \geq 2.$$

This means that in order to decide if our f is good, we just need to check the coefficients of the Mahler expansion.

Example 2.1. Take $f(x) = 2x^2 - x + 1$. The Mahler expansion is

$$f(x) = 1 + x + (2x^2 - 2x) = 1 + \binom{x}{1} + 4\binom{x}{2},$$

so $a_0 = 1$, $a_1 = 1$, $a_2 = 4$, and every other $a_m = 0$ so f is transitive.

We can watch it happen with 3 bits (so modulo 8). Starting from $x_0 = 0$:

$$0 \to 1 \to 2 \to 7 \to 4 \to 5 \to 6 \to 3 \to 0,$$

one loop through all eight values, exactly as the theorem promises. The lowest bits along this orbit, $0, 1, 0, 1, 0, 1, 0, 1, \ldots$, are the beginning of the keystream.

If f is transitive then, over one full period, the orbit lands on each value exactly once. In bits this means every block of k low-order bits appears equally often and the period is as long as it can be. This is what the standard randomness tests look for, which makes such a function f good for cryptography.

Unfortunately, looking random is necessary but not sufficient for a secure cipher. A stream can be perfectly uniform and still be predictable. In practice a transitive f may be used inside a larger design, with extra scrambling layered on top of it.

3 Breaking Codes with p-adic Numbers

The previous section used p-adic analysis to build a primitive. Another important application of the p-adics to cryptography is a family of elliptic curves whose security goes away when lifting the problem from $\mathbb{F}_p$ to $\mathbb{Q}_p$.

Elliptic curves and their discrete logarithm

An *elliptic curve* over $\mathbb{F}_p$ (with $p > 3$) is the set of solutions to

$$E : y^2 = x^3 + ax + b, \qquad a, b \in \mathbb{F}_p, \quad 4a^3 + 27b^2 \neq 0,$$

together with a single extra point O at infinity. The $(4a^3 + 27b^2)$ term is part of the discriminant which ensures that the curve is smooth. The points $E(\mathbb{F}_p)$ form a finite abelian group where three points sum to O exactly when they are collinear, making O the identity. The size of the group can be found with Hasse's Theorem on elliptic curves,

$$\#E(\mathbb{F}_p) = p + 1 - a_p, \qquad |a_p| \leq 2\sqrt{p},$$

Cryptographically, $E(\mathbb{F}_p)$ plays the role of the group G from Section 1. Given a base point P and a multiple $Q = nP$, the *elliptic curve discrete logarithm problem* (ECDLP) is to recover n. For a generic curve the best known algorithms (Pollard's ρ, baby-step/giant-step) take time about $\sqrt{p}$ and this underlies essentially all elliptic-curve cryptography.

Definition 3.1 (Anomalous curve)**.** An elliptic curve E over $\mathbb{F}_p$ is *anomalous* if $\#E(\mathbb{F}_p) = p$.

For such a curve $E(\mathbb{F}_p) \cong \mathbb{Z}/p\mathbb{Z}$, a cyclic group of order exactly p.

Lifting the curve to $\mathbb{Q}_p$

The curve E is given by an equation $y^2 = x^3 + ax + b$ with $a, b \in \mathbb{F}_p$. Choose any $\tilde{a}, \tilde{b} \in \mathbb{Z}_p$ reducing to a, b modulo p, which gives a curve $\tilde{E}$ over $\mathbb{Q}_p$ whose reduction mod p is E. Reducing mod p is a group homomorphism $\pi : \tilde{E}(\mathbb{Q}_p) \to E(\mathbb{F}_p)$, and any point of $E(\mathbb{F}_p)$ can be lifted to a point of $\tilde{E}(\mathbb{Q}_p)$ by solving the equation p-adically (Hensel's lemma).

The points of $\tilde{E}(\mathbb{Q}_p)$ that reduce to the identity O form a subgroup, which is E_1 or the *kernel of reduction*. These are the points sitting infinitesimally close to O.

The formal logarithm

Recall that the ordinary logarithm turns multiplication into addition: $\log(uv) = \log u + \log v$. An elliptic curve has an exact analogue near its identity. Using the local coordinate $t = -x/y$, which is small (in fact $t \in p\mathbb{Z}_p$) exactly for the points of E_1, there is a power series

$$\log_E(t) = t + (\text{higher-order terms})$$

that converts the curve's addition law into ordinary addition of p-adic numbers. It is a group isomorphism

$$\log_E : E_1 \xrightarrow{\sim} (p\mathbb{Z}_p, +).$$

The complicated chord-and-tangent group law then just becomes addition in $p\mathbb{Z}_p$ and the discrete log is just division.

Theorem 3.1 (Smart; Satoh–Araki; Semaev, 1997–1999)**.** For an anomalous elliptic curve $E/\mathbb{F}_p$, the discrete logarithm problem on $E(\mathbb{F}_p)$ can be solved in time polynomial in $\log p$ by lifting to $\mathbb{Q}_p$ and applying the formal logarithm.

We want the n with $Q = nP$. Build a function ψ from curve points to $\mathbb{F}_p$ like this: given $R \in E(\mathbb{F}_p)$, lift it to $\tilde{R}$ and form $p\tilde{R}$. Because the curve has exactly p points, $pR = O$ in $E(\mathbb{F}_p)$, so $p\tilde{R}$ reduces to O and it lands in E_1, where $\log_E$ is available. Set

$$\psi(R) = \frac{\log_E(p\tilde{R})}{p} \bmod p \ \in \ \mathbb{F}_p.$$

(Dividing by p is legitimate because $\log_E(p\tilde{R}) \in p\mathbb{Z}_p$) Since $\log_E$ turns curve-addition into number-addition, ψ is additive:

$$\psi(R + S) = \psi(R) + \psi(S).$$

In other words ψ flattens the curve group into $(\mathbb{F}_p, +)$. Applying it to $Q = nP$ gives $\psi(Q) = n\,\psi(P)$, so

$$n \equiv \frac{\psi(Q)}{\psi(P)} \pmod{p},$$

a single division in $\mathbb{F}_p$ (valid whenever $\psi(P) \neq 0$). The whole computation is a few curve operations, one power-series evaluation, and one division which makes analogous curves much easier to break.

There are three points that are worth noting in this process. First, anomalous curves are usually pretty scarce as a random curve over $\mathbb{F}_p$ is anomalous with probability on the order of $1/\sqrt{p}$ so it's hard to encounter one by accident. Second, the construction has a single failure since for one special choice of lift the denominator $\psi(P)$ comes out 0 and you have to relift which is why the algorithm is described as choosing the lift at random. Third, there is a pretty simple way to defend against it. One computes $\#E(\mathbb{F}_p)$ when generating parameters and discards the curve if it equals p. Every modern elliptic-curve standard does this. Even though it can't really be applied, it's applicatoin can be seen by restricting the selection of curves.

References

[1] V. S. Anashin, *Uniformly distributed sequences of p-adic integers*, Mathematical Notes **55** (1994), no. 1–2, 109–133.

[2] K. Mahler, *An interpolation series for continuous functions of a p-adic variable*, Journal für die reine und angewandte Mathematik **199** (1958), 23–34.

[3] N. P. Smart, *The discrete logarithm problem on elliptic curves of trace one*, Journal of Cryptology **12** (1999), no. 3, 193–196.