Mathematical Explanations for Elliptic Curves Project

Lucas Ashbury-Bridgwood

May 6, 2015

1 Addition of Points

1.1 Addition of Infinite Points

Let P and Q be points where at least one of which is infinite. The sum $P + Q$ is computed as follows. WLOG assume $P = \infty$. If $Q = \infty$ also, then $\infty = \infty + \infty = P + Q$. Otherwise Q is finite, in which case $Q = \infty + Q = P + Q$, ∞ being an additive identity.

1.2 Addition of Finite Points

Let P and Q be finite points. The sum $P + Q$ is computed as follows. First it is checked whether P and Q come from the same curve. If they do not, the program does not compute anything. Suppose the points come from the same curve. The program then follows the algorithm outlined in Introduction to Cryptography with Coding Theory (2nd Ed.) by Trappe and Washington on p. 352:

1. Let the common curve be $y^2 \equiv x^3 + bx + c \mod n$ for $b, c \in \mathbb{Z}_{\geq 0}$ and $n \in \mathbb{Z}_{\geq 1}$, and let $P = (x_1, y_1)$ and $Q = (x_2, y_2)$
2. Compute Δy and Δx:
   (a) If $P \neq Q$, compute $\Delta y \equiv y_2 - y_1 \mod n$ and $\Delta x \equiv x_2 - x_1$
   (b) If $P = Q$, compute $\Delta y \equiv 3{x_1}^2 + b \mod n$ and $\Delta x \equiv 2y_1$
3. If $\Delta x \equiv 0 \mod n$, then $P + Q = \infty$ and we are done
4. Otherwise, check if Δx has a multiplicative inverse $\mod n$, which is equivalent to checking if $\gcd(\Delta x, n) = 1$. If there is no inverse, then the sum is not defined on the curve are we are done
5. Otherwise compute $m \equiv \Delta y / \Delta x \mod n \equiv \Delta y \cdot \Delta x^{-1}$, the slope
6. Compute $x_3 \equiv m^2 - x_1 - x_2 \mod n$, $y_3 \equiv m(x_1 - x_3) - y_1$, and return the (finite) point $P + Q = (x_3, y_3)$

2 Scaling of Points (nP)

Let P be a point, finite or infinite, and let $n \in \mathbb{Z}_{\geq 0}$ be a scaling integer . nP is computed as follows. The programs checks the trivial cases first: If $P = \infty$ then ∞ is returned, if $n = 0$ then ∞ is returned, and if $n = 1$ then P is returned.[1] Suppose $n \geq 2$. In this case the program essentially writes n in binary as

$$n = 2^{x_1} + \cdots + 2^{x_k}$$

for some $x_1, \ldots, x_k \in \mathbb{Z}_{\geq 0}$ such that $x_1 < \cdots < x_k$ and some $k \in \mathbb{Z}_{\geq 1}$ (since $n \geq 2$), and computes

$$nP = 2^{x_1}P + \cdots + 2^{x_k}P$$

This is done by:

1. Computing $2^{x_1}P$ by successive doubling of P, i.e. computing $2P$, $2\,(2P)$, until $2^{x_1}P$ is reached
2. Then computing $2^{x_2-x_1}\,(2^{x_1}P) = 2^{x_2}P$ by successive doubling of $2^{x_1}P$, i.e. computing $2\,(2^{x_1}P)$, $2\,(2\,(2^{x_1}P))$, until $2^{x_2-x_1}\,(2^{x_1}P)$ is reached
3. Then computing $2^{x_3-x_2}\,(2^{x_2}P) = 2^{x_3}P$ likewise
4. Continuing until up to and including $2^{x_k}P$ is computed
5. Summing up $2^{x_1}P + \cdots + 2^{x_k}P$ pairwise from left to right, i.e. computing $2^{x_1}P + 2^{x_2}P$, then computing $(2^{x_1}P + 2^{x_2}P) + 2^{x_3}P$, etc.

The program does *not* implement this algorithm exactly: Both N is written in binary and the sum is computed *progressively* over a while loop, vs for example N being written in binary in one instant. A few remarks:

1. Scaling of points nP for $n \geq 2$ essentially uses only pairwise addition of points.
2. Successive scaling means that in computing nP for $n \geq 2$, if the highest power of 2 with n written in binary like above is M, then $2P, 2^2P, \ldots, 2^MP$ are all and only calculated.

[1] Is it true that $0 \cdot \infty = \infty$, and that $(0,0) = \infty$? Trappe and Washington on p. 352 explain that the points on such an elliptic curve form an abelian group with addition where ∞ is the identity, so by uniqueness should not $\infty = (0,0)$, $(0,0)$ also being an additive identity?