# Compact Representation of Elliptic Curve Points over $F_{2^n}$

Gadiel Seroussi
Computer Systems Laboratory
HPL-98-94 (R.1)
September, 1998

finite fields,
elliptic curves,
cryptography

A method is described to represent points on elliptic curves over $F_{2^n}$, in the context of elliptic curve cryptosystems, using $n$ bits. The method allows for full recovery of the $x$ and $y$ components of the point. This improves on the naive representation using $2n$ bits and on a previously known compressed representation using $n + 1$ bits. Since $n$ bits are necessary to represent a point in the general case of a cryptosystem over $F_{2^n}$, the representation described in this note is minimal.

Internal Accession Date Only

# 1 Background

Elliptic curve (EC) cryptography is gaining favor as an efficient and attractive alternative to the more conventional public key schemes, e.g., RSA.

EC cryptosystems are based on operations involving points on an *elliptic curve* over a finite field. Popular choices for the underlying finite field are $F_p$, the integers modulo $p$ for a (large) prime number $p$, and $F_{2^n}$, a finite field of characteristic two and dimension $n$. This note focuses on the latter type of field. The following paragraphs give background on finite fields and elliptic curves just sufficient to describe the proposed method. For more detailed background, see, for instance, [1] and the extensive bibliography therein.

The elements of $F_{2^n}$ are represented by binary vectors of length $n$. Addition in the field is a bitwise "exclusive or" operation, and field multiplication, in one of its possible forms, can be implemented as multiplication of binary polynomials of degree at most $n-1$ modulo a binary irreducible polynomial of degree $n$ (this is referred to as a *polynomial* representation).

A *point* is a pair $(x, y)$ of elements of $F_{2^n}$. The points of interest for the EC cryptosytem will be those satisfying an equation of the form

$$y^2 + xy + x^3 + a_2 x^2 + a_6 = 0, \tag{1}$$

where $a_2$ and $a_6$ are fixed elements of $F_{2^n}$, and all operations are over $F_{2^n}$. Points in $F_{2^n}^2$ satisfying Equation (1), together with a postulated *point at infinity* are referred to as *rational points* on the elliptic curve. Since the underlying field is finite, the number of rational points on the curve is finite, and will be denoted by $|E|$.[1]

It turns out that an *addition* operation can be defined on the elliptic curve points, and that the points, together with this operation, form an abelian group. By the Hasse Theorem, the size of the abelian group is known to fall in the interval

$$q + 1 - 2\sqrt{q} \leq |E| \leq q + 1 + 2\sqrt{q}, \tag{2}$$

where $q = 2^n$. For an effective elliptic curve cryptosystem, the coefficients $a_2$ and $a_6$ are chosen so that the elliptic curve group has a large cyclic subgroup of prime size $p$, i.e., $|E|$ can be written as $|E| = s \cdot p$, where $s$ is a small integer and $p$ is prime. A point $P$ of order $p$ is then chosen as the generator of the cyclic subgroup, and all EC cryptographic protocols are based on computing points of the form

$$kP = \underbrace{P + P + P + \cdots + P}_{k \text{ times}}.$$

[1] In the more general theory, points over the *algebraic closure* $\overline{K}$ of $F_{2^n}$ are considered, and the set of curve points with coordinates in an intermediate field $K$, $F_{2^n} \subseteq K \subseteq \overline{K}$ is generally denoted by $E(K)$. For $K = \overline{K}$, this set is infinite. In this note, we restrict our attention to the points in $E(F_{2^n})$, and we drop the dependency on the field from the notation.

With well chosen curve parameters, if $k$ and $P$ are given it is fairly easy to compute $kP$, but the inverse problem, i.e. recover $k$ from $P$ and $kP$ is computationally unfeasible as per current algorithmic knowledge. This inverse problem is known as the elliptic curve variant of the *discrete logarithm problem.*

The *trace* of $x \in F_{2^n}$ over $F_2$ is defined by

$$T(x) = \sum_{i=0}^{n-1} x^{2^i}.$$

It is well known that $T(x) \in \{0, 1\}$ for all $x \in F_{2^n}$, and that the trace is a linear operator, i.e., $T(a + b) = T(a) + T(b)$. Also, for all $x \in F_{2^n}$, $T(x^2) = T(x)$.

Dividing Equation (1) by $x^2$, and writing $z = y/x$, we obtain

$$z^2 + z + x + a_2 + \frac{a_6}{x^2} = 0. \tag{3}$$

It is known that this equation has a solution in $z$ if and only if $T(x + a_2 + a_6/x^2) = 0$. If $z_0$ is such a solution, then $z_0 + 1$ is also a solution. In terms of the original equation, if $y$ is a solution for a given $x$, then so is $y + x$.

In the prior art, a *compressed* representation of rational points is defined [2], based on the observation that given the $x$ coordinate of a point $(x, y)$, the $y$ coordinate can be obtained by solving the quadratic equation (1) in $y$, or its equivalent (3) in $z$. Such a quadratic equation will have two solutions in general. Therefore, a bit is necessary to specify which solution corresponds to the point $(x, y)$ at hand (the second solution corresponds to the point $(x, x + y)$). Thus, the point representation requires $n + 1$ bits ($n$ for $x$, and 1 to break the ambiguity in $y$), as opposed to $2n$ bits in a straightforward representation.

In this note, we observe that in fact, only $n - 1$ bits are required to describe $x$ for rational points used in an EC cryptosystem. Thus, a point can be represented in $n$ bits, in a form that allows full recovery of $x$ and $y$. The advantages of this approach are the obvious savings in representation length, and the fact that the entities stored and transmitted in the cryptosystem (field elements, points) can all be represented with $n$-bit vectors, without need for "odd pieces" (e.g. the extra bit in compressed point representation, which was implemented in [2] to occupy a full byte).

# 2 Compact representation

The proposed representation is based on the following facts:

1. Given an arbitrary point $P = (x_1, y_1)$, the point $2P = P + P = (x_2, y_2)$ satisfies

$$x_2 = x_1^2 + \frac{a_6}{x_1^2} \tag{4}$$

(see [1]).

2. Given that $P$ is a point on the curve, $z_1 = y_1/x_1$ and $x_1$ must satisfy (3). Therefore, we must have

$$T(x_1 + a_2 + \frac{a_6}{x_1^2}) = 0.$$

Recalling the properties of the trace operator, and the fact that $-a = a$ for all $a \in F_{2^n}$, the last equation implies

$$T(x_1^2 + a_2 + \frac{a_6}{x_1^2}) = 0,$$

or equivalently, using also (4),

$$T(a_2) = T(x_1^2 + \frac{a_6}{x_1^2}) = T(x_2).$$

Now, $P$ is a generic point, so the last equation implies that the $x$ coordinate of any point of the form $2P$ must satisfy

$$T(x) = T(a_2). \tag{5}$$

3. We now recall that most common protocols in an EC cryptosystem use points belonging to a cyclic subgroup of order $p$ of the curve group, where $p$ is a large prime and thus odd. Therefore, *every* point $Q$ in the cyclic subgroup can be written as $Q = 2P$ for some other point $P$ in the subgroup and, hence, the constraint (5) is satisfied by all the points of interest.

4. The coefficient $a_2$ is part of the definition of the cryptosystem, and therefore known to all parties before any meaningful use of curve points can be made. Therefore, $T(a_2)$ is a binary constant.

5. The trace operator, being linear, can be implemented as an inner product

$$T(x) = \mathbf{t} \cdot \mathbf{x} = \sum_{i=0}^{n-1} t_i x_i, \tag{6}$$

where $\mathbf{x} = (x_0, x_1, \ldots, x_{n-1})$ denotes the $n$-dimensional binary vector representing $x$, and $\mathbf{t}$ is and $n$-dimensional binary vector with entries given by $t_i = T(\alpha_i)$, where $[\alpha_0\, \alpha_1\, \cdots\, \alpha_{n-1}]$ is the basis used to represent $F_{2^n}$ over $F_2$. Thus, $\mathbf{t}$ is easy to compute a priori, and it is guaranteed that $t_i = 1$ for at least one $i$, $0 \leq i \leq n-1$.

6. To represent $x$ in $n-1$ bits, knowing that it satisfies (5), we choose a coordinate $i$ such that $t_i = 1$, and we eliminate that coordinate from $\mathbf{x}$. To reconstruct $\mathbf{x}$, the missing coordinate is uniquely recovered by forcing $\mathbf{x}$ to satisfy

$$\mathbf{t} \cdot \mathbf{x} = b,$$

where $b = T(a_2)$. The coordinate "punctured" from $\mathbf{x}$ can be used to accomodate the bit necessary to specify $y$. In polynomial representation, when $n$ is odd, it is known that $t_0 = 1$. This is true also in the case of *normal bases*, another popular field representation, where $\mathbf{t} = [1\,1\,\ldots\,1]$. In those cases we can choose $i = 0$ and puncture the first bit position in $\mathbf{x}$.

Finally, it is known that $|E|$ satisfies $|E| \equiv 2b \bmod 4$ (where, in an abuse of notation, $b$ is regarded as an integer). Thus, the maximum possible value of $p$ is $|E|/2$, and indeed subgroups attaining this maximum exist for certain values of $a_2$ and $a_6$. Recalling that $|E|$ falls in the interval given by (2), it follows that values of $p > q/2 = 2^{n-1}$ are possible, as shown in the example below. It follows that $n$ bits are *necessary* to represent a curve point in a cryptosystem supporting all possible values of $n$, $a_2$, and $a_6$ (a further reduction of one bit could be possible in principle for the case $T(a_2) = 0$, although no efficient method to obtain such a reduction is known).

**Example.** Let $n = 163$, and let $f(x) = x^{163} + x^7 + x^6 + x^3 + 1$ be the irreducible polynomial used to represent $F_{2^{163}}$. Let $a_2 = 1$ and

$$a_6 = 6\text{DBA}33035286\text{FB}596884\text{FC}7\text{B}3148\text{D}5\text{B}2\text{A}0\text{F}180\text{F}76,$$

where hexadecimal notation is used in the natural way to group bits, and bits are ordered left-to-right from most significant (coefficient of $x^{162}$) to least significant (coefficient of $x^0$). The corresponding elliptic curve $E$ has order

$$|E| = 11692013098647223345629480233048147171889149744282 = 2p,$$

where

$$p = 5846006549323611672814740116524073585944574872141$$

is a prime satisfying $p = 2^{162} + 7856589415073208447002 37$.

The question of whether an infinite sequence of values of $n$ exists for which curves with $p > 2^{n-1}$ can be found is open, and related to the (hard) question of whether there is an infinite sequence of primes in the set

$$\bigcup_j \left\{ i \,|\, 2^j + 1 - 2\sqrt{2^j} \le 2i \le 2^j + 1 + 2\sqrt{2^j} \right\},$$

see [3]. The answer to the question is conjectured to be positive.

# 3 References


[1] Alfred J. Menezes, *Elliptic curve cryptosystems*, Kluwer Academic Publishers, 1993.

[2] *IEEE P1363 Standard for RSA, Diffie-Hellman and Related Publ ic-Key Cryptography - Elliptic Curve Systems*, February 6, 1997.

[3] N. Koblitz. Constructing elliptic curve cryptosystems in characteristic 2. In *Advances in Cryptology, CRYPTO 90*, A.J. Menezes and S.A. Vanstone, editors, Springer Verlag, LNCS 537, 1991, pp. 156–167.