A Study of Koblitz Curves

Tawny Lim

Department of Computer Science,
University of California, Santa Barbara, CA 93106
E-mail: tlim@cs.ucsb.edu

Abstract— **Koblitz curves are a type of elliptic curve that are defined over GF(2). These curves are advantageous in that they can be used to create point multiplication algorithms without the need for point doubling [2]. This paper aims to understand the uses of Koblitz curves in cryptography, including their advantages and disadvantages. We will begin by analyzing the properties of Koblitz curves. We will then look into the advantages of Koblitz curves, mainly, the speeding up of scalar multiplications [4], and later examine the methods proposed by Solinas to further enhance the execution speeds of point multiplication on Koblitz curves [6]. Afterwards, we will delve into a brief discussion concerning the current state of security in using curves on binary fields and the possible risks of using Koblitz curves in the future [5].**

I. Introduction

In 1991, Neal Koblitz presented what he called the *anomalous binary curves*

$$\begin{aligned} E : y^2 + xy &= x^3 + x^2 + 1 \\ \tilde{E} : y^2 + xy &= x^3 + 1 \end{aligned} \tag{1}$$

defined over GF(2) [4]. These curves were discovered by Koblitz to have the efficient property that scalar multiplications $[k]P$ could be handled using only point additions when the binary representation of k contained less than or equal to four zeros [3]. Koblitz further found that for these curves over fields GF(2^m), where $m \leq 4$, $[k]P$ could be calculated using only point additions when k had binary representations with two to four zeros [3]. These curves henceforth became known as Koblitz curves.

Since their introduction, the use of Koblitz curves have been continuously expanded and improved upon. Currently, Koblitz curves in the field GF(2^k) are assumed to have k defined as a prime number in order for the curves to have orders that are either prime or very nearly prime [2], [6]. This prevents the curves from being broken by common algorithms.

With the knowledge and technology at the time of this discovery, the Diffie-Hellman cryptosystem that resulted from these Koblitz curves were considered secure [3], and over the years, many algorithms arose to further increase the efficiency of the scalar multiplications [6].

II. Efficiency

A *Frobenius map* is a map τ on a field of q elements, GF(q), such that $\tau(x, y) = (x^q, y^q)$. If the trace of the Frobenius map of an elliptic curve is equal to 1, we call the curve *anomalous*. τ then satisfies the characteristic equation $T^2 - T + q = 0$. Thus, when $q = 2^k$, the Frobenius map becomes $\tau : (x, y) \rightarrow (x^{2^k}, y^{2^k})$. From here, we see that scalar multiplication by a factor of 2^k can be greatly simplified by using $\tau^2 - \tau + 2^k = 0$:

$$\tau^2(P) - \tau(P) + 2^k(P) = 0$$

becomes

$$2^k P = -\tau(P) + \tau^2(P)$$

reducing the k point additions to just one point addition [3].

From here, Koblitz presents the following theorem:

Theorem 1: Let E be an anomalous elliptic curve defined over F_q, and let $\tilde{E}$ be its twist.

(a) If P is an $\mathbf{F_{q^n}}$-point on E (or $\tilde{E}$), then the multiple $[q]P$ can be computed with a single addition of points (together wth shift operations for the computation of $x \rightarrow x^q$ in a normal basis of $\mathbf{F_q^n}$).

(b) In the special case $q = 2$, any of the multiples $[2^l]P$ for $l \leq 4$ can be computed with a single addition of points.

With this result, point doubling can be conducted almost for free [3]. Furthermore, for points in GF(2^k), the Frobenius map on E is $\tau = \frac{1+\sqrt{-7}}{2}$. Then as τ is an element of norm 2 in the ring $\mathbf{Z}[\frac{1+\sqrt{-7}}{2}]$, any element of the ring can be written as a linear combination of τ^j, with coefficients $\{0, 1\}$. Any integer n is an element of this ring, and thus, a linear combination of the τ^j with Hamming size less than s can be chosen for n. From here, we have that scalar multiplication $[n]P$ can be reduced to less than s point additions:

$$[n]P = \sum c_j \tau^j(P).$$

A similar argument can be made for $\tilde{E}$, but in this case, $\tau = \frac{-1+\sqrt{-7}}{2}$, so the expansion of n becomes a linear combination of τ^j, with coefficients $\{0, -1\}$ [3].

This expansion of n is known as the *τ-adic representation* [2].

III. Solinas: Further Efficiency

In 2000, a paper by Solinas proposed a method that improves the speed of scalar multiplication by 50% when compared to any previous version [6].

A. Areas of Improvement

Solinas highlights three improvements which are unique to the elliptic case:

1. Rather than using a random process, the curve and the base field on which it is defined can be selected to optimize the efficiency of scalar multiplication.
2. Because subtraction on elliptic curves is as efficient as addition, by allowing subtractions, we can replace the binary expansion of n seen in the previous section with a signed binary expansion, $n = \sum c_j \tau^j$, where $c_j = \{-1, 0, 1\}$.
3. Complex multiplication with algebraic integers can be used from a set of operations that comes with every elliptic curve over a finite field.

All three of these areas are used in the creation of Solinas' new algorithm [6].

B. Algorithm

B.1 Nonadjacent Forms

The *nonadjacent form* (NAF) of n is a signed binary expansion with the property that two consecutive coefficients cannot be nonzero. Every integer n can be represented with a unique NAF, and there are several algorithms to find the NAF of n from its binary expansion [6]. The use of NAF's rather than ordinary binary expansion signifcantly reduces the number of terms required for integer expansion.

One method of finding the NAF involves repeated division by 2, allowing remainders $\{0, \pm 1\}$ chosen such that the quotient is even.

ROUTINE 4 (NAF)

Input:
a positive integer n

Output:
NAF(n)

Computation:
Set $c \leftarrow n$
Set $\mathcal{S} \leftarrow \langle\rangle$
While $c > 0$
  If c odd
    then
      set $u \leftarrow 2 - (c \bmod 4)$
      set $c \leftarrow c - u$
    else
      set $u \leftarrow 0$
  Prepend u to $\mathcal{S}$
  Set $c \leftarrow c/2$
EndWhile
Output $\mathcal{S}$

Fig. 1: An algorithm to compute the NAF

However, the computation of the NAF can be made more efficient by the proposed methods in II.A. For example, for NAF's with length l, the following addition-subtraction method

ROUTINE 6 (ADDITION-SUBTRACTION METHOD)

Input:
a positive integer n
an elliptic curve point P

Output:
the point nP

Computation:
Set $c \leftarrow n$
Set $Q \leftarrow \mathcal{O}$, $P_0 \leftarrow P$
While $c > 0$
  If c odd then
    set $u \leftarrow 2 - (c \bmod 4)$
    set $c \leftarrow c - u$
    if $u = 1$ then set $Q \leftarrow Q + P_0$
    if $u = -1$ then set $Q \leftarrow Q - P_0$
  Set $c \leftarrow c/2$
  Set $P_0 \leftarrow 2P_0$
EndWhile
Output Q

Fig. 2: Addition-Subtraction computation of the NAF

has an average density of nonzero coefficients of

$$\frac{2^l(3l-4) - (-1)^l(6l-4)}{9(l-1)(2^l - (-1)l)} \approx \frac{1}{3} \tag{2}$$

compared to the binary expansions, which have an average density of about $\frac{1}{2}$, so we see that binary methods are less efficient, requiring about 12% more elliptic operations than addition-subtraction [6].

B.2 τ-adic NAF

In [6], Solinas takes this a step further and introduces the τ-adic NAF as an alternative to the normal NAF we saw in Figure 1. By replacing a (signed) binary expansion with a (signed) τ-adic expansion, n can be represented as the sum and difference of powers of τ, and when GF(2^k) is represented as a normal basis, multiplication by τ is essentially free [6].

As an example, given a point $P = (x, y)$ on E and an integer $n = 9$, the scalar multiplication $[9]P$ can be computed using the representation of 9 as its τ-adic NAF. That is, $9 = \tau^5 - \tau^3 + 1$, so the scalar multiplication can be computed as follows [6]:

$$[9]P = (x^{2^5}, y^{2^5}) - (x^{2^3}, y^{2^3}) + (x, y)$$

By modifying the original NAF algorithm in Figure 1, we obtain the algorithm to compute the τ-adic NAF seen in Figure 3. Unlike the original NAF algorithm where the NAF was obtained through repeated division by 2, here we divide repeatedly by τ. The only possible remainders after this division are 1 or -1, so we choose the value that allows the quotient to be divisible by τ [6].

ALGORITHM 1 *(τ-adic NAF)*

Input:
integers r_0, r_1

Output:
TNAF$(r_0 + r_1\,\tau)$

Computation:
```
Set c0 ← r0,  c1 ← r1
Set S ← ⟨⟩
While c0 ≠ 0 or c1 ≠ 0
    If c0 odd
        then
            set u ← 2 − (c0 − 2 c1 mod 4)
            set c0 ← c0 − u
        else
            set u ← 0
    Prepend u to S
    Set (c0, c1) ← (c1 + μ c0/2, −c0/2)
EndWhile
Output S
```

Fig. 3: Computation of the τ-adic NAF

To prove that the τ-adic NAF is more efficient than regular NAF, Solinas presented the following [2]:

Theorem 2: Let $\kappa \in \mathbf{Z}_\tau$, $\kappa \neq 0$.

If the length $l(\kappa)$ of the τ-adic NAF(κ) is greater than 30, then

$$\log_2(N(\kappa)) - 0.55 < l(\kappa) < \log_2(N(\kappa)) + 3.52.$$

In addition, as with addition-subtraction NAF's, the average density of nonzero coefficients for τ-adic NAF's of length l is given by (2), and thus can be approximated by $\frac{1}{3}$ [6].

B.3 Reduced τ-adic NAF

A problem that occurs when using the τ-adic NAF is that the number of nonzero terms, or *Hamming weight*, is twice as long as that of the ordinary NAF. Because of this, the advantages gained by eliminating point doublings is reduced by the doubling of the number of point additions. Then the τ-adic method is not as advantageous as it could be [6].

To fix this problem, Solinas replaces the τ-adic NAF with a *reduced* τ-adic NAF. The reduced τ-adic NAF is constructed so that it is equivalent to the ordinary τ-adic NAF (that is, for λ and ρ in $\mathbf{Z}[\tau]$, $\lambda P = \rho P$ for all P on the Koblitz curve), but only half as long [6].

The reduced τ-adic NAF can then be applied to improve the efficiency of scalar multiplication on curves in GF(2^k). Then the algorithm, seen in Figure 4, requires only $k/3$ point additions and no point doubling. This makes the algorithm at least 50% faster than the previous algorithms [6].

ALGORITHM 3 *Scalar Multiplication on Koblitz Curves*

Per-Curve Parameters:
m, a, s_0, s_1, r

Input:
n, *a positive integer less than* $r/2$
P, *a point in the main subgroup*

Output:
nP

Computation:
```
Compute (r0, r1) ← n mod δ      (via (74))
Set Q ← O
    P0 ← P
While r0 ≠ 0 or r1 ≠ 0
    If r0 odd then
        set u ← 2 − (r0 − 2 r1 mod 4)
        set r0 ← r0 − u
        if u = 1 then set Q ← Q + P0
        if u = −1 then set Q ← Q − P0
    Set P0 ← τ P0          (=RightShift[P0])
    Set (r0, r1) ← (r1 + μ r0/2, −r0/2)
EndWhile
Output Q
```

Fig. 4: Algorithm for Scalar Multiplication

IV. Current State

With the significant improvements in efficiency that Solinas created, Koblitz curves became "so nice to work with that they reportedly became known as 'magic curves' within the U.S. National Security Agency" [4]. In fact, of all the special classes of elliptic curves, only Koblitz curves are approved for practical use by major industrial standards [4], and until recently, the NSA supported the use of the five Koblitz curves listed in [7]: K-163, K-233, K-283, K-409, and K-571.

However, there is an increasing concern over the quantum computer, as well as a necessity for quantum-safe cryptography that cannot be solved with a quantum computer. Due to this, in August 2015, the NSA announced plans to transition away from ECC and into quantum-resistant algorithms. Until these new algorithms are developed, the current algorithms, known as Suite B, will continue to be used [8].

In addition to this, the long-term security of Koblitz curves has become questionable [5]. In 2014, Galbraith and Gebregiyorgis presented their approach to using summation-polynomial methods to attack the ECDLP on curves over binary fields [1]. Because of the progress they made in breaking the cryptographic system, curves defined over prime fields have become safer choices for use over curves defined on binary fields [5].

V. Conclusion

We see that Koblitz curves are a special class of curves defined over binary fields and allow for greatly improved efficiency through their ability to implement scalar multiplications by using only point additions. Due to this, Koblitz curves have been popular and there have been many successful attempts to further increase their efficiency.

However, due to recent progress in cracking curves over binary fields and increasing concern over the advent of the quantum computer, the confidence towards the long-term security of Koblitz curves is waning. Sentiment is shifting away from supporting the use of these curves, and once the NSA releases its new suite of quantum-resistant algorithms, we may see that the use of elliptic curves, and thus Koblitz curves, disappears altogether.

References

[1] S. Galbraith and S. Gebregiyorgis, "Summation polynomial algorithms for elliptic curves in characteristic two." *Progress in Cryptology INDOCRYPT*. LNCS 8885 (2014): 409-427.

[2] Hankerson, Darrel, Menezes, Alfred, and Vanstone, Scott. *Guide to Elliptic Curve Cryptography.* New York: Springer, 2004.

[3] Koblitz, Neal. "CM curves with good cryptographic properties." *Proc. Crypto* (1991): 279-287.

[4] Koblitz, Neal. "Good and Bad Uses of Elliptic Curves in Cryptography." *Moscow Mathematical Journal* 2.4 (Oct.-Dec. 2002): 693-715.

[5] Koblitz, Neal, and Menezes, Alfred J. "A Riddle Wrapped in an Enigma." (2015).

[6] Solinas, Jerome A. "Efficient Arithmetic on Koblitz Curves" *Designs, Codes and Cryptography* 19 (2000): 195-249.

[7] "Recommended Elliptic Curves for Federal Government Use." (1999).

[8] NSA. 'NSA Suite B Cryptography' (2015) Web. `https://www.nsa.gov/ia/programs/suiteb_cryptography/`