Optimal Tower Fields

Selçuk Baktir, *Student Member*, *IEEE*, and Berk Sunar, *Member*, *IEEE*

Abstract—We introduce a new tower field representation, *optimal tower fields* (OTFs), that facilitates efficient finite field operations. The recursive direct inversion method we present has significantly lower complexity than the known best method for inversion in optimal extension fields (OEFs), i.e., Itoh-Tsujii's inversion technique. The complexity of our inversion algorithm is shown to be $O(m^2)$, significantly better than that of the Itoh-Tsujii algorithm, i.e., $O(m^2(\log_2 m))$. This complexity is further improved to $O(m^{\log_2 3})$ by utilizing the Karatsuba-Ofman algorithm. In addition, we show that OTFs may be converted to OEF representation via a simple permutation of the coefficients and, hence, OTF operations may be utilized to achieve the OEF arithmetic operations whenever a corresponding OTF representation exists. While the original OTF multiplication and squaring operations require slightly more additions than their OEF counterparts, due to the free conversion, both OTF operations may be achieved with the complexity of OEF operations.

Index Terms—Optimal tower fields, OEF, finite fields, multiplication, inversion, elliptic curve cryptography.

1 Introduction

Elliptic curve cryptography relies on efficient algorithms for finite field arithmetic. For instance, the elliptic curve digital signature algorithm requires efficient addition, multiplication, and inversion in finite fields of size larger than 2^{160}. This poses a significant problem in embedded systems where computational power is quite limited and public-key operations are unacceptably slow [1]. Hence, efficient algorithms for finite field operations have been closely investigated [2], [3]. Besides the standard basis, alternative representations such as the normal bases [4], [5], [6] and dual bases [4], [7], [8], [9] have been studied. *Optimal Extension Fields* [10], [11] have been found to be especially successful in embedded software implementations of elliptic curve schemes. The arithmetic operations in OEFs are much more efficient than in characteristic two extensions or prime fields due to the use of a large characteristic ground field and the selection of a binomial as the field polynomial. In the elliptic curve scalar-point multiplication, a large number of field multiplications and inversions are computed. Inversion is inherently more complex and at least several times more costly than multiplication. Hence, despite recent improvements, inversion is still the slowest operation in elliptic curve implementations. In this paper, we address this issue by introducing a specialized tower field representation.

2 Background

2.1 Optimal Extension Fields and Their Arithmetic

Optimal extension fields were introduced by Bailey and Paar in [11]. The main idea is to use a generating polynomial of the form $P(x) = x^m - w$ to construct the extension field $GF(p^m)$, where p is selected as a *pseudo-Mersenne prime* given in the form $2^k \pm c$ with $\log_2 c < \lfloor \frac{k}{2} \rfloor$. The pseudo-Mersenne form allows efficient reduction in the ground field operations. The following theorem [12] provides a simple means to identify irreducible binomials that can be used in OEF construction:

Theorem 1. *Let $m \geq 2$ be an integer and $w \in GF(q)^*$. Then, the binomial $x^m - w$ is irreducible in $GF(q)[x]$ if and only if the following three conditions are satisfied:*

1. *Each prime factor of m divides the order e of w in $GF(q)^*$;*
2. *The prime factors of m do not divide $\frac{q-1}{e}$;*
3. *$q = 1 \bmod 4$ if $m = 0 \bmod 4$.*

The representation of OEF elements utilizes the standard basis. An element $A \in GF(p^m)$ is represented as

$$A = \sum_{i=0}^{m-1} a_i x^i = a_0 + a_1 x + a_2 x^2 + \ldots + a_{m-1} x^{m-1},$$

where $a_i \in GF(p)$. The OEF arithmetic operations are performed as follows.

Addition/Subtraction. The addition/subtraction of two field elements $A, B \in GF(p^m)$ is performed in the usual way by adding/subtracting the polynomial coefficients in $GF(p)$ as follows:

$$A \pm B = \sum_{i=0}^{m-1} a_i x^i \pm \sum_{i=0}^{m-1} b_i x^i = \sum_{i=0}^{m-1} (a_i \pm b_i) x^i.$$

Multiplication. Let $A, B \in GF(p^m)$. Their product $C = A \cdot B$ is computed in two steps:

1. Polynomial multiplication:

$$C' = A \cdot B = \sum_{i=0}^{2m-2} c'_i x^i.$$

• *The authors are with the Electrical and Computer Engineering Department, Worcester Polytechnic Institute, 100 Institute Rd., Worcester, MA 01609. E-mail: {selcuk, sunar}@wpi.edu.*

Manuscript received 19 Mar. 2003; revised 8 Dec. 2003; accepted 19 Mar. 2004.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number 118467.

2. Modular reduction:

$$\begin{aligned} C &= C'(\bmod P(x)) \\ &= \sum_{i=0}^{2m-2} c'_i x^i (\bmod x^m - w) \\ &= \sum_{i=0}^{m-1} (c'_i + w c'_{i+m}) x^i. \end{aligned}$$

Here, we set $c'_{2m-1} = 0$. In the first step, the ordinary product of two polynomials is computed. In the reduction step, the binomial $P(x) = x^m - w$ facilitates efficient reduction. The reduction may be realized by only $m-1$ constant coefficient multiplications by w, and $m-1$ additions.

Inversion. An elegant method for inversion was introduced by Itoh and Tsujii [5]. Let $A \in GF(p^m)$. We want to find $B = A^{-1} \bmod P(x)$. The algorithm works in four steps:

1. Compute the exponentiation A^{r-1} in $GF(p^m)$, where $r = \frac{p^m-1}{p-1}$;
2. Compute the product $A^r = (A^{r-1}) \cdot A$;
3. Compute the inversion $(A^r)^{-1}$ in $GF(p)$;
4. Compute the product $A^{r-1} \cdot (A^r)^{-1} = A^{-1}$.

For the particular choice of

$$r = \frac{p^m - 1}{p - 1},$$

A^r belongs to the ground field $GF(p)$ [12]. This allows the inversion in Step 3 to be computed in the ground field $GF(p)$ instead of the larger field $GF(p^m)$. For the exponentiation A^{r-1} in Step 1, we expand the exponent $r-1$ as follows:

$$r - 1 = \frac{p^m - 1}{p - 1} - 1 = p^{m-1} + p^{m-2} + \ldots + p^2 + p.$$

The exponentiation requires the computation of powers A^{p^i} for $1 \le i \le m-1$. The original Itoh-Tsujii algorithm proposes to use a normal basis representation over $GF(2)$ which turns these exponentiations into simple bitwise rotations. In [13], this technique was adapted to work efficiently in the standard basis as well. In the same reference, it was shown that A^{r-1} can be computed by performing

$$\#MUL = \lfloor \log_2(m-1) \rfloor + HW(m-1) - 1$$

multiplications and

$$\#EXP = \lfloor \log_2(m-1) \rfloor + HW(m-1)$$

p^ith power exponentiations in $GF(p^m)$, where $HW(m)$ denotes the Hamming weight of m. For the details of this algorithm, we refer the reader to [13]. Instead, we briefly outline how A^{p^i} can be computed efficiently for the special case of OEFs. We continue to draw from [13], which considers the computation of A^{p^i} in the standard basis. A^{p^i} is the ith iterate of the *Frobenius map* defined as $\sigma(A) = A^p$. We will make use of the following properties of the Frobenius map:

- $\sigma(A+B) = \sigma(A) + \sigma(B)$ for any $A, B \in GF(p^m)$ (Linearity Property),
- $\sigma(a) = a^p = a$ for any $a \in GF(p)$ (Fermat's Little Theorem).

Using these rules, the exponentiation $A^{p^i} = \sigma^i(A)$ is simplified as

$$A^{p^i} = \left(\sum_{j=0}^{m-1} a_j x^j \right)^{p^i} = \sum_{j=0}^{m-1} (a_j x^j)^{p^i} = \sum_{j=0}^{m-1} a_j x^{jp^i}. \quad (1)$$

We focus on the powers x^{jp^i} for $0 < i, j \le m-1$ in the summation. The following theorem allows further simplification:

Theorem 2 [13]. *Let $P(x)$ be an irreducible polynomial of the form $P(x) = x^m - w$ over $GF(p)$, e an integer, $P(\alpha) = 0$, and it is understood that $p \ge 3$. Then,*

$$\alpha^e = w^t \alpha^s,$$

where $s = e \bmod m$ and $t = \frac{e-s}{m}$.

Hence, it is possible to precompute w^t and s for all values the exponent e takes in the summation in (1), i.e., $e = jp^i$ for $0 < i, j \le m-1$. Utilizing a lookup table with entries

$$c_j = w^{\frac{jp^i - (jp^i \bmod m)}{m}},$$

one may realize the exponentiation A^{p^i} using only $m-1$ constant coefficient multiplications required to compute the terms $c_j a_j$ for $0 < j \le m-1$ and some additions for properly collecting the reduced terms of the polynomial in (1). The lookup table is relatively small in size since, for most OEFs, m is small and $c_i \in GF(p)$. The following claim was made in [10] to further simplify the reduction.

Claim 1 [10].

$$(x^j)^{p^i} \bmod P(x) = w^t x^j,$$

where $x^j \in GF(p)[x]$, i is an arbitrary positive rational integer, and other variables are as defined in Theorem 2.

Flawed Proof. Since $P(x)$ is an irreducible binomial, by Theorem 1, $m|(p-1)$, which implies

$$p \bmod m = (p-1) + 1 \bmod m = 1.$$

Thus, $s \bmod m = jp^i \bmod m = j$. □

Claim 1 states that the positions of the terms in the summation in (1) stay fixed when the p^ith power is taken. This means that, in the summation, no two terms will have the same power and, therefore, no additions will be needed for the exponentiation. However, there is a flaw in the proof. The proof begins by assuming that m divides $p-1$ based on the first condition of Theorem 1. This will not always be correct. According to this condition, each prime factor of m has to divide the order of w. If m has repeated factors that the order of w does not have with the same multiplicity, m will not divide the order of w and, hence, will not divide $p-1$. Claim 1 may be fixed by explicitly requiring m to divide the order of w as follows.

Corollary 1. *If m divides the order of w, then*

$$(x^j)^{p^i} \bmod P(x) = w^t x^j,$$

where $x^j \in GF(p)[x]$, i is an arbitrary positive rational integer and other variables are as defined in Theorem 2.

Corollary 1 eliminates additions in the computation of A^{p^i}; however, it adds yet another restriction to OEF construction. We introduce the following theorem which shows that the exponentiation A^{p^i} may be achieved by a simple scaled permutation of the coefficients of the polynomial representation of A.

Theorem 3. *For an irreducible binomial $P(x) = x^m - w$ defined over $GF(p)$, the following identity holds for an arbitrary positive integer i and $A \in GF(p^m)$:*

$$A^{p^i} = \left(\sum_{j=0}^{m-1} a_j x^j\right)^{p^i} = \sum_{j=0}^{m-1} (a_j c_{s_j}) x^{s_j},$$

where $s_j = jp^i \bmod m$ and $c_{s_j} = w^{\frac{jp^i - s_j}{m}}$. Furthermore, the s_j values are distinct for $0 \le j \le m-1$.

Proof. Let $s_j = jp^i \bmod m$. Then, $m|(jp^i - s_j)$ and we can write

$$x^{jp^i} = (x^m)^{\frac{jp^i - s_j}{m}} x^{s_j} = w^{\frac{jp^i - s_j}{m}} x^{s_j}.$$

Assigning $c_{s_j} = w^{\frac{jp^i - s_j}{m}}$, we obtain the summation

$$A^{p^i} = \left(\sum_{j=0}^{m-1} a_j x^j\right)^{p^i} = \sum_{j=0}^{m-1} a_j x^{jp^i} = \sum_{j=0}^{m-1} (a_j c_{s_j}) x^{s_j}.$$

Next, we prove by contradiction that the s_j values are distinct for $0 \le j \le m-1$. Assume there is a collision $s_{j_1} = s_{j_2}$ with $0 \le j_1 \ne j_2 \le m-1$, then

$$j_1 p^i = j_2 p^i \pmod m$$
$$(j_1 - j_2) p^i = 0 \pmod m.$$

According to Theorem 1, all prime factors of m divide $p-1$. Thus, m and p are relatively prime and the above expression is satisfied only when $j_1 - j_2 = 0$, a contradiction. □

Using the method in Theorem 3, exponentiations of degree p^i may be implemented with the help of a lookup table of precomputed c_{s_j} values, using not more than $m-1$ constant coefficient multiplications and no additions.

2.2 Direct Inversion

A method for the direct computation of an inverse $B = A^{-1}$ in $GF(q^m)$ has long been used [14], [15]. The standard basis representation of $A \in GF(q^m)$ is given as $A(x) = \sum_{i=0}^{m-1} a_i x^i$, where $a_i \in GF(q)$. We consider the product $C(x) = A(x)B(x) = 1 \pmod{P(x)}$, where $P(x)$ denotes the irreducible field polynomial. This means the first coefficient of the product $A(x)B(x)$ is one and all other coefficients are zeros. By expressing the coefficients of the product in terms of the coefficients of A and B, a system of m linear equations is formed. Solving these equations for B's coefficients yields the inverse expressed in terms of the coefficients of A. The main advantage of using the Direct Inversion technique is that one can compute the inverse of an extension field element by doing operations only in the ground field $GF(q)$. We illustrate this technique with the following two examples:

Example 1. Direct Inversion in $GF(q^2)$. Let $A(x) \in GF(q^2)$ and $A(x) = a_0 + a_1 x$, where $a_0, a_1 \in GF(q)$, with the irreducible field polynomial selected as $P(x) = x^2 - p_0$, where $p_0 \in GF(q)$. Then,

$$\begin{aligned} A(x)B(x) \pmod{P(x)} &= (a_0 + a_1 x) \\ &\quad \cdot (b_0 + b_1 x) \pmod{P(x)} \\ &= (a_0 b_0 + p_0 a_1 b_1) \\ &\quad + (a_0 b_1 + a_1 b_0) x \\ &= 1. \end{aligned}$$

The coefficients yield the following system of equations:

$$\begin{pmatrix} a_0 & p_0 a_1 \\ a_1 & a_0 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Solving the system of equations gives

$$b_0 = a_0 \Delta^{-1} \quad \text{and} \quad b_1 = -a_1 \Delta^{-1}, \tag{2}$$

where $\Delta = a_0^2 - p_0 a_1^2$.

Example 2. Direct Inversion in $GF(q^3)$. Let $A(x) \in GF(q^3)$ and $A(x) = a_0 + a_1 x + a_2 x^2$, where $a_0, a_1, a_2 \in GF(q)$, with the irreducible field polynomial selected as $P(x) = x^3 - p_0$, where $p_0 \in GF(q)$. Then,

$$\begin{aligned} A(x)B(x) &= (a_0 + a_1 x + a_2 x^2)(b_0 + b_1 x + b_2 x^2) \pmod{P(x)} \\ &= a_0 b_0 + a_1 b_2 p_0 + a_2 b_1 p_0 \\ &\quad + (a_0 b_1 + a_1 b_0 + a_2 b_2 p_0) x \\ &\quad + (a_0 b_2 + a_1 b_1 + a_2 b_0) x^2 \\ &= 1. \end{aligned}$$

The coefficients yield the following system of equations:

$$\begin{pmatrix} a_0 & p_0 a_2 & p_0 a_1 \\ a_1 & a_0 & p_0 a_2 \\ a_2 & a_1 & a_0 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}.$$

Solving the system of equations gives

$$b_0 = (a_0^2 - a_1 a_2 p_0) \Delta^{-1}, \tag{3}$$

$$b_1 = (a_2^2 p_0 - a_0 a_1) \Delta^{-1}, \tag{4}$$

$$b_2 = (a_1^2 - a_0 a_2) \Delta^{-1}, \tag{5}$$

where

$$\Delta = a_0^3 - 3 a_0 a_1 a_2 p_0 + a_1^3 p_0 + a_2^3 p_0^2.$$

3 Optimal Tower Fields

A field obtained by repeatedly extending a ground field with a series of same degree irreducible polynomials is commonly referred to as a tower field. To construct a tower field $GF(q^{t^k})$, one needs k irreducible polynomials $P_i(x)$ for $0 < i \le k$, each of degree t and irreducible over $GF(q^{t^{i-1}})$.

The selection of these polynomials determines the representation and, thus, the efficiency of the field operations. In this paper, we limit our attention to a subclass of tower fields which we define as follows:

Definition 1. *An Optimal Tower Field (OTF) is a finite field $GF(q^{t^k})$ such that*

1. *q is a pseudo-Mersenne prime,*
2. *$GF(q^{t^k})$ is constructed by an ensemble of binomials $P_i(x) = x^t - \alpha_{i-1}$ irreducible over $GF(q^{t^{i-1}})$ with $P_i(\alpha_i) = 0$, for $0 < i \leq k$.*

The definition requires a set of related irreducible binomials. Before presenting an explicit construction, we first develop the following theorems.

Theorem 4. *For an OTF constructed using the binomials $P_i(x) = x^t - \alpha_{i-1}$, for $i > 0$, the roots α_i are related as*

$$\mathrm{ord}(\alpha_i) = t\ \mathrm{ord}(\alpha_{i-1}) = t^i \mathrm{ord}(\alpha_0),$$

where $\mathrm{ord}(a)$ denotes the order of field element a.

Proof. We consider the powers of α_i: $\alpha_i, \alpha_i^2, \alpha_i^3, \ldots, \alpha_i^t = \alpha_{i-1}$. Note that the first power of α_i that yields α_{i-1} is t. Likewise, the first power of α_{i-1} that will reach α_{i-2} is t. Hence, the t^2th power of α_i yields α_{i-2}. This process is repeated until α_0 is reached at the t^ith power of α_i. Since $\alpha_i^{t^i} = \alpha_0$, the first power of $\alpha_i^{t^i}$ that yields 1 is $\mathrm{ord}(\alpha_0)$. Hence, $\mathrm{ord}(\alpha_i) = t^i \mathrm{ord}(\alpha_0)$. □

Theorem 5. *If there exists an irreducible binomial $Q(x) = x^t - a$ over $GF(q)$, then $\frac{q^t-1}{q-1}$ is divisible by t.*

Proof. If $Q(x) = x^t - a$ is irreducible over $GF(q)$, then all three conditions of Theorem 1 are satisfied. We construct another binomial by choosing an arbitrary primitive element $a' \in GF(q)$: $P(x) = x^t - a'$. This binomial is irreducible over $GF(q)$ since the three conditions of Theorem 1 are satisfied, i.e.:

1. The order of a primitive element in $GF(q)$ is $q-1$. Since $\mathrm{ord}(a) | q-1$ and each prime factor of t divides $\mathrm{ord}(a)$, each prime factor of t also divides $\mathrm{ord}(a') = q-1$;
2. Prime factors of t do not divide $\frac{q-1}{e'} = 1$, where $e' = q-1$ is the order of a';
3. t and q are the same for $Q(x)$ and $P(x)$; since the condition $q = 1 \bmod 4$ if $t = 0 \bmod 4$ was satisfied for $Q(x)$, it will be satisfied for $P(x)$ as well.

Now, consider the root of $P(x) = x^t - a'$. According to Theorem 4, the root of $P(x)$ will have order $t(q-1)$. The order of the multiplicative subgroup of $GF(q^t)$ is $q^t - 1$. Thus, the order of the root of $P(x)$, i.e., $t(q-1)$, divides the multiplicative group order $q^t - 1$. Consequently, $\frac{q^t-1}{q-1}$ is divisible by t. □

Theorem 6. *Let $P(x) = x^t - a$ be an irreducible binomial over $GF(q)$ and t' denote the product of the prime factors of t. Then, $q^t = 1 \pmod{t't}$.*

Proof. Since $P(x)$ is given as irreducible, the first condition of Theorem 1, i.e., $t' | e$, is met. The order e always divides the group order $q-1$ and, therefore, $t' | q-1$. The binomial $P(x)$ is given as irreducible over $GF(q)$. Therefore, according to Theorem 5, $\frac{q^t-1}{q-1}$ is divisible by t or, equivalently, $t(q-1) | q^t - 1$. Hence, $t't | q^t - 1$ and $q^t = 1 \pmod{t't}$. □

The following theorem establishes the necessary and sufficient conditions for the existence of OTFs:

Theorem 7. *Given an irreducible binomial $P_1(x) = x^t - \alpha_0$ over $GF(q)$, all binomials of the form $P_i(x) = x^t - \alpha_{i-1}$ over $GF(q^{t^{i-1}})$, where $P_i(\alpha_i) = 0$ for $i > 0$, are also irreducible provided that none of the prime factors of t divides $\frac{q^t-1}{t(q-1)}$.*

Proof. We need to show that the conditions in Theorem 1 are valid for all binomials $P_i(x) = x^t - \alpha_{i-1}$ for $i > 1$. The first condition is always satisfied since all $\mathrm{ord}(\alpha_i)$ are multiples of t for $i > 0$ (Theorem 4). The third condition is also always satisfied since it was satisfied for the first irreducible polynomial, i.e., if $t = 0 \bmod 4$ and $q = 1 \bmod 4$, then always $q^n = 1 \bmod 4$. We will now show that, once the second condition is satisfied for the first irreducible binomial $P_1(x)$, it will be satisfied for all the other binomials, $P_i(x)$ for $i > 1$, provided that none of the prime factors of t divides $\frac{q^t-1}{t(q-1)}$. We use induction. We want to prove that, if no prime factor of t divides $\frac{q^{t^n}-1}{t^n \cdot \mathrm{ord}(\alpha_0)}$, then no prime factor of t divides $\frac{q^{t^{n+1}}-1}{t^{n+1} \cdot \mathrm{ord}(\alpha_0)}$. The latter is factorized as follows:

$$\frac{q^{t^{n+1}} - 1}{t^{n+1} \cdot \mathrm{ord}(\alpha_0)} = \frac{q^{t^n} - 1}{t^n \cdot \mathrm{ord}(\alpha_0)} \cdot \frac{\sum_{i=0}^{t-1} q^{it^n}}{t}. \tag{6}$$

The first factor is not divisible by any prime factor of t. The second factor must be shown to be indivisible as well. For this, we use the result of Theorem 6, i.e.,

$$q^t = 1 \pmod{t't}$$

to simplify the summation for $n > 0$ as follows:

$$\sum_{i=0}^{t-1} (q^t)^{it^{n-1}} = \sum_{i=0}^{t-1} (1)^{it^{n-1}} \pmod{t't} = t \pmod{t't}.$$

Hence,

$$\sum_{i=0}^{t-1} q^{it^n} = k't't + t$$

for some integer k' and

$$\frac{\sum_{i=0}^{t-1} q^{it^n}}{t} = k't' + 1$$

and, therefore,

$$\frac{\sum_{i=0}^{t-1} q^{it^n}}{t} = 1 \pmod{u}$$

for any prime factor u of t. Hence, $\frac{\sum_{i=0}^{t-1} q^{it^n}}{t}$ isn't divisible by any prime factor of t. The condition still remains for $n = 0$, i.e., $\frac{q^t-1}{t(q-1)}$ should not be divisible by any prime factor of t. □

Theorem 7 provides a simple means for checking the existence of OTFs for chosen values of q, t, and α_0. It should be noted that the existence condition is not dependent on k,

TABLE 1
n, c, α_0 Values for Building $GF(q^{2^k})$ OTFs with $q = 2^n + c$, $7 \leq n \leq 16$; $-5 \leq c \leq 5$; $-5 \leq \alpha_0 \leq 5$

n	c	α_0	n	c	α_0	n	c	α_0	n	c	α_0
8	1	−5	9	−3	2	11	5	2	14	−3	−2
8	1	−3	9	−3	3	11	5	5	14	−3	2
8	1	3	10	−3	−2	12	−3	−5	16	1	−5
8	1	5	10	−3	2	12	−3	−2	16	1	−3
9	−3	−3	11	5	−5	12	−3	2	16	1	3
9	−3	−2	11	5	−2	12	−3	5	16	1	5

which greatly simplifies the construction. Table 1 and Table 2 provide lists of practical OTF constructions for $GF(q^{2^k})$ and $GF(q^{3^k})$.

4 Conversion between OTFs and OEFs

An OTF $GF(q^{t^k})$ is isomorphic to an OEF $GF(q^m)$ if $m = t^k$. Before explaining how an associated OEF is obtained from a given OTF, we introduce the following theorem.

Theorem 8. *For a given OTF $GF(q^{t^k})$, if $t = 2 \bmod 4$, then $q = 1 \bmod 4$.*

Proof. From Theorem 5, we know that $\frac{q^t-1}{q-1}$ is divisible by t. The extension degree t is given as even ($t = 2 \bmod 4$). Likewise, $q-1$ is even since q is a prime. Therefore, $(q-1)t$ and $q^t - 1$ are divisible by 4. Since $q^t = 1 \bmod 4$, then either $q = 1 \bmod 4$ or $q = 3 \bmod 4$. We want to show that $q = 3 \bmod 4$ is never satisfied, hence, $q = 1 \bmod 4$.

If $t = 2 \bmod 4$, it can be written as $t = 4r + 2 = 2(2r + 1)$ for some integer $r \geq 0$. According to Theorem 7, none of the prime factors of t divides $\frac{q^t-1}{t(q-1)}$, hence, 2 does not divide

$$\begin{aligned}\frac{q^t-1}{t(q-1)} &= \frac{q^{2(2r+1)}-1}{2(2r+1)(q-1)} \\ &= \frac{(q^2-1)(\sum_{i=0}^{2r} q^{2i})}{2(2r+1)(q-1)} \\ &= \frac{(q+1)(\sum_{i=0}^{2r} q^{2i})}{2(2r+1)}.\end{aligned}$$

Therefore, 4 does not divide

$$\frac{(q+1)(\sum_{i=0}^{2r} q^{2i})}{2r+1},$$

and $q+1$ is not divisible by 4. Since $q \neq 3 \bmod 4$, it follows that $q = 1 \bmod 4$. □

Constructing an equivalent OEF representation from a given OTF representation is established by the following theorem.

Theorem 9. *For a given OTF representation of $GF(q^{t^k})$ with $P_i(x) = x^t - \alpha_{i-1}$ irreducible over $GF(q^{t^{i-1}})$ for $0 < i \leq k$, there exists an associated OEF representation with irreducible polynomial $Q(x) = x^m - w$ such that $Q(\alpha_k) = 0$, $m = t^k$, and $w = \alpha_0$.*

Proof. Consider the set of relations $P_i(\alpha_i) = \alpha_i^t - \alpha_{i-1} = 0$ for $0 < i \leq k$. In the first relation, $P_1(\alpha_1) = \alpha_1^t - \alpha_0 = 0$, by repeatedly substituting α_i^t in place of α_{i-1}, the following relation is obtained:

TABLE 2
n, c, α_0 Values for Building $GF(q^{3^k})$ OTFs with $q = 2^n + c$, $7 \leq n \leq 16$; $-5 \leq c \leq 5$; $-5 \leq \alpha_0 \leq 5$

n	c	α_0	n	c	α_0	n	c	α_0	n	c	α_0
7	−1	3	11	5	−4	12	3	2	14	−3	−4
7	−1	−3	11	5	−5	12	3	−2	14	−3	−5
10	−3	5	12	−3	5	12	3	−4	16	3	4
10	−3	−5	12	−3	4	12	3	−5	16	3	3
11	5	5	12	−3	2	14	−3	5	16	3	2
11	5	4	12	−3	−2	14	−3	4	16	3	−2
11	5	3	12	−3	−4	14	−3	3	16	3	−3
11	5	2	12	−3	−5	14	−3	2	16	3	−4
11	5	−2	12	3	5	14	−3	−2			
11	5	−3	12	3	4	14	−3	−3			

$$Q(\alpha_k) = \alpha_k^{t^k} - \alpha_0 = 0.$$

Hence, we obtain the binomial $Q(x) = x^{t^k} - \alpha_0$ with $Q(\alpha_k) = 0$. This binomial has the form of an OEF binomial $x^m - w$, with $w = \alpha_0$ and $m = t^k$. We want to show that $Q(x)$ satisfies the three conditions of Theorem 1, hence being irreducible over $GF(q)$ and generating the field $GF(q^m)$. We know that $P_1(x) = x^t - \alpha_0$ is irreducible over $GF(q)$ and it satisfies the three conditions of Theorem 1. Since the prime factors of t and t^k are the same, $w = \alpha_0$, and q is identical for both binomials, the first two of the three conditions are satisfied for $Q(x) = x^{t^k} - \alpha_0$ as well. For the third condition, if $t = 0 \bmod 4$, then $t^k = 0 \bmod 4$ and, therefore, $q = 1 \bmod 4$ for both cases. On the other hand, if $t \neq 0 \bmod 4$, then there are three cases we need to consider:

- If $t = 1 \bmod 4$, then $t^k = 1 \bmod 4$, hence condition 3 disappears for $Q(x)$.
- If $t = 2 \bmod 4$, then $t^k = 0 \bmod 4$, thus condition 3 applies in this case. Theorem 8 confirms that $q = 1 \bmod 4$ whenever $t = 2 \bmod 4$. Therefore, condition 3 is satisfied.
- If $t = -1 \bmod 4$, then either $t^k = 1 \bmod 4$ or $t^k = -1 \bmod 4$, hence, condition 3 disappears.

We have proven that $Q(x)$ satisfies the three conditions of Theorem 1, hence, it is irreducible over $GF(q)$ and constructs an OEF. □

We have shown that the OTF construction leads to an associated OEF representation. Likewise, it is possible to construct an OTF representation from a given OEF representation as introduced in the following theorem.

Theorem 10. *For an OEF representation of a finite field $GF(q^m)$ with irreducible polynomial $Q(x) = x^m - w$, if $m = t^k$ and none of the prime factors of t divide $\frac{q^t-1}{t(q-1)}$, then there exists an OTF representation such that $P_1(x) = x^t - \alpha_0$ and $\alpha_0 = w$.*

Proof. We need to show that $P_1(x) = x^t - w$ is irreducible. As stated, $Q(x) = x^m - w = x^{t^k} - w$ is irreducible. Therefore, it satisfies the three conditions of Theorem 1. Since the prime factors of t are identical to the prime factors of t^k and w is the same for $Q(x)$ and $P_1(x)$, conditions 1 and 2 are both satisfied for $P_1(x)$. For condition 3, there are two cases, either $t = 0 \bmod 4$ or $t \neq 0 \bmod 4$. In the first case, $t^k = 0 \bmod 4$ and, therefore, $q = 1 \bmod 4$ for both cases. Alternatively, if $t \neq 0 \bmod 4$, then the condition

disappears. We have shown that $P_1(x) = x^t - w$ is irreducible, hence, all of the conditions of Theorem 7 are satisfied. □

In order to construct explicit conversion rules between the OTF and OEF representations, we briefly introduce some notation. The OTF representation of an element $A \in GF(q^{t^k})$ is given as follows:

$$A = a_0 + a_1\alpha_k + a_2\alpha_k^2 + \ldots + a_{t-1}\alpha_k^{t-1},$$

where $a_i \in GF(q^{t^{k-1}})$ for $0 \leq i < t$. Similarly, a_i are represented as polynomials over $GF(q^{t^{k-2}})$ in the subfield:

$$a_i = \sum_{j=0}^{t-1} a_{ij}\alpha_{k-1}^j, \quad \text{for } 0 \leq i < t. \tag{7}$$

This process continues for k levels until the ground field of the tower, i.e., $GF(q)$, is reached. Note that, in this notation, in each level, a new value from the range $[0, t-1]$ is appended to coefficient indices. Hence, a coefficient in the ground field $GF(q)$ will have a k-digit t-ary number as index.

To obtain the OEF standard basis representation of A, each α_{i-1} is repeatedly replaced by α_i^t for $1 < i \leq k$ until a univariate polynomial in α_k with coefficients in the ground field $GF(q)$ is obtained. This polynomial is now in OEF standard basis representation over $GF(q)$ with $Q(\alpha_k) = \alpha_k^m - w$, where $m = t^k$ and $w = \alpha_0$, as the modulus polynomial. The relation between the OTF representation and the OEF representation defines the conversion. The following theorem constructs an explicit rule for conversion.

Theorem 11. *For a given OTF $GF(q^{t^k})$ and OEF $GF(q^m)$ association, the conversion from one representation to the other is a simple permutation of the coefficients. The permutation maps a coefficient a_ℓ of an element A in $GF(q^m)$ (or $GF(q^{t^k})$) to the corresponding coefficient in the other representation $GF(q^{t^k})$ (or $GF(q^m)$) whose index is determined by the t-ary value of the mirror image of ℓ.*

Proof. We make the following observation in (7). The index ℓ of an element a_ℓ determines its position and the power of α_i it multiplies with in the $(k+1-i)$th level of the OTF representation for $1 \leq i \leq k$. For instance, in the $(k+1-i)$th level, the ith digit ℓ_{i-1} of ℓ is appended to the index ℓ. ℓ_{i-1} gives the power of α_i that the coefficient a_ℓ multiplies with in this level. Collecting the α_i in k levels, we obtain the following multiplier for the coefficient a_ℓ in the ground field $GF(q)$:

$$\prod_{i=0}^{k-1} \alpha_{i+1}^{l_i} = \prod_{i=0}^{k-1} \alpha_k^{t^{k-1-i}l_i}.$$

The RHS follows from $\alpha_i = \alpha_k^{t^{k-i}}$. Considering the exponent $t^{k-1-i}l_i$, we notice that the list l is effectively reversed. Therefore, a coefficient a_l is mapped to the location specified by the mirror image of l. Since the indexes of the coefficients are not repeated, neither will the index of the converted coefficient repeat and the conversion will always be a permutation. □

We illustrate the conversion technique by the following example.

Example 3. Let $GF(q^{2^3})$ denote an OTF. The binomials used in the construction are given as $P_1(x) = x^2 - \alpha_0$, $P_2(x) = x^2 - \alpha_1$, and $P_3(x) = x^2 - \alpha_2$, defined over $GF(q)$, $GF(q^2)$, and $GF(q^{2^2})$, respectively, and $P_1(\alpha_1) = 0$, $P_2(\alpha_2) = 0$, and $P_3(\alpha_3) = 0$. An element $A \in GF(q^{2^3})$ has the following OTF representation:

$$\begin{aligned} A = &((a_{000} + a_{001}\alpha_1) + (a_{010} + a_{011}\alpha_1)\alpha_2) \\ &+ ((a_{100} + a_{101}\alpha_1) + (a_{110} + a_{111}\alpha_1)\alpha_2)\alpha_3. \end{aligned}$$

According to Theorem 11, we obtain the indices of the permuted coefficients by taking the mirror image of their indices in the OTF representation. For instance, a_{011} will be mapped to the coefficient whose index is the binary value of the mirror image of its index, $(110)_2 = 6$, and, thus, a_{011} will become the coefficient of α_3^6. We obtain the OEF standard basis representation as

$$\begin{aligned} A = &a_{000} + a_{100}\alpha_3 + a_{010}\alpha_3^2 + a_{110}\alpha_3^3 \\ &+ a_{001}\alpha_3^4 + a_{101}\alpha_3^5 + a_{011}\alpha_3^6 + a_{111}\alpha_3^7 \end{aligned}$$

with field polynomial $Q(x) = x^8 - \alpha_0$, and $Q(\alpha_3) = 0$. The result is easily verified by converting the OTF representation directly by replacing α_2 with α_3^2 and α_1 with α_3^4.

5 Complexity Analysis

In this section, we derive the complexities for arithmetic operations in OTF representations for second and third degree extensions. We introduce the following notation for the complexities:

- $\mathcal{A}_k$: complexity of addition operation in $GF(q^{t^k})$,
- $\mathcal{S}_k$: complexity of squaring operation in $GF(q^{t^k})$,
- $\mathcal{M}_k$: complexity of multiplication operation in $GF(q^{t^k})$,
- $\mathcal{C}_k$: complexity of multiplication of a $GF(q)$ element with an element of $GF(q^{t^k})$,
- $\mathcal{I}_k$: complexity of inversion operation in $GF(q^{t^k})$.

Before we derive the computational complexities for OTF arithmetic operations, we make the following observation: Let $A \in GF(q^{t^i})$, where $A = a_0 + a_1\alpha_i + a_2\alpha_i^2 + a_3\alpha_i^3 + \cdots + a_{t-1}\alpha_i^{t-1}$ and $a_0, a_1, \ldots, a_{t-1} \in GF(q^{t^{i-1}})$. Consider the product

$$\begin{aligned} A\alpha_i = \left(\sum_{j=0}^{t-1} a_j\alpha_i^j\right)\alpha_i &= \sum_{j=0}^{t-1} a_j\alpha_i^{j+1} \\ &= a_{t-1}\alpha_{i-1} + \sum_{j=1}^{t-1} a_{j-1}\alpha_i^j. \end{aligned} \tag{8}$$

The effect of multiplying an element $A \in GF(q^{t^i})$ with α_i is to rotate the coefficients $a_0, a_1, \ldots, a_{t-1}$ of A to the right and scale the first coefficient after rotation with α_{i-1}. However, note that multiplication of the first coefficient after rotation with α_{i-1}, i.e., $a_{t-1}\alpha_{i-1}$, will be similarly transformed in the subfield. Hence, the transformation progresses until $GF(q)$

is reached, where the modulus polynomial is $\alpha_1^t - \alpha_0$ and scaling a coefficient in $GF(q)$ with $\alpha_0 \in GF(q)$ means a constant multiplication in $GF(q)$. The complexity of this operation is denoted by $\mathcal{C}_0$.

In our treatment below, we *ignore* the cost of reading and writing data associated with the permutation and rotation operations. Note that, in a software implementation, the rotation or permutation operations can be absorbed into the overall computation since these operations are fixed permutations, i.e., data independent permutations, in our algorithm. For instance, in the operation following the permutation, the accesses to the coefficients of the operand can be modified to absorb the permutation operation. The updated access pattern may be hardcoded since the permutation is fixed and predetermined. Thereby, any cost associated with the permutation (or rotation) may be easily eliminated. Hence, no additional memory I/O or swapping of registers is involved and any such permutations come completely for free.

In a VLSI or hardware implementation, a similar approach might be used by hardcoding the permutation operations into hardware with proper wirings. The permutation operations may be easily realized by rewiring. This does not require any gates and causes minimal delay. With careful implementation, any overhead due to permutations may be avoided. However, for longer operands (i.e., larger fields), the rewiring operation may become more difficult due to possible wire crossings and the need for additional components. This may negatively affect the cost/performance and render the analysis in the paper imprecise. For hardware implementations, the complexity analysis here is intended more to serve as a means of comparison with other architectures and, therefore, should not be used as an exact guidance for cost and performance of an actual implementation.

5.1 Cost of $GF(q^{2^k})$ Operations in OTF Representation

We assume that the tower field $GF(q^{2^k})$ is constructed using a series of irreducible binomials of the form $P_i(x) = x^2 - \alpha_{i-1}$ over $GF(q^{2^{i-1}})$ for $0 < i \leq k$, where $\alpha_i \in GF(q^{2^i})$ is a root of $P_i(x)$.

Addition: For $A, B \in GF(q^{2^i})$, the addition operation

$$A + B = (a_0 + a_1\alpha_i) + (b_0 + b_1\alpha_i) = (a_0 + b_0) + (a_1 + b_1)\alpha_i$$

requires two $GF(q^{2^{i-1}})$ additions, hence, $\mathcal{A}_i = 2\mathcal{A}_{i-1}$ and the complexity of $GF(q^{2^k})$ addition in nonrecursive form is

$$\mathcal{A}_k = 2^k\mathcal{A}_0.$$

Multiplication: The multiplication operation

$$AB = (a_0 + a_1\alpha_i)(b_0 + b_1\alpha_i)$$

becomes

$$\begin{aligned} AB &= a_0b_0 + (a_0b_1 + a_1b_0)\alpha_i + a_1b_1\alpha_i^2 \\ &= (a_0b_0 + a_1b_1\alpha_{i-1}) + (a_0b_1 + a_1b_0)\alpha_i. \end{aligned}$$

The computation requires four multiplications and two additions in the subfield $GF(q^{2^{i-1}})$ and the multiplication of $a_1b_1 \in GF(q^{2^{i-1}})$ by α_{i-1} which has complexity $\mathcal{C}_0$ (8). Thus,

$$\mathcal{M}_i = 4\mathcal{M}_{i-1} + 2\mathcal{A}_{i-1} + \mathcal{C}_0.$$

The nonrecursive form for the complexity of $GF(q^{2^k})$ multiplication is obtained as

$$\begin{aligned} \mathcal{M}_k &= 4^k\mathcal{M}_0 + \sum_{j=1}^{k} 4^{k-j}(2^j\mathcal{A}_0 + \mathcal{C}_0) \\ &= 4^k\mathcal{M}_0 + (4^k - 2^k)\mathcal{A}_0 + \frac{1}{3}(4^k - 1)\mathcal{C}_0. \end{aligned} \tag{9}$$

The complexity may be improved by using the Karatsuba-Ofman algorithm [16]. To achieve the four products, only three multiplications will be needed:

$$\begin{aligned} AB = &(a_0b_0 + a_1b_1\alpha_{i-1}) \\ &+ ((a_0 + a_1)(b_0 + b_1) - a_0b_0 - a_1b_1)\alpha_i. \end{aligned} \tag{10}$$

With the application of the Karatsuba method, the complexity changes as follows:

$$\mathcal{M}_i^{KOA} = 3\mathcal{M}_{i-1}^{KOA} + 5\mathcal{A}_{i-1} + \mathcal{C}_0.$$

The complexity of $GF(q^{2^k})$ multiplication with the Karatsuba technique is found as

$$\begin{aligned} \mathcal{M}_k^{KOA} &= 3^k\mathcal{M}_0 + \sum_{j=1}^{k} 3^{k-j}(2^{j-1}5\mathcal{A}_0 + \mathcal{C}_0) \\ &= 3^k\mathcal{M}_0 + 5(3^k - 2^k)\mathcal{A}_0 + \frac{1}{2}(3^k - 1)\mathcal{C}_0. \end{aligned}$$

Squaring: The squaring operation $A^2 = (a_0 + a_1\alpha_i)^2$ becomes

$$\begin{aligned} A^2 &= a_0^2 + 2a_0a_1\alpha_i + a_1^2\alpha_i^2 \\ &= (a_0^2 + a_1^2\alpha_{i-1}) + 2a_0a_1\alpha_i, \end{aligned}$$

which may be achieved by two squarings, one multiplication, two additions in the subfield $GF(q^{2^{i-1}})$, and one constant multiplication in the ground field $GF(q)$ as explained in (8):

$$\mathcal{S}_i = 2\mathcal{S}_{i-1} + \mathcal{M}_{i-1} + 2\mathcal{A}_{i-1} + \mathcal{C}_0.$$

Since the multiplication complexity $\mathcal{M}_{i-1}$ depends on the use of the Karatsuba algorithm, we obtain two nonrecursive complexity equations for the cost of squaring in $GF(q^{2^k})$:

$$\begin{aligned} \mathcal{S}_k &= 2^k\mathcal{S}_0 + \sum_{j=1}^{k} 2^{k-j}(\mathcal{M}_{j-1} + \mathcal{C}_0 + 2^j\mathcal{A}_0) \\ &= 2^k\mathcal{S}_0 + \frac{1}{2}(4^k - 2^k)\mathcal{M}_0 + \frac{1}{2}(k2^k + 4^k - 2^k)\mathcal{A}_0 \\ &\quad + \frac{1}{6}(4^k + 3 \cdot 2^k - 4)\mathcal{C}_0, \end{aligned} \tag{11}$$

$$\begin{aligned} \mathcal{S}_k^{KOA} &= 2^k\mathcal{S}_0 + \sum_{j=1}^{k} 2^{k-j}(\mathcal{M}_{j-1} + \mathcal{C}_0 + 2^j\mathcal{A}_0) \\ &= 2^k\mathcal{S}_0 + (3^k - 2^k)\mathcal{M}_0 + \frac{1}{2}(3^k - 1)\mathcal{C}_0 \\ &\quad + \frac{1}{2}(10 \cdot 3^k - 3k2^k - 10 \cdot 2^k)\mathcal{A}_0. \end{aligned}$$

Inversion: The inverse of an element in $GF(q^{2^i})$ may be computed by the application of the Direct Inversion technique as shown in (2):

$$b_0 = a_0(a_0^2 - \alpha_{i-1}a_1^2)^{-1} \quad \text{and} \quad b_1 = -a_1(a_0^2 - \alpha_{i-1}a_1^2)^{-1}.$$

The computation requires two squarings, one addition, one inversion, and two multiplications in $GF(q^{2^{i-1}})$, and one constant multiplication in the ground field $GF(q)$ that comes from the multiplication of α_{i-1} with a_1^2, as explained in (8). This can be expressed as follows:

$$\mathcal{I}_i = \mathcal{I}_{i-1} + 2\mathcal{S}_{i-1} + 2\mathcal{M}_{i-1} + \mathcal{A}_{i-1} + \mathcal{C}_0.$$

The aggregate cost of inversion in $GF(q^{2^k})$ is found as the summation

$$\mathcal{I}_k = \mathcal{I}_0 + \sum_{j=0}^{k-1}(2\mathcal{S}_j + 2\mathcal{M}_j + \mathcal{A}_j + \mathcal{C}_0),$$

which is simplified as follows:

$$\begin{aligned}\mathcal{I}_k = \mathcal{I}_0 &+ (4^k - 2^k)\mathcal{M}_0 + 2(2^k - 1)\mathcal{S}_0 \\ &+ (k2^k + 4^k - 4\cdot 2^k + 3)\mathcal{A}_0 \\ &+ \frac{1}{3}(4^k + 3\cdot 2^k - 3k - 4)\mathcal{C}_0.\end{aligned} \tag{12}$$

It is possible to slightly improve the number of constant multiplications by applying OEF multiplication and squaring to OTF inversion. The conversion is a mere permutation and, therefore, comes for free. The improved complexity is found as

$$\begin{aligned}\mathcal{I}_k = \mathcal{I}_0 &+ (4^k + 2^k - 2)\mathcal{M}_0 + (4^k + 2^{k+1} - 6k - 3)\mathcal{A}_0 \\ &+ (2^{k+2} - 3k - 4)\mathcal{C}_0.\end{aligned} \tag{13}$$

Instead, if the Karatsuba Algorithm introduced in (10) is used for all multiplications, the complexity is further reduced to

$$\begin{aligned}\mathcal{I}_k^{KOA} = \mathcal{I}_0 &+ 2(3^k - 2^k)\mathcal{M}_0 + 2(2^k - 1)\mathcal{S}_0 \\ &+ (10\cdot 3^k - 3k2^k - 13\cdot 2^k + 3)\mathcal{A}_0 \\ &+ (3^k - k - 1)\mathcal{C}_0.\end{aligned} \tag{14}$$

5.2 Cost of $GF(q^{3^k})$ Operations in OTF Representation

In the OTF representation, the tower field $GF(q^{3^k})$ is constructed using a series of irreducible binomials of the form $P_i(x) = x^3 - \alpha_{i-1}$ over $GF(q^{3^{i-1}})$ for $0 < i \le k$, where $\alpha_i \in GF(q^{3^i})$ is a root of $P_i(x)$.

Addition: Similarly to the $GF(q^{2^k})$ case, the addition operation in $GF(q^{3^i})$ requires three $GF(q^{3^{i-1}})$ additions, hence, $\mathcal{A}_i = 3\mathcal{A}_{i-1}$ and, in nonrecursive form, the complexity of addition in $GF(q^{3^k})$ is

$$\mathcal{A}_k = 3^k\mathcal{A}_0.$$

Multiplication: The multiplication

$$AB = (a_0 + a_1\alpha_i + a_2\alpha_i^2)(b_0 + b_1\alpha_i + b_2\alpha_i^2)$$

may be computed as

$$\begin{aligned}AB &= a_0b_0 + (a_0b_1 + a_1b_0)\alpha_i + (a_2b_0 + a_1b_1 + a_0b_2)\alpha_i^2 \\ &\quad + (a_2b_1 + a_1b_2)\alpha_i^3 + a_2b_2\alpha_i^4 \\ &= a_0b_0 + (a_2b_1 + a_1b_2)\alpha_{i-1} \\ &\quad + (a_0b_1 + a_1b_0 + a_2b_2\alpha_{i-1})\alpha_i \\ &\quad + (a_2b_0 + a_1b_1 + a_0b_2)\alpha_i^2.\end{aligned}$$

Hence, the complexity of $GF(q^{3^i})$ multiplication is

$$\mathcal{M}_i = 9\mathcal{M}_{i-1} + 6\mathcal{A}_{i-1} + 2\mathcal{C}_0.$$

The nonrecursive equation for the complexity of multiplication in $GF(q^{3^k})$ is obtained as

$$\begin{aligned}\mathcal{M}_k &= 9^k\mathcal{M}_0 + \sum_{j=1}^{k} 9^{k-j}(6\cdot 3^{j-1}\mathcal{A}_0 + 2\mathcal{C}_0) \\ &= 9^k\mathcal{M}_0 + (9^k - 3^k)\mathcal{A}_0 + \frac{1}{4}(9^k - 1)\mathcal{C}_0.\end{aligned} \tag{15}$$

Using the ternary version of the Karatsuba method, the complexity might be improved:

$$\begin{aligned}AB = D_0 &+ (D_5 - D_1 - D_2)\alpha_{i-1} \\ &+ (D_3 - D_1 - D_0 + D_2\alpha_{i-1})\alpha_i \\ &+ (D_4 - D_2 - D_0 + D_1)\alpha_i^2,\end{aligned}$$

where

$$\begin{aligned}D_0 &= a_0b_0 \\ D_1 &= a_1b_1 \\ D_2 &= a_2b_2 \\ D_3 &= (a_0 + a_1)(b_0 + b_1) \\ D_4 &= (a_0 + a_2)(b_0 + b_2) \\ D_5 &= (a_1 + a_2)(b_1 + b_2).\end{aligned} \tag{16}$$

This reduces the number of multiplications in exchange of extra additions. The complexity of $GF(q^{3^i})$ multiplication becomes

$$\mathcal{M}_i^{KOA} = 6\mathcal{M}_{i-1} + 15\mathcal{A}_{i-1} + 2\mathcal{C}_0.$$

In nonrecursive form, the complexity in $GF(q^{3^k})$ is obtained as

$$\begin{aligned}\mathcal{M}_k^{KOA} &= 6^k\mathcal{M}_0 + \sum_{j=1}^{k} 6^{k-j}(3^{j-1}15\mathcal{A}_0 + 2\mathcal{C}_0) \\ &= 6^k\mathcal{M}_0 + \frac{2}{5}(6^k - 1)\mathcal{C}_0 + 5(6^k - 3^k)\mathcal{A}_0.\end{aligned}$$

Squaring: The squaring operation of $A \in GF(q^{3^i})$ is achieved as follows:

$$\begin{aligned}A^2 &= a_0^2 + 2a_0a_1\alpha_i + (2a_2a_0 + a_1^2)\alpha_i^2 + 2a_2a_1\alpha_i^3 + a_2^2\alpha_i^4 \\ &= a_0^2 + 2a_2a_1\alpha_{i-1} + (2a_0a_1 + a_2^2\alpha_{i-1})\alpha_i + (2a_2a_0 + a_1^2)\alpha_i^2.\end{aligned}$$

This may be realized with complexity

$$\mathcal{S}_i = 3\mathcal{S}_{i-1} + 3\mathcal{M}_{i-1} + 6\mathcal{A}_{i-1} + 2\mathcal{C}_0.$$

The nonrecursive complexities for $GF(q^{3^k})$ are obtained as

TABLE 3
Comparison of OEF and OTF Complexities ($\delta = \lfloor \log_2(m-1) \rfloor + HW(m-1)$)

Operation	$\mathcal{M}_0$	$\mathcal{A}_0$
$\mathcal{M}(m)$ (OEF)	m^2	m^2-1
$\mathcal{M}(m)$ (OTF2)	m^2	$\frac{1}{3}(4m^2-3m-1)$
$\mathcal{M}(m)$ (OTF3)	m^2	$\frac{1}{4}(5m^2-4m-1)$
$\mathcal{S}(m)$ (OEF)	$\frac{1}{2}(m^2+m)$	$\frac{1}{2}(m^2+5m-8)$, m even $\frac{1}{2}(m^2+3m-2)$, m odd
$\mathcal{S}(m)$ (OTF2)	$\frac{1}{2}(m^2+m)$	$\frac{1}{6}(4m^2+3m\log_2 m-4)$
$\mathcal{S}(m)$ (OTF3)	$\frac{1}{2}(m^2+m)$	$\frac{1}{8}(5m^2+8m\log_3 m-5)$
$\mathcal{I}(m)$ (OEF)	$(\delta-1)m^2+\delta(m-1)+2m$	$(\delta-1)(m^2-1)+m-1$
$\mathcal{I}(m)$ (OTF2)	m^2+m-2	$\frac{1}{3}(4m^2+3m\log_2 m-9m+5)$
$\mathcal{I}(m)$ (OTF3)	$\frac{1}{16}(21m^2+12m-33)$	$\frac{1}{64}(105m^2-176m-8\log_3 m+96m\log_3 m+71)$

$$\begin{aligned}\mathcal{S}_k &= 3^k\mathcal{S}_0+\sum_{j=1}^{k}3^{k-j}(3\mathcal{M}_{j-1}+6\cdot 3^{j-1}\mathcal{A}_0+2\mathcal{C}_0)\\ &= 3^k\mathcal{S}_0+\frac{1}{2}(9^k-3^k)\mathcal{M}_0+\frac{1}{8}(9^k+4\cdot 3^k-5)\mathcal{C}_0\\ &\quad+\frac{1}{2}(9^k-3^k+2k3^k)\mathcal{A}_0,\end{aligned} \tag{17}$$

$$\begin{aligned}\mathcal{S}_k^{KOA} &= 3^k\mathcal{S}_0+\sum_{j=1}^{k}3^{k-j}(3\mathcal{M}_{j-1}+6\cdot 3^{j-1}\mathcal{A}_0+2\mathcal{C}_0)\\ &= 3^k\mathcal{S}_0+(6^k-3^k)\mathcal{M}_0+\frac{2}{5}(6^k-1)\mathcal{C}_0\\ &\quad+(5\cdot 6^k-3k3^k-5\cdot 3^k)\mathcal{A}_0.\end{aligned}$$

Inversion: The inverse of an element $A \in GF(q^{3^i})$ may be computed by the application of the Direct Inversion technique, as shown in (3), (4), (5):

$$\begin{aligned}b_0 &= \Delta^{-1}(a_0^2-a_1a_2\alpha_{i-1}),\\ b_1 &= \Delta^{-1}(a_2^2\alpha_{i-1}-a_0a_1),\\ b_2 &= \Delta^{-1}(a_1^2-a_0a_2),\end{aligned}$$

where

$$\Delta = a_0^3+((a_1^2-3a_0a_2)a_1+a_2^3\alpha_{i-1})\alpha_{i-1}.$$

The computation requires three squarings, nine multiplications, eight additions, and one inversion in $GF(q^{3^{i-1}})$, and four constant multiplications in the ground field $GF(q)$ due to the property mentioned in (8). It is also assumed that multiplication by 3 is realized with two additions. The resulting complexity is expressed as follows:

$$\mathcal{I}_i = \mathcal{I}_{i-1}+3\mathcal{S}_{i-1}+9\mathcal{M}_{i-1}+8\mathcal{A}_{i-1}+4\mathcal{C}_0.$$

The aggregate cost of inversion in $GF(q^{3^k})$ is expressed as the summation

$$\mathcal{I}_k = \mathcal{I}_0+\sum_{i=0}^{k-1}3\mathcal{S}_{i-1}+9\mathcal{M}_{i-1}+8\mathcal{A}_{i-1}+4\mathcal{C}_0,$$

which is simplified as follows:

$$\begin{aligned}\mathcal{I}_k &= \mathcal{I}_0+\frac{3}{2}(3^k-1)\mathcal{S}_0+\frac{1}{16}(21\cdot 9^k-12\cdot 3^k-9)\mathcal{M}_0\\ &\quad+\frac{1}{64}(21\cdot 9^k+48\cdot 3^k-8k-69)\mathcal{C}_0\\ &\quad+\frac{1}{16}(21\cdot 9^k+24\cdot 3^k k-56\cdot 3^k+35)\mathcal{A}_0.\end{aligned} \tag{18}$$

The complexity $\mathcal{I}_k$ in $GF(q^{3^k})$ when OEF multiplication and squaring are utilized for all multiplications and squarings is found as

$$\begin{aligned}\mathcal{I}_k &= \mathcal{I}_0+\frac{3}{16}(7\cdot 9^k+4\cdot 3^k-11)\mathcal{M}_0\\ &\quad+2(3^{k+1}-4k-3)\mathcal{C}_0\\ &\quad+\frac{1}{16}(21\cdot 9^k+4\cdot 3^k-25)\mathcal{A}_0.\end{aligned} \tag{19}$$

On the other hand if the Karatsuba Algorithm introduced in (16) is used for all multiplications, the complexity is further reduced to

$$\begin{aligned}\mathcal{I}_k^{KOA} &= \mathcal{I}_0+\frac{3}{2}(3^k-1)\mathcal{S}_0+\frac{1}{10}(24\cdot 6^k-15\cdot 3^k-9)\mathcal{M}_0\\ &\quad+\frac{1}{25}(24\cdot 6^k-20k-24)\mathcal{C}_0\\ &\quad+\frac{1}{4}(48\cdot 6^k-18\cdot 3^k k-77\cdot 3^k+29)\mathcal{A}_0.\end{aligned} \tag{20}$$

6 Comparison of OTF and OEF Complexities

We base our treatment of OEFs on the detailed complexity analysis given in [11]. In the derivation of the complexities, we assume small w values for OEFs and small α_0 values for OTFs, e.g., w and α_0 are usually selected as small numbers such as 2 or 3. Since multiplication of a ground field $GF(q)$ element by a small integer can be performed by simple addition(s) and/or shift(s), the complexity of this operation can be approximated as the complexity of addition and, therefore, we assume $\mathcal{C}_0 \cong \mathcal{A}_0$. We also assume $\mathcal{S}_0 = \mathcal{M}_0$ for simplification. Table 3 summarizes the complexities compiled from [11] and from (9), (11), (12), (15), (17), (18). The complexities are derived in terms of m, the extension degree for OEFs ($GF(q^m)$), and $m = t^k$ for OTFs ($GF(q^{t^k})$). To differentiate OTFs of form $GF(q^{2^k})$ and $GF(q^{3^k})$, OTF2 and OTF3 are used, respectively.

Table 3 shows that the number of ground field $GF(q)$ multiplications used in performing OTF multiplications and squarings are identical to those of OEF operations. OEF multiplication and squaring appear to be slightly more efficient than OTF multiplication and squaring due to a reduced number of additions. However, it was shown earlier that OTFs can be converted to an OEF representation and back via simple permutation. Therefore, we shall assume that the OEF multiplication and squaring algorithms are used to

TABLE 4
Number of $GF(q)$ Operations for OTF Arithmetic

m	$\mathcal{M}(m)$	$\mathcal{S}(m)$	$\mathcal{I}(m)$
8	$64\mathcal{M}_0 + 77\mathcal{A}_0$	$36\mathcal{M}_0 + 54\mathcal{A}_0$	$\mathcal{I}_0 + 70\mathcal{M}_0 + 84\mathcal{A}_0$
9	$81\mathcal{M}_0 + 92\mathcal{A}_0$	$45\mathcal{M}_0 + 68\mathcal{A}_0$	$\mathcal{I}_0 + 111\mathcal{M}_0 + 136\mathcal{A}_0$
16	$256\mathcal{M}_0 + 325\mathcal{A}_0$	$136\mathcal{M}_0 + 202\mathcal{A}_0$	$\mathcal{I}_0 + 270\mathcal{M}_0 + 355\mathcal{A}_0$
27	$729\mathcal{M}_0 + 884\mathcal{A}_0$	$378\mathcal{M}_0 + 536\mathcal{A}_0$	$\mathcal{I}_0 + 975\mathcal{M}_0 + 1244\mathcal{A}_0$
32	$1024\mathcal{M}_0 + 1333\mathcal{A}_0$	$528\mathcal{M}_0 + 762\mathcal{A}_0$	$\mathcal{I}_0 + 1054\mathcal{M}_0 + 1426\mathcal{A}_0$

achieve OTF multiplications and squarings with no overhead in the conversion. In Table 4 and Table 9 (see the Appendix) the number of operations required for multiplication and squaring are tabulated for practical values of m. These tables show that performing the two operations in OTFs and OEFs can be considered to be of equal complexity for practical considerations.

Note that the single inversion in the ground field $GF(q)$ required for $\mathcal{I}(m)$ is not shown in the table. This inversion can be performed with table lookup and its cost $\mathcal{I}_0$ is negligible for large m. Also, omitting $\mathcal{I}_0$ does not make much difference in terms of comparing the relative performances of the two inversion algorithms since both the OTF and OEF inversion algorithms require a ground field $GF(q)$ inversion. For inversion, the new algorithm presents a significant improvement over the OEF inversion algorithm. In Table 3, the number of multiplications required for OEF inversion is given as $(\delta-1)m^2 + \delta(m-1) + 2m$. The value of δ depends on the bit length and the Hamming weight of $m-1$. In the best case for m, the Hamming weight may be as low as 1, leading to a $\lfloor \log_2(m-1) \rfloor m^2 + (\lfloor \log_2(m-1) \rfloor + 1)(m-1) + 2m$ complexity. The number of additions grows similarly with δ. In practical terms, this means that the Itoh-Tsujii inversion technique will cost at least $\log_2(m-1)$ field multiplications when applied to OEFs. In elliptic curve implementations based on OEFs, typically an inversion/multiplication cost ratio of larger than 4 is observed. On the other hand, as seen in Table 3, the inversion complexity for OTFs grows linearly with m^2, both in the number of multiplications and additions. For both $GF(q^{2^k})$ and $GF(q^{3^k})$ OTFs, the inversion/multiplication cost ratio is slightly larger than one. The asymptotic complexity of the OTF inversion algorithm makes it even more desirable for larger values of m. For instance, for $m = 32$, the complexity of inversion in OEFs is found as $\mathcal{I}_0 + 8535\mathcal{M}_0 + 8215\mathcal{A}_0$, whereas, for OTFs, it is only $\mathcal{I}_0 + 1054\mathcal{M}_0 + 1426\mathcal{A}_0$.

TABLE 5
Number of $GF(q)$ Operations for OTF Arithmetic with OEF Multiplication and Squaring

m	$\mathcal{M}(m)$	$\mathcal{S}(m)$	$\mathcal{I}(m)$
8	$64\mathcal{M}_0 + 63\mathcal{A}_0$	$36\mathcal{M}_0 + 48\mathcal{A}_0$	$\mathcal{I}_0 + 70\mathcal{M}_0 + 80\mathcal{A}_0$
9	$81\mathcal{M}_0 + 80\mathcal{A}_0$	$45\mathcal{M}_0 + 53\mathcal{A}_0$	$\mathcal{I}_0 + 111\mathcal{M}_0 + 136\mathcal{A}_0$
16	$256\mathcal{M}_0 + 255\mathcal{A}_0$	$136\mathcal{M}_0 + 164\mathcal{A}_0$	$\mathcal{I}_0 + 270\mathcal{M}_0 + 311\mathcal{A}_0$
27	$729\mathcal{M}_0 + 728\mathcal{A}_0$	$378\mathcal{M}_0 + 404\mathcal{A}_0$	$\mathcal{I}_0 + 975\mathcal{M}_0 + 1091\mathcal{A}_0$
32	$1024\mathcal{M}_0 + 1023\mathcal{A}_0$	$528\mathcal{M}_0 + 588\mathcal{A}_0$	$\mathcal{I}_0 + 1054\mathcal{M}_0 + 1166\mathcal{A}_0$

TABLE 6
Number of $GF(q)$ Operations for OTF Arithmetic with Karatsuba

m	$\mathcal{M}(m)$	$\mathcal{S}(m)$	$\mathcal{I}(m)$
8	$27\mathcal{M}_0 + 108\mathcal{A}_0$	$27\mathcal{M}_0 + 72\mathcal{A}_0$	$\mathcal{I}_0 + 52\mathcal{M}_0 + 120\mathcal{A}_0$
9	$36\mathcal{M}_0 + 149\mathcal{A}_0$	$36\mathcal{M}_0 + 95\mathcal{A}_0$	$\mathcal{I}_0 + 84\mathcal{M}_0 + 217\mathcal{A}_0$
16	$81\mathcal{M}_0 + 365\mathcal{A}_0$	$81\mathcal{M}_0 + 269\mathcal{A}_0$	$\mathcal{I}_0 + 160\mathcal{M}_0 + 489\mathcal{A}_0$
27	$216\mathcal{M}_0 + 1031\mathcal{A}_0$	$216\mathcal{M}_0 + 788\mathcal{A}_0$	$\mathcal{I}_0 + 516\mathcal{M}_0 + 1919\mathcal{A}_0$
32	$243\mathcal{M}_0 + 1176\mathcal{A}_0$	$243\mathcal{M}_0 + 936\mathcal{A}_0$	$\mathcal{I}_0 + 484\mathcal{M}_0 + 1774\mathcal{A}_0$

The complexity of OTF inversion may be slightly improved by utilizing OEF multiplication and squaring in implementing the OTF inversion. Then, the complexity of $GF(q^{2^k})$ inversion is derived in (13) as

$$\begin{aligned}\mathcal{I}(m)^* = \mathcal{I}_0 &+ (m^2 + m - 2)\mathcal{M}_0 \\ &+ (m^2 + 2m - 6\log_2 m - 3)\mathcal{A}_0 \\ &+ (4m - 3\log_2 m - 4)\mathcal{C}_0\end{aligned} \tag{21}$$

and, for $GF(q^{3^k})$ in (19) as

$$\begin{aligned}\mathcal{I}(m)^* = \mathcal{I}_0 &+ \frac{3}{16}(7m^2 + 4m - 11)\mathcal{M}_0 \\ &+ 2(3m - 4\log_3 m - 3)\mathcal{C}_0 \\ &+ \frac{1}{16}(21m^2 + 4m - 25)\mathcal{A}_0.\end{aligned} \tag{22}$$

In Table 5, the number of ground field operations are summarized for practical values of m.

These complexities are dramatically reduced by using the Karatsuba-Ofman algorithm for multiplications, yielding the following inversion complexities in terms of the word length m. For $GF(q^{2^k})$, the complexity is derived in (14) as

$$\begin{aligned}\mathcal{I}(m)^{KOA} = &(2m^{\log_2 3} - 2)\mathcal{M}_0 \\ &+ (11m^{\log_2 3} - 3m\log_2 m - 13m - \log_2 m + 2)\mathcal{A}_0\end{aligned}$$

and for $GF(q^{3^k})$ in (20) as

$$\begin{aligned}\mathcal{I}(m)^{KOA} = &\frac{12}{5}\left(m^{\log_3 6} - 1\right)\mathcal{M}_0 + \\ &\left(\frac{324}{25}m^{\log_3 6} - \frac{9}{2}m\log_3 m - \frac{77}{4}m - \frac{4}{5}\log_3 m + \frac{629}{100}\right)\mathcal{A}_0.\end{aligned}$$

Table 6 provides the number of ground field operations for several values of m. For $m = 32$, the complexity of inversion is found as $I_0 + 484\mathcal{M}_0 + 1774\mathcal{A}_0$, with a significantly lower number of multiplications compared to the standard version of the OTF inversion technique.

In order to verify the theoretical performance benefits of OTF inversion over Itoh-Tsujii inversion for OEFs, we implemented both inversion algorithms on the ARM7TDMI platform, which is a representative of the popular platforms for embedded systems like PDAs and mobile phones. Inversion was performed for a medium-sized field $GF(4093^{16})$ and a large-sized field $GF(1021^{32})$ whose elements can be represented with 192 and 320 bits, respectively. The inversion routines were developed in plain C using the ARM Developer Suite version 1.0.1, which includes the Metrowerks Codewarrior compiler, and the performance of both inversion algorithms was measured

using the "ARMulator" emulation engine. According to our implementation results, OTF inversion performed 6.3 times faster than Itoh-Tsujii's algorithm for the 192 bit field and 8.5 times faster than Itoh-Tsujii's algorithm for the 320 bit field, which very closely verifies the theoretical complexities given in Table 3. Furthermore, we can easily conclude that these speedups in inversion would result in more than 2 to 3 times improvement in the speed of an elliptic curve cryptosystem.

7 Generalization of OTFs

The OTF definition given in Section 3 restricts the ground field to $GF(q)$, a prime field, and the extension degree to a power of an integer. For instance, for an extension degree of $12 = 3 \cdot 2^2$ or $72 = 3^2 \cdot 2^3$, Definition 1 does not allow an OTF construction. This may be too restrictive for certain applications. However, by allowing $GF(q)$ to be an extension field, this restriction may be overcome. Note that $GF(q)$ may itself be in a specialized field representation, e.g., an OTF with $GF(q) = GF(p^{t_1^{k_1}})$. In this case, we may link the roots of the irreducible binomials constructing the two OTFs by choosing α_0 of the OTF $GF((p^{t_1^{k_1}})^{t_2^{k_2}})$ to be the root of the generating binomial of the OTF $GF(p^{t_1^{k_1}})$.

For such a generalization, the theorems introduced for OTF construction, i.e., Theorems 4, 5, 6, and 7, will still apply. Furthermore, the complexity analysis presented in Section 5 and the conversion rule described in Section 4 will continue to hold over $GF(p^{t_1^{k_1}})$. Rules for conversion between the OTF $GF((p^{t_1^{k_1}})^{t_2^{k_2}})$ and the OEF $GF(p^{t_1^{k_1} t_2^{k_2}})$ representations similar to those described in Section 4 will apply. Conversion between the two field representations becomes a simple permutation of the ground field $GF(p)$ coefficients, as before. Also, the advantage shown in (8) still applies when multiplying α_0 of the OTF $GF(q^{t_2^{k_2}})$, with an element in $GF(q)$.

It is possible to attain extension degrees of the form $t_1^{k_1} \cdot t_2^{k_2} \cdots t_n^{k_n}$ by repeatedly extending OTFs and by linking their representations through the constant term α_0, as described earlier. For instance, one may construct:

- $GF(p^{t_1^{k_1}})$ by extending the prime field $GF(p)$, with $P_i(x) = x^{t_1} - \alpha_{i-1}^{(1)}$ and $P_i(\alpha_i^{(1)}) = 0$ for $1 \le i \le k_1$,
- $GF((p^{t_1^{k_1}})^{t_2^{k_2}})$ by extending $GF(p^{t_1^{k_1}})$, with $P_i(x) = x^{t_2} - \alpha_{i-1}^{(2)}$ and $P_i(\alpha_i^{(2)}) = 0$ for $1 \le i \le k_2$,
- $GF((p^{t_1^{k_1}})^{t_2^{k_2}})^{t_3^{k_3}})$ by extending $GF((p^{t_1^{k_1}})^{t_2^{k_2}})$, with $P_i(x) = x^{t_3} - \alpha_{i-1}^{(3)}$ and $P_i(\alpha_i^{(3)}) = 0$ for $1 \le i \le k_3$, etc.

The second tower field $GF((p^{t_1^{k_1}})^{t_2^{k_2}})$ is linked to the first tower field $GF(p^{t_1^{k_1}})$ by choosing $\alpha_0^{(2)} = \alpha_{k_1}^{(1)}$. Likewise, the third tower field is linked to the second by setting $\alpha_0^{(3)} = \alpha_{k_2}^{(2)}$. In general, the jth tower field is linked to the $(j-1)$st by selecting $\alpha_0^{(j)} = \alpha_{k_{j-1}}^{(j-1)}$. This procedure is continued until the desired extension degree is reached. For the construction, Theorems 1 and 7 may still be used with minimal or no change.

Table 8 (Appendix) gives a list of $p = 2^n + c$ and α_0 values costructing generalized OTFs of the form $GF((p^{3^2})^2)$, where only the first extension of the generalized OTF $GF((p^{3^2})^{2^k})$ is constructed on top of the OTF $GF(p^{3^2})$ with first irreducible binomial $P_1(x) = x^3 - \alpha_0$, for $-5 \le c \le 5$, $7 \le n \le 16$, and $-5 \le \alpha_0 \le 5$. Table 7 (Appendix) gives a list of $p = 2^n + c$ and w values constructing generalized OTFs of the form $GF((p^3)^{2^k})$, where an OTF is contructed on top of the OEF $GF(p^3)$ with irreducible binomial $P(x) = x^3 - w$, for $-5 \le c \le 5$, $7 \le n \le 16$, and $-5 \le w \le 5$. Table 9 (Appendix) gives a list of $q = 2^n + c$ and α_0 values costructing generalized OTFs of the form $GF((q^{3^2})^{2^k})$, where an OTF is contructed on top of the OTF $GF(q^{3^2})$ with first irreducible binomial $P_1(x) = x^3 - \alpha_0$, for $-5 \le c \le 5$, $7 \le n \le 16$, and $-5 \le \alpha_0 \le 5$. The complexities of arithmetic operations for some generalized OTFs of the form $GF((p^3)^{2^k})$, $GF((q^{3^2})^2)$, and $GF((q^{3^2})^{2^k})$ are listed in Tables 11, 12, and 13 (Appendix).

8 On the Security of Optimal Tower Fields

The special structure of OTFs certainly requires special attention. OTFs belong to a subclass of OEFs and, therefore, they inherit some of the security characteristics of OEFs. The main difference of OTFs is the restriction to highly composite extension degrees. There is usually a reluctance to use composite extension fields for elliptic curve cryptography due to the possibility for their weaknesses against certain types of attacks. A technique, commonly known as Weil descent, was shown to be successful in attacking elliptic curve cryptosystems over composite extension fields with characteristic 2 [17], [18]. However, it is pointed out in [17] that the same technique does not seem to apply to fields of odd characteristic and, hence, OEFs may be considered secure against Weil descent. In [19], different finite fields are investigated for use in elliptic curve cryptography. Again, the weakness of composite extension fields of characteristic 2 against Weil descent is mentioned. The same reference also explains that the application of Weil descent to curves defined over OEFs does not lead to the same nice results as obtained in the case of characteristic 2.

9 Conclusion

In this paper, we introduced a new tower field representation, Optimal Tower Fields, and outlined a construction technique which establishes the conditions for their existence. It was also shown that OTF elements can be converted to OEF representation and back with a simple permutation. Thus, OTF operations (such as the Direct Inversion technique) are accessible from the OEF representation whenever a suitable OTF exists. This also means that cryptographic applications built over OTFs inherit the security characteristics of the ones built over OEFs.

TABLE 7
n, c, w Values for Buiding $GF((p^3)^{2^k})$ Generalized OTFs with Irreducible Binomial $P(x) = x^3 - w$ for OEF $GF(p^3)$, $p = 2^n + c$, $7 \le n \le 16$; $-5 \le c \le 5$; $-5 \le w \le 5$

n	c	w	n	c	w	n	c	w	n	c	w
11	5	5	11	5	−5	12	−3	−2	14	−3	−2
11	5	2	12	−3	5	12	−3	−5			
11	5	−2	12	−3	2	14	−3	2			

TABLE 8
n, c, α_0 Values for Building $GF((p^{3^2})^2)$ Generalized OTFs with First Irreducible Binomial $P_1(x) = x^3 - \alpha_0$ for OTF $GF(p^{3^2})$, $p = 2^n + c$, $7 \le n \le 16$; $-5 \le c \le 5$; $-5 \le \alpha_0 \le 5$

n	c	α_0	n	c	α_0	n	c	α_0	n	c	α_0
7	−1	3	12	−3	5	12	3	−4	16	3	2
11	5	5	12	−3	2	12	3	−5	16	3	−4
11	5	2	12	−3	−2	14	−3	2			
11	5	−2	12	−3	−5	14	−3	−2			
11	5	−5	12	3	2	16	3	3			

TABLE 9
n, c, α_0 Values for Bulding $GF((q^{3^2})^{2^k})$ Generalized OTFs with First Irreducible Binomial $P_1(x) = x^3 - \alpha_0$ for Constructing OTF $GF(q^{3^2})$, $q = 2^n + c$, $7 \le n \le 16$; $-5 \le c \le 5$; $-5 \le \alpha_0 \le 5$

n	c	α_0	n	c	α_0	n	c	α_0	n	c	α_0
11	5	−5	11	5	5	12	−3	2	14	−3	2
11	5	−2	12	−3	−5	12	−3	5			
11	5	2	12	−3	−2	14	−3	−2			

TABLE 10
Number of $GF(p)$ Operations for OEF Arithmetic

m	$\mathcal{M}(m)$	$\mathcal{S}(m)$	$\mathcal{I}(m)$
6	$36\mathcal{M}_0 + 35\mathcal{A}_0$	$21\mathcal{M}_0 + 29\mathcal{A}_0$	$\mathcal{I}_0 + 140\mathcal{M}_0 + 110\mathcal{A}_0$
8	$64\mathcal{M}_0 + 63\mathcal{A}_0$	$36\mathcal{M}_0 + 48\mathcal{A}_0$	$\mathcal{I}_0 + 307\mathcal{M}_0 + 259\mathcal{A}_0$
9	$81\mathcal{M}_0 + 80\mathcal{A}_0$	$45\mathcal{M}_0 + 53\mathcal{A}_0$	$\mathcal{I}_0 + 293\mathcal{M}_0 + 248\mathcal{A}_0$
12	$144\mathcal{M}_0 + 143\mathcal{A}_0$	$78\mathcal{M}_0 + 98\mathcal{A}_0$	$\mathcal{I}_0 + 810\mathcal{M}_0 + 726\mathcal{A}_0$
16	$256\mathcal{M}_0 + 255\mathcal{A}_0$	$136\mathcal{M}_0 + 164\mathcal{A}_0$	$\mathcal{I}_0 + 1673\mathcal{M}_0 + 1545\mathcal{A}_0$
18	$324\mathcal{M}_0 + 323\mathcal{A}_0$	$171\mathcal{M}_0 + 203\mathcal{A}_0$	$\mathcal{I}_0 + 1758\mathcal{M}_0 + 1632\mathcal{A}_0$
24	$576\mathcal{M}_0 + 575\mathcal{A}_0$	$300\mathcal{M}_0 + 344\mathcal{A}_0$	$\mathcal{I}_0 + 4264\mathcal{M}_0 + 4048\mathcal{A}_0$
27	$729\mathcal{M}_0 + 728\mathcal{A}_0$	$378\mathcal{M}_0 + 404\mathcal{A}_0$	$\mathcal{I}_0 + 4610\mathcal{M}_0 + 4394\mathcal{A}_0$
32	$1024\mathcal{M}_0 + 1023\mathcal{A}_0$	$528\mathcal{M}_0 + 588\mathcal{A}_0$	$\mathcal{I}_0 + 8535\mathcal{M}_0 + 8215\mathcal{A}_0$
36	$1296\mathcal{M}_0 + 1295\mathcal{A}_0$	$666\mathcal{M}_0 + 734\mathcal{A}_0$	$\mathcal{I}_0 + 9424\mathcal{M}_0 + 9100\mathcal{A}_0$

TABLE 11
Number of Ground Field Operations for Generalized OTF Arithmetic

m	$\mathcal{M}(m)$	$\mathcal{S}(m)$	$\mathcal{I}(m)$
6	$36\mathcal{M}_0 + 39\mathcal{A}_0$	$21\mathcal{M}_0 + 31\mathcal{A}_0$	$\mathcal{I}_0 + 42\mathcal{M}_0 + 48\mathcal{A}_0$
12	$144\mathcal{M}_0 + 169\mathcal{A}_0$	$78\mathcal{M}_0 + 114\mathcal{A}_0$	$\mathcal{I}_0 + 156\mathcal{M}_0 + 195\mathcal{A}_0$
18	$324\mathcal{M}_0 + 387\mathcal{A}_0$	$171\mathcal{M}_0 + 247\mathcal{A}_0$	$\mathcal{I}_0 + 363\mathcal{M}_0 + 466\mathcal{A}_0$
24	$576\mathcal{M}_0 + 701\mathcal{A}_0$	$300\mathcal{M}_0 + 422\mathcal{A}_0$	$\mathcal{I}_0 + 600\mathcal{M}_0 + 774\mathcal{A}_0$
36	$1296\mathcal{M}_0 + 1585\mathcal{A}_0$	$666\mathcal{M}_0 + 918\mathcal{A}_0$	$\mathcal{I}_0 + 1353\mathcal{M}_0 + 1753\mathcal{A}_0$

TABLE 12
Number of Ground Field Operations for Generalized OTF Arithmetic with OEF Multiplication and Squaring

m	$\mathcal{M}(m)$	$\mathcal{S}(m)$	$\mathcal{I}(m)$
6	$36\mathcal{M}_0 + 35\mathcal{A}_0$	$21\mathcal{M}_0 + 29\mathcal{A}_0$	$\mathcal{I}_0 + 42\mathcal{M}_0 + 48\mathcal{A}_0$
12	$144\mathcal{M}_0 + 143\mathcal{A}_0$	$78\mathcal{M}_0 + 98\mathcal{A}_0$	$\mathcal{I}_0 + 156\mathcal{M}_0 + 183\mathcal{A}_0$
18	$324\mathcal{M}_0 + 323\mathcal{A}_0$	$171\mathcal{M}_0 + 203\mathcal{A}_0$	$\mathcal{I}_0 + 363\mathcal{M}_0 + 412\mathcal{A}_0$
24	$576\mathcal{M}_0 + 575\mathcal{A}_0$	$300\mathcal{M}_0 + 344\mathcal{A}_0$	$\mathcal{I}_0 + 600\mathcal{M}_0 + 678\mathcal{A}_0$
36	$1296\mathcal{M}_0 + 1295\mathcal{A}_0$	$666\mathcal{M}_0 + 734\mathcal{A}_0$	$\mathcal{I}_0 + 1353\mathcal{M}_0 + 1483\mathcal{A}_0$

Multiplication and squaring in OEFs was found to be slightly more efficient (in the number of additions) than in OTFs. However, since OEF operations are directly accessible via a simple permutation conversion that comes for free, these two operations may be realized in OTFs with the same complexity as in OEFs.

TABLE 13
Number of Ground Field Operations for Generalized OTF Arithmetic when the Karatsuba-Ofman Algorithm Is Used

m	$\mathcal{M}(m)$	$\mathcal{S}(m)$	$\mathcal{I}(m)$
6	$18\mathcal{M}_0 + 67\mathcal{A}_0$	$18\mathcal{M}_0 + 40\mathcal{A}_0$	$\mathcal{I}_0 + 36\mathcal{M}_0 + 66\mathcal{A}_0$
12	$54\mathcal{M}_0 + 232\mathcal{A}_0$	$54\mathcal{M}_0 + 160\mathcal{A}_0$	$\mathcal{I}_0 + 108\mathcal{M}_0 + 287\mathcal{A}_0$
18	$108\mathcal{M}_0 + 493\mathcal{A}_0$	$108\mathcal{M}_0 + 358\mathcal{A}_0$	$\mathcal{I}_0 + 228\mathcal{M}_0 + 715\mathcal{A}_0$
24	$162\mathcal{M}_0 + 757\mathcal{A}_0$	$162\mathcal{M}_0 + 577\mathcal{A}_0$	$\mathcal{I}_0 + 324\mathcal{M}_0 + 1084\mathcal{A}_0$
36	$324\mathcal{M}_0 + 1570\mathcal{A}_0$	$324\mathcal{M}_0 + 1246\mathcal{A}_0$	$\mathcal{I}_0 + 660\mathcal{M}_0 + 2436\mathcal{A}_0$

The main advantage in using OTFs is in the recursive direct inversion method we have introduced. It was determined that OTF inversion is at least a few times more efficient than the OEF Itoh-Tsujii inversion technique, even when fast Frobenius maps apply. OTF inversion requires $m^2 + m - 2$ ground field $GF(q)$ multiplications, whereas an OTF multiplication may be realized by using m^2 ground field $GF(q)$ multiplications. Hence, for practical purposes, OTF inversion may be considered to have the same complexity as OTF/OEF multiplication, assuming the ground field $GF(q)$ inversion may be realized efficiently. Furthermore, the asymptotic complexity of OTF inversion, i.e., $O(m^2)$, is surprisingly lower than the $O(m^2 \log_2 m)$ complexity of Itoh Tsujii's inversion technique. By using the Karatsuba-Ofman algorithm, an improved version of the OTF direct inversion algorithm was presented which achieves an even better $O(m^{\log_2 3})$ asymptotic complexity.

APPENDIX

See Tables 7, 8, 9, 10, 11, 12, and 13.

ACKNOWLEDGMENTS

This work was supported by the US National Science Foundation under Grant No. ANI-0112889.

REFERENCES

[1] A. Woodbury, D.V. Bailey, and C. Paar, "Elliptic Curve Cryptography on Smart Cards without Coprocessors," *Proc. IFIP CARDIS 2000, Fourth Smart Card Research and Advanced Application Conf.*, Sept. 2000.

[2] R. Schroeppel, H. Orman, S. O'Malley, and O. Spatscheck, "Fast Key Exchange with Elliptic Curve Systems," *Proc. Advances in Cryptology—CRYPTO '95*, D. Coppersmith, ed., pp. 43-56, 1995.

[3] Ç.K. Koç and T. Acar, "Montgomery Multplication in $GF(2^k)$," *Design, Codes, and Cryptography*, vol. 14, no. 1, pp. 57-69, 1998.

[4] I.S. Hsu, T.K. Truong, L.J. Deutsch, and I.S. Reed, "A Comparison of VLSI Architecture of Finite Field Multipliers Using Dual-, Normal-, or Standard Bases," *IEEE Trans. Computers*, vol. 37, no. 6, pp. 735-739, June 1988.

[5] T. Itoh and S. Tsujii, "A Fast Algorithm for Computing Multiplicative Inverses in $GF(2^m)$ Using Normal Bases," *Information and Computation*, vol. 78, pp. 171-177, 1988.

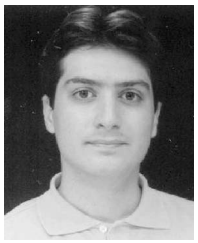
[6] B. Sunar, "Fast Galois Field Arithmetic for Elliptic Curve Cryptography and Error Control Codes," PhD thesis, Dept. of Electrical & Computer Eng., Oregon State Univ., Corvallis, Nov. 1998.

[7] W. Geiselmann and D. Gollmann, "Self-Dual Bases in F_{q^n}," *Designs, Codes, and Cryptography*, vol. 3, pp. 333-345, 1993.

[8] S.T.J. Fenn, M. Benaissa, and D. Taylor, "Finite Field Inversion over the Dual Base," *IEEE Trans. Very Large Scale Integration (VLSI) Systems*, vol. 4, pp. 134-136, Mar. 1996.

[9] M.A. Hasan, "Double-Basis Multiplicative Inversion over $GF(2^m)$," *IEEE Trans. Computers*, vol. 47, no. 9, pp. 960-970, Sept. 1998.

[10] D.V. Bailey and C. Paar, "Efficient Arithmetic in Finite Field Extensions with Application in Elliptic Curve Cryptography," *J. Cryptology,* vol. 14, no. 3, pp. 153-176, 2001.
[11] D.V. Bailey and C. Paar, "Optimal Extension Fields for Fast Arithmetic in Public-Key Algorithms," *Proc. Advances in Cryptology—CRYPTO '98,* H. Krawczyk, ed., pp. 472-485, 1998.
[12] R. Lidl and H. Niederreiter, *Finite Fields,* vol. 20 of *Encyclopedia of Math. and Its Applications.* Reading, Mass.: Addison-Wesley, 1983.
[13] J. Guajardo and C. Paar, "Itoh-Tsujii Inversion in Standard Basis and Its Application in Cryptography," *Design, Codes, and Cryptography,* no. 25, pp. 207-216, 2002.
[14] G.I. Davida, "Inverse of Elements of a Galois Field," *Electronic Letters,* vol. 8, pp. 518-520, Oct. 1972.
[15] M. Morii and M. Kasahara, "Efficient Construction of Gate Circuit for Computing Multiplicative Inverses over $GF(2^m)$," *Trans. IEICE,* vol. E 72, pp. 37-42, Jan. 1989.
[16] A. Karatsuba and Y. Ofman, "Multiplication of Multidigit Numbers on Automata," *Sov. Phys. Dokl. (English translation),* vol. 7, no. 7, pp. 595-596, 1963.
[17] P. Gaudry, F. Hess, and N.P. Smart, "Constructive and Destructive Facets of Weil Descent on Elliptic Curves," *J. Cryptology,* vol. 15, pp. 19-46, 2002.
[18] N. P. Smart, "How Secure Are Elliptic Curves over Composite Extension Fields?" Technical Report CSTR-00-017, Dept. of Computer Science, Univ. of Bristol, Nov. 2000.
[19] N.P. Smart, "A Comparison of Different Finite Fields for Use in Elliptic Curve Cryptosystems," Technical Report CSTR-00-007, Dept. of Computer Science, Univ. of Bristol, June 2000.



Selçuk Baktir received the BSc degree in electrical engineering from Bilkent University, Ankara, Turkey, in 2001 and the MSc degree in electrical and computer engineering from Worcester Polytechnic Institute, Worcester, Massachusetts, in 2003. He is currently pursuing the PhD degree. His research interests include cryptography and information security, network security, finite fields, number theory, discrete mathematics, and algebra. He is a student member of the IEEE, the IEEE Computer Society, the IEEE Information Theory Society, the IEEE Communications Society, and the International Association of Cryptologic Research (IACR).

Berk Sunar received the BSc degree in electrical and electronics engineering from Middle East Technical University in 1995 and the PhD degree in electrical and computer engineering (ECE) from Oregon State University in December 1998. After briefly working as a member of the research faculty at Oregon State University, he joined Worcester Polytechnic Institute as an assistant professor. He currently heads the Cryptography and Information Security Laboratory (CRIS). He received the US National Science Foundation CAREER award in 2002. His research interests include finite fields, elliptic curve cryptography, low-power crypographic hardware design, and computer arithmetic. He is a member of the IEEE, the IEEE Computer Society, the ACM, and the International Association of Cryptologic Research (IACR) professional societies.