Multiplication and Squaring on Pairing-Friendly Fields

Augusto Jun Devegili[1]*, Colm Ó hÉigeartaigh[2], Michael Scott[2], and Ricardo Dahab[1]

[1] Instituto de Computação, Universidade Estadual de Campinas
Caixa Postal 6176
13084-971 Campinas, SP, Brazil
{augusto,rdahab}@ic.unicamp.br

[2] School of Computing, Dublin City University
Dublin 9, Ireland
{coheigeartaigh,mike}@computing.dcu.ie

Abstract. Pairing-friendly fields are finite fields that are suitable for the implementation of cryptographic bilinear pairings. In this paper we review multiplication and squaring methods for pairing-friendly fields $\mathbb{F}_{p^k}$ with $k \in \{2, 3, 4, 6\}$. For composite k, we consider every possible towering construction. We compare the methods to determine which is most efficient based on the number of basic $\mathbb{F}_p$ operations, as well as the best constructions for these finite extension fields. We also present experimental results for every method.

Keywords: Finite fields, pairings, efficient implementation.

1 Introduction

In recent years, bilinear pairings have been used to construct new cryptographic schemes with many novel and exciting properties, such as the one-round, three-party key agreement protocol of Joux [11] and the identity-based encryption scheme of Boneh and Franklin [4]. Pairing computation is generally the most intensive computational requirement in pairing-based schemes, and therefore it is of paramount importance to implement pairings in an efficient manner.

Pairing-friendly fields were introduced by Koblitz and Menezes [14] as an efficient way to implement the various finite fields extensions required for pairing-based cryptography. There are two advantages to constructing finite fields in this manner. Firstly, they provide for efficient arithmetic, as multiplication by certain terms that define the field is trivial. Secondly, they allow for an easy way to analyse the cost of multiplications, something which has been utilised by various parties ([7, 8, 10]) to conduct a theoretical analysis of the cost of computing pairings using these fields.

* Funded by the Brazilian Government/Coordination for the Improvement of Higher Education Personnel (CAPES)

However, these analyses consider that squaring takes as much time or as many ground field operations as multiplication. Let $k = 2^i 3^j$ be the extension degree of a pairing-friendly field. Koblitz and Menezes [14] define the function $\nu(k) = 3^i 5^j$ to compute the number of ground field operations needed to multiply two elements in $\mathbb{F}_{p^k}$ — multiplication in a quadratic extension can be computed with three multiplications in the ground field using Karatsuba, and multiplication in a cubic extension can be computed with five multiplications in the ground field using Toom-Cook-3. However, there are situations where Karatsuba is more efficient than Toom-Cook for cubic extensions. Furthermore, squaring in quadratic extensions needs fewer ground field multiplications. As pairing-friendly fields have an even extension degree in order to facilitate the denominator elimination optimisation, there is always at least one quadratic extension. Therefore, the number of ground field operations for squaring is certainly less than that provided by the function $\nu(k)$.

The contribution of this paper is as follows. In Section 2, we review the definition of pairing-friendly fields and methods for multiplication and squaring. In Sections 3, 4, 5, and 6, we analyse methods for multiplication and squaring in quadratic, cubic, quartic and sextic extensions. In Section 7 and the appendix, we present our experimental results on the implementation of every method for multiplication and squaring in the various extension fields, as well as different constructions for composite extension degrees. We conclude in Section 8, and provide recommendations on the most efficient methods for each extension field.

2 Pairing-Friendly Fields

When computing pairings, one must construct a representation for the finite field $\mathbb{F}_{p^k}$, where k is the embedding degree of the curve in question. The finite field $\mathbb{F}_{p^k}$ is implemented as $\mathbb{F}_p[X]/(f(X))$, where $f(X)$ is an irreducible polynomial of degree k over $\mathbb{F}_p$. Elements of $\mathbb{F}_{p^k}$ are represented using the polynomial basis $(1, X, X^2, \ldots, X^{k-1})$, where X is a root of the irreducible polynomial over $\mathbb{F}_{p^k}$. Multiplication is computed as a multiplication of polynomials, followed by a reduction modulo the irreducible polynomial $f(X)$, which can be built into the formulæ for multiplication.

Koblitz and Menezes [14] define a field $\mathbb{F}_{p^k}$ as being pairing-friendly if the prime characteristic $p \equiv 1 \pmod{12}$ and the embedding degree $k = 2^i 3^j$, $i > 0$. Let β be an element of $\mathbb{F}_p$ that is neither a square nor a cube in $\mathbb{F}_p$. Then the binomial $X^k - \beta$ is irreducible over $\mathbb{F}_p[X]$ and hence defines the field $\mathbb{F}_{p^k}$. Therefore, the field $\mathbb{F}_{p^k}$ can be constructed as a series of quadratic and cubic Kummer extensions by adjoining either the square root or cube root of β, and successively the square root or cube root of that, until the field is constructed. If $\mathbb{F}_{p^k}$ can be constructed by towering quadratic extensions alone, i.e. if $j = 0$, then the condition on the prime p simplifies to $p \equiv 1 \pmod{4}$, and β need only be a non-square in $\mathbb{F}_p$. Note that $k = 2^i 3^j$ supports almost all embedding degrees used in practise, i.e. $k = 2, 4, 6, 8, 12, 24$, etc.

The advantage of constructing finite fields in this manner is that a small value of β can be exploited to simplify arithmetic. Multiplication by β is achieved by a simple reordering of terms and roughly $|\beta|$ additions, which is far less expensive than a general multiplication.

As this paper focuses on multiplication and squaring, we recall briefly various techniques in the literature (see [2, 13] for an overview) for multiplying two n-digit integers. The basic Schoolbook multiplication method for multiplying two n-word integers has complexity $O(n^2)$. Karatsuba [12] presented the first sub-quadratic time multiplication method, which has complexity $O(n^{\lg(3)}) \approx O(n^{1.585})$. Karatsuba's method proceeds by splitting the integers into two parts, and therefore it is natural to consider using it to implement multiplication in quadratic field extensions. Toom [19] and later Cook [6] extended Karatsuba's method by splitting the integers into three parts. The resulting Toom-Cook algorithm has complexity $O(n^{1.46})$, and can be used to implement multiplication in a cubic field extension.

Assume $\mathbb{F}_{p^k}$ is a direct extension of $\mathbb{F}_p$ and an element $a \in \mathbb{F}_{p^k}$ is represented as $a_0 + a_1X + \cdots + a_{k-1}X^{k-1}$. We perform multiplication and squaring of elements in $\mathbb{F}_{p^k}$ as the polynomial multiplication (resp. squaring) in $\mathbb{F}_p[X]$ and then reduction by $(X^k - \beta)$. The Schoolbook method for multiplying two elements $a, b \in \mathbb{F}_{p^k}$ is defined as

$$c = ab = \left(\sum_{i=0}^{k-1} a_i X^i\right)\left(\sum_{i=0}^{k-1} b_i X^i\right) \pmod{(X^k - \beta)},$$

and the coefficients of c after the reduction by $(X^k - \beta)$ are

$$c_i = \sum_{j=0}^{i} a_j b_{i-j} + \beta\left(\sum_{j=i+1}^{k-1} a_j b_{i-j+k}\right) \pmod{p}.$$

The Karatsuba method for computing the product $c = ab \in \mathbb{F}_{p^2}$ proceeds by precomputing $v_0 = a_0b_0$, $v_1 = a_1b_1$ and then

$$\begin{aligned} c_0 &= v_0 + \beta v_1 \\ c_1 &= (a_0 + a_1)(b_0 + b_1) - v_0 - v_1. \end{aligned}$$

To compute the product $c = ab \in \mathbb{F}_{p^3}$, the Karatsuba method precomputes $v_0 = a_0b_0$, $v_1 = a_1b_1$, $v_2 = a_2b_2$ and then

$$\begin{aligned} c_0 &= v_0 + \beta\left((a_1 + a_2)(b_1 + b_2) - v_1 - v_2\right) \\ c_1 &= (a_0 + a_1)(b_0 + b_1) - v_0 - v_1 + \beta v_2 \\ c_2 &= (a_0 + a_2)(b_0 + b_2) - v_0 + v_1 - v_2. \end{aligned}$$

Weimerskirch and Paar discuss in [20] the Karatsuba method and its application to arbitrary degree polynomials.

The Toom-Cook method is based on interpolation: given the evaluation of a degree-n polynomial at $n+1$ distinct points, it is possible to uniquely determine it. Before the reduction modulo $(X^k - \beta)$, the product of two elements $a, b \in \mathbb{F}_{p^k}$ is a degree-$(2k-2)$ polynomial. The Toom-Cook method applied to multiplication of polynomials is essentially an interpolation: choose a family $\{x_i\}_{0 \le i < 2k-1}$ of distinct points in $\mathbb{F}_p$, evaluate the product $a(x_i)b(x_i) \in \mathbb{F}_p$ on every point of the family, and interpolate the evaluation points in order to (uniquely) obtain the product $ab \in \mathbb{F}_p[X]$. Then reduce it modulo $(X^k - \beta)$ to obtain $c = ab \in \mathbb{F}_{p^k}$.

The efficiency of the Toom-Cook method depends on the choice of the points x_i and the method for interpolation, and it is still not clear which yields the most efficient Toom-Cook implementation. See [3, 5] for a recent discussion on these issues. In this text, we use the points $0 < x_i < k-2$ and their additive inverses, $k-2$, 0, and ∞ for computing the evaluation points: these are the values that give the least absolute value integers in the formula, and we use the Lagrange method for interpolation.

The Toom-Cook method needs to compute divisions by integers $2, 3, \ldots, 2(k-1)$ [2, 5]. Division by two can be efficiently implemented using bit shifts, but division by integers greater than two is more costly. Notice that in the context of pairing computation the pairing value may be freely multiplied by any integer (in fact, by any element of a proper subfield of $\mathbb{F}_{p^k}$) without affecting the value of the pairing — the final exponentiation wipes out these factors [1, Corollary 1]. Multiplying the product $ab \in \mathbb{F}_{p^k}$ by the least common multiple of all the divisors that appear in the Toom-Cook formulæ avoids all divisions and may be used as a replacement for the original Toom-Cook in the context of pairing computation. We call this method Toom-Cook-x.

Other methods for multiplication and squaring in finite extension fields include the asymmetric squaring formulæ by Chung and Hasan [5], and Montgomery's improved Karatsuba-like formulæ [17].

Our analysis of the cost for multiplying and squaring elements in finite extension fields is based on the following operations on the ground field: multiplication (M), squaring (S), addition or subtraction (A), division by 2 (D_2), multiplication by β (B), and multiplication by small (word-size) integers $(M_{\mathbb{Z}})$. Note that A, D_2, B, $M_{\mathbb{Z}}$ are linear on the extension degree, while M and S are super-linear.

Note that it is possible to implement $\mathbb{F}_{p^k}$ operations using lazy reduction (or accumulation and reduction) [9, 15, 18]. In this technique, instead of computing a modular reduction at every ground field operation, the operands are computed as integers, accumulated during various operations and later reduced modulo p. This saves reduction operations at the expense of greater memory requirements, poor interaction with Montgomery reduction and more additions. We leave lazy reduction outside the scope of the present work, but it might be worthwhile to further research this matter.

A remark on multiplication by β: throughout this text, $x + \beta y$ is costed as $A+B$ (one addition and one multiplication by β). If $\beta = -2$, then $x+\beta y = x-2y$, so the actual cost is $2A$. Another example: $2x+\beta y$ is costed as $2A+B$. Again, if $\beta = -2$, then $2x+\beta y = 2(x-y)$, and the actual cost is $2A$. Yet another example: $x + \beta x$ is costed as $A + B$. If $\beta = -1$, then the actual cost is 0; if $\beta = -2$, the actual cost is A. Depending on the value of β, the actual cost of a formula varies.

It is important to bear this is mind while reading the next sections. For the sake of simplicity, it is possible to assume that $B = (|\beta| - 1)$ as an approximation.

3 Quadratic extensions

We construct a quadratic extension as $\mathbb{F}_{p^2} = \mathbb{F}_p[X]/(X^2 - \beta)$, where β is a quadratic non-residue in $\mathbb{F}_p$. An element $\alpha \in \mathbb{F}_{p^2}$ is represented as $\alpha_0 + \alpha_1 X$, where $\alpha_i \in \mathbb{F}_p$.

The Schoolbook method computes the product $c = ab$ as

$$\begin{aligned} c_0 &= a_0 b_0 + \beta a_1 b_1 \\ c_1 &= a_0 b_1 + a_1 b_0, \end{aligned}$$

which costs $4M + 2A + B$, and the square $c = a^2$ as

$$\begin{aligned} c_0 &= a_0^2 + \beta a_1^2 \\ c_1 &= 2 a_0 a_1, \end{aligned}$$

which takes $M + 2S + 2A + B$. The Karatsuba method computes $c = ab$ by first precomputing the values $v_0 = a_0 b_0$, $v_1 = a_1 b_1$. Then the multiplication is performed as

$$\begin{aligned} c_0 &= v_0 + \beta v_1 \\ c_1 &= (a_0 + a_1)(b_0 + b_1) - v_0 - v_1, \end{aligned}$$

which costs $3M + 5A + B$ in total. Adapting the Karatsuba formula for squaring, we first precompute $v_0 = a_0^2, v_1 = a_1^2$ and then compute the square $c = a^2$ as

$$\begin{aligned} c_0 &= v_0 + \beta v_1 \\ c_1 &= (a_0 + a_1)^2 - v_0 - v_1, \end{aligned}$$

which costs $3S + 4A + B$. There is a well-known squaring formula for complex arithmetic[3] that computes the square $c_0 + c_1 i = (a_0 + a_1 i)^2$ as

$$\begin{aligned} c_0 &= (a_0 + a_1)(a_0 - a_1) \\ c_1 &= 2 a_0 a_1. \end{aligned}$$

This is actually a special case of a squaring formula that we refer to as complex squaring. We precompute $v_0 = a_0 a_1$, and then the square $c = a^2$ is computed as

$$\begin{aligned} c_0 &= (a_0 + a_1)(a_0 + \beta a_1) - v_0 - \beta v_0 \\ c_1 &= 2 v_0, \end{aligned}$$

which takes $2M + 4A + 2B$.

[3] The arithmetic of complex numbers $a + bi$ where a, b are real numbers and i is the imaginary square root of -1.

Tables 1 and 2 list the multiplication and squaring costs of the methods above, and the conditions where one method is faster than the others. For sufficiently large moduli, Karatsuba is the most efficient multiplication method. Under the assumption that a modular squaring in $\mathbb{F}_p$ takes approximately the same time as modular multiplication in $\mathbb{F}_p$, complex squaring is the most efficient method.

Table 1. Summary of multiplication costs for $\mathbb{F}_{p^2}$

$\mathbb{F}_{p^2}$ Method	> Linear	Linear
Schoolbook	$4M$	$2A + B$
Karatsuba	$3M$	$5A + B$

Table 2. Summary of squaring costs for $\mathbb{F}_{p^2}$

$\mathbb{F}_{p^2}$ Method	> Linear	Linear
Schoolbook	$M + 2S$	$2A + B$
Karatsuba	$3S$	$4A + 2B$
Complex	$2M$	$4A + 2B$

4 Cubic extensions

We construct a cubic extension as $\mathbb{F}_{p^3} = \mathbb{F}_p[X]/(X^3 - \beta)$, where β is a cubic non-residue in $\mathbb{F}_p$. An element $\alpha \in \mathbb{F}_{p^3}$ is represented as $\alpha_0 + \alpha_1 X + \alpha_2 X^2$, where $\alpha_i \in \mathbb{F}_p$.

The Schoolbook method computes the product $c = ab$ as

$$\begin{aligned} c_0 &= a_0 b_0 + \beta(a_1 b_2 + a_2 b_1) \\ c_1 &= a_0 b_1 + a_1 b_0 + \beta a_2 b_2 \\ c_2 &= a_0 b_2 + a_1 b_1 + a_2 b_0, \end{aligned}$$

which takes $9M + 6A + 2B$, and the square $c = a^2$ as

$$\begin{aligned} c_0 &= a_0^2 + 2\beta a_1 a_2 \\ c_1 &= 2a_0 a_1 + \beta a_2^2 \\ c_2 &= a_1^2 + 2a_0 a_2, \end{aligned}$$

which costs $3M + 3S + 6A + 2B$. The Karatsuba method for computing the product $c = ab$ starts by precomputing the values $v_0 = a_0 b_0$, $v_1 = a_1 b_1$, $v_2 =$

a_2b_2, which costs $3M$. Then the multiplication is performed as

$$\begin{aligned}
c_0 &= v_0 + \beta((a_1 + a_2)(b_1 + b_2) - v_1 - v_2)\\
c_1 &= (a_0 + a_1)(b_0 + b_1) - v_0 - v_1 + \beta v_2\\
c_2 &= (a_0 + a_2)(b_0 + b_2) - v_0 + v_1 - v_2,
\end{aligned}$$

which takes $3M + 15A + 2B$, for a total of $6M + 13A + 2B$. Adapting the Karatsuba formula to compute the square $c = a^2$, we precompute the values $v_0 = a_0^2$, $v_1 = a_1^2$, $v_2 = a_2^2$, which costs $3S$. Then the squaring is performed as

$$\begin{aligned}
c_0 &= v_0 + \beta((a_1 + a_2)^2 - v_1 - v_2)\\
c_1 &= (a_0 + a_1)^2 - v_0 - v_1 + \beta v_2\\
c_2 &= (a_0 + a_2)^2 - v_0 + v_1 - v_2,
\end{aligned}$$

which takes $3S + 12A + 2B$, for a total of $6S + 13A + 2B$. The Toom-Cook-3 method starts by computing five interpolation points – these are evaluations of the product $a(X)b(X)$ with $X \in \{0, \pm 1, 2, \infty\}$:

$$\begin{aligned}
v_0 &= a(0)b(0) &&= a_0b_0\\
v_1 &= a(1)b(1) &&= (a_0 + a_1 + a_2)(b_0 + b_1 + b_2)\\
v_2 &= a(-1)b(-1) &&= (a_0 - a_1 + a_2)(b_0 - b_1 + b_2)\\
v_3 &= a(2)b(2) &&= (a_0 + 2a_1 + 4a_2)(b_0 + 2b_1 + 4b_2)\\
v_4 &= a(\infty)b(\infty) &&= a_2b_2,
\end{aligned}$$

which costs $5M + 14A$. Notice that for squaring the interpolation points are computed as

$$\begin{aligned}
v_0 &= a(0)^2 &&= a_0^2\\
v_1 &= a(1)^2 &&= (a_0 + a_1 + a_2)^2\\
v_2 &= a(-1)^2 &&= (a_0 - a_1 + a_2)^2\\
v_3 &= a(2)^2 &&= (a_0 + 2a_1 + 4a_2)^2\\
v_4 &= a(\infty)^2 &&= a_2^2,
\end{aligned}$$

which takes $5S + 7A$. Then the interpolation is performed as

$$\begin{aligned}
c_0 &= v_0 + \beta((1/2)v_0 - (1/2)v_1 - (1/6)v_2 + (1/6)v_3 - 2v_4)\\
c_1 &= -(1/2)v_0 + v_1 - (1/3)v_2 - (1/6)v_3 + 2v_4 + \beta v_4\\
c_2 &= -v_0 + (1/2)v_1 + (1/2)v_2 - v_4.
\end{aligned}$$

If we compute the product $c = 6ab$, we remove all the divisions needed by Toom-Cook-3. Precomputation is the same as in Toom-Cook-3, and the interpolation is performed as

$$\begin{aligned}
c_0 &= 6v_0 + \beta(3v_0 - 3v_1 - v_2 + v_3 - 12v_4)\\
c_1 &= -3v_0 + 6v_1 - 2v_2 - v_3 + 12v_4 + 6\beta v_4\\
c_2 &= -6v_0 + 3v_1 + 3v_2 - 6v_4.
\end{aligned}$$

Considering that additions chains are faster than multiplication by the coefficients in the formula, we can compute the product $c = 6ab$ using Toom-Cook-3x with cost $5M + 40A + 2B$. If $\beta = -2$, then the cost is reduced to $5M + 35A$.

Chung and Hasan [5] have recently derived three asymmetric squaring formulæ for degree-2 polynomials. For the first formula (CH-SQR1), we first precompute the values

$$\begin{aligned}
s_0 &= a_0^2 \\
s_1 &= 2a_0a_1 \\
s_2 &= (a_0 + a_1 - a_2)(a_0 - a_1 - a_2) \\
s_3 &= 2a_1a_2 \\
s_4 &= a_2^2,
\end{aligned}$$

which costs $3M + 2S + 5A$. Then the squaring is computed as

$$\begin{aligned}
c_0 &= s_0 + \beta s_3 \\
c_1 &= s_1 + \beta s_4 \\
c_2 &= s_1 + s_2 + s_3 - s_0 - s_4,
\end{aligned}$$

which costs $6A + 2B$, giving a total cost of $3M + 2S + 11A + 2B$. For the second formula (CH-SQR2), we first precompute the values

$$\begin{aligned}
s_0 &= a_0^2 \\
s_1 &= 2a_0a_1 \\
s_2 &= (a_0 - a_1 + a_2)^2 \\
s_3 &= 2a_1a_2 \\
s_4 &= a_2^2,
\end{aligned}$$

which costs $2M + 3S + 4A$. Then the squaring is computed as

$$\begin{aligned}
c_0 &= s_0 + \beta s_3 \\
c_1 &= s_1 + \beta s_4 \\
c_2 &= s_1 + s_2 + s_3 - s_0 - s_4,
\end{aligned}$$

which costs $6A + 2B$, giving a total cost of $2M + 3S + 10A + 2B$. For the third formula (CH-SQR3), we first precompute the values

$$\begin{aligned}
s_0 &= a_0^2 \\
s_1 &= (a_0 + a_1 + a_2)^2 \\
s_2 &= (a_0 - a_1 + a_2)^2 \\
s_3 &= 2a_1a_2 \\
s_4 &= a_2^2 \\
t_1 &= (s_1 + s_2)/2,
\end{aligned}$$

which costs $1M + 4S + 5A + 1D_2$, where D_2 denotes a division by 2. Then the squaring is computed as

$$\begin{aligned} c_0 &= s_0 + \beta s_3 \\ c_1 &= s_1 - s_3 - t_1 + \beta s_4 \\ c_2 &= t_1 - s_0 - s_4, \end{aligned}$$

which costs $6A + 2B$, giving a total cost of $1M + 4S + 11A + 2B + 1D_2$. As with Toom-Cook, we can compute the square $c = 2a^2$ to avoid divisions in the CH-SQR3 formula. We precompute s_0, s_1, s_2, s_3, s_4 as in CH-SQR3, which costs $1M + 4S + 4A$. Then the square is computed as

$$\begin{aligned} c_0 &= 2s_0 + 2\beta s_3 \\ c_1 &= s_1 - s_2 - 2s_3 + 2\beta s_4 \\ c_2 &= -2s_0 + s_1 + s_2 - 2s_4, \end{aligned}$$

which costs $10A + 2B$, for a total cost of $1M + 4S + 14A + 2B$.

Table 3 lists the multiplicative costs of each of these methods, and the conditions where each method is faster than the others. With five multiplications, Toom-Cook-3x is the most efficient method for multiplication in cubic extensions for sufficiently large moduli.

Table 3. Summary of multiplicative costs for cubic extensions

Method	> Linear	Linear
Schoolbook	$9M$	$6A + 2B$
Karatsuba	$6M$	$13A + 2B$
Toom-Cook-3x	$5M$	$33A + 2B$

Table 4. Summary of squaring costs for cubic extensions

Method	> Linear	Linear
Schoolbook	$3M + 3S$	$6A + 2B$
Karatsuba	$6S$	$13A + 2B$
Toom-Cook-3x	$5S$	$33A + 2B$
CH-SQR1	$3M + 2S$	$11A + 2B$
CH-SQR2	$2M + 3S$	$10A + 2B$
CH-SQR3	$1M + 4S$	$11A + 2B + 1D_2$
CH-SQR3x	$1M + 4S$	$14A + 2B$

5 Quartic extensions

We consider two possibilities for building a quartic extension: quadratic over quadratic and direct quartic.

5.1 Quadratic over quadratic

We construct a quartic extension as $\mathbb{F}_{p^4} = \mathbb{F}_{p^2}[Y]/(Y^2 - \gamma)$, where $\gamma = \sqrt{\beta}$ is a quadratic non-residue in $\mathbb{F}_{p^2}$. An element $\alpha \in \mathbb{F}_{p^4}$ is represented as $\alpha_0 + \alpha_1 Y$, where $\alpha_i \in \mathbb{F}_{p^2}$.

Multiplication and squaring can be computed with the same methods for quadratic extensions described in Section 3, replacing multiplication by β with multiplication by γ. Notice that $\beta \in \mathbb{F}_p$ and $\gamma \in \mathbb{F}_{p^2}$. In fact, γ is represented by $0 + 1X = X$, and the product $b = \gamma a$, with $a = a_0 + a_1 Y$ and $a_i = a_{i,0} + a_{i,1} X$, is computed as

$$\begin{aligned} b_i &= \gamma a_i \\ &= X(a_{i,0} + a_{i,1} X) \\ &= \beta a_{i,1} + a_{i,0} X, \end{aligned}$$

amounting to four permutations and two multiplications by β.

The corresponding costs based on $\mathbb{F}_p$ operations depend on the choice of multiplication methods for the (bottom) quadratic extension field. Table 5 shows the multiplicative costs for $\mathbb{F}_{p^4}$ as a quadratic over quadratic, and Table 6 shows the squaring costs.

Table 5. Summary of multiplicative costs for quartic extensions as quadratic over quadratic

$\mathbb{F}_{p^4}$ method	Schoolbook		Karatsuba	
$\mathbb{F}_{p^2}$ method	> Linear	Linear	> Linear	Linear
Schoolbook	$16M$	$12A + 5B$	$12M$	$16A + 4B$
Karatsuba	$12M$	$24A + 5B$	$9M$	$25A + 4B$

Table 6. Summary of squaring costs for quartic extensions as quadratic over quadratic

$\mathbb{F}_{p^4}$ method	Schoolbook		Karatsuba		Complex	
$\mathbb{F}_{p^2}$ method	> Linear	Linear	> Linear	Linear	> Linear	Linear
Schoolbook	$6M + 4S$	$10A + 4B$	$3M + 6S$	$14A + 5B$	$8M$	$12A + 4B$
Karatsuba	$3M + 6S$	$17A + 6B$	$9S$	$20A + 8B$	$6M$	$18A + 4B$
Karatsuba/Complex	$7M$	$17A + 6B$	$6M$	$20A + 8B$		

5.2 Direct quartic

We construct a quartic extension as $\mathbb{F}_{p^4} = \mathbb{F}_p[X]/(X^4 - \beta)$, where β is a quartic non-residue in $\mathbb{F}_p$. An element $\alpha \in \mathbb{F}_{p^4}$ is represented as $\alpha_0 + \alpha_1 X + \alpha_2 X^2 + \alpha_3 X^3$, where $\alpha_i \in \mathbb{F}_p$.

The Schoolbook method computes the product $c = ab$ as

$$\begin{aligned}
c_0 &= a_0b_0 + \beta(a_1b_3 + a_2b_2 + a_3b_1)\\
c_1 &= a_0b_1 + a_1b_0 + \beta(a_2b_3 + a_3b_2)\\
c_2 &= a_0b_2 + a_1b_1 + a_2b_0 + \beta a_3b_3\\
c_3 &= a_0b_3 + a_1b_2 + a_2b_1 + a_3b_0,
\end{aligned}$$

which costs $16M + 12A + 3B$, and the square $c = a^2$ as

$$\begin{aligned}
c_0 &= a_0^2 + \beta(2a_1a_3 + a_2^2)\\
c_1 &= 2(a_0a_1 + \beta a_2a_3)\\
c_2 &= 2a_0a_2 + a_1^2 + \beta a_3^2\\
c_3 &= 2(a_0a_3 + a_1a_2),
\end{aligned}$$

which costs $6M + 4S + 10A + 3B$.

For the Toom-Cook-4 method, we define the set of points $P = \{0, \pm1, \pm2, 3, \infty\}$ and we precompute the following evaluation points for multiplication:

$$\begin{aligned}
v_0 &= a(0)b(0) &&= a_0b_0\\
v_1 &= a(1)b(1) &&= (a_0 + a_1 + a_2 + a_3)(b_0 + b_1 + b_2 + b_3)\\
v_2 &= a(-1)b(-1) &&= (a_0 - a_1 + a_2 - a_3)(b_0 - b_1 + b_2 - b_3)\\
v_3 &= a(2)b(2) &&= (a_0 + 2a_1 + 4a_2 + 8a_3)(b_0 + 2b_1 + 4b_2 + 8b_3)\\
v_4 &= a(-2)b(-2) &&= (a_0 - 2a_1 + 4a_2 - 8a_3)(b_0 - 2b_1 + 4b_2 - 8b_3)\\
v_5 &= a(3)b(3) &&= (a_0 + 3a_1 + 9a_2 + 27a_3)(b_0 + 3b_1 + 9b_2 + 27b_3)\\
v_6 &= a(\infty)b(\infty) &&= a_3b_3,
\end{aligned}$$

which costs $7M + 44A$, and the following evaluation points for squaring:

$$\begin{aligned}
v_0 &= a(0)^2 &&= a_0^2\\
v_1 &= a(1)^2 &&= (a_0 + a_1 + a_2 + a_3)^2\\
v_2 &= a(-1)^2 &&= (a_0 - a_1 + a_2 - a_3)^2\\
v_3 &= a(2)^2 &&= (a_0 + 2a_1 + 4a_2 + 8a_3)^2\\
v_4 &= a(-2)^2 &&= (a_0 - 2a_1 + 4a_2 - 8a_3)^2\\
v_5 &= a(3)^2 &&= (a_0 + 3a_1 + 9a_2 + 27a_3)^2\\
v_6 &= a(\infty)^2 &&= a_3^2,
\end{aligned}$$

which costs $7S + 22A$. The interpolation is performed as

$$\begin{aligned}
c_0 &= v_0 + \beta((1/4)v_0 - (1/6)(v_1 + v_2) + (1/24)(v_3 + v_4) - 5v_6)\\
c_1 &= -(1/3)v_0 + v_1 - (1/2)v_2 - (1/4)v_3 + (1/20)v_4 + (1/30)v_5 - 12v_6\\
&\quad + \beta(-(1/12)(v_0 - v_1) + (1/24)(v_2 - v_3) - (1/120)(v_4 - v_5) - 3v_6)\\
c_2 &= -(5/4)v_0 + (2/3)(v_1 + v_2) - (1/24)(v_3 + v_4) + 4v_6 + \beta v_6\\
c_3 &= (1/12)(5v_0 - 7v_1) - (1/24)(v_2 - 7v_3 + v_4 + v_5) + 15v_6.
\end{aligned}$$

If we compute $c = 120ab$ we eliminate all the divisions needed by the Toom-Cook-4 method:

$$\begin{aligned}
c_0 &= 120v_0 + \beta(30v_0 - 20(v_1 + v_2) + 5(v_3 + v_4) - 600v_6)\\
c_1 &= -40v_0 + 120v_1 - 60v_2 - 30v_3 + 6v_4 + 4v_5 - 1440v_6\\
&\quad + \beta(-10(v_0 - v_1) + 5(v_2 - v_3) - v_4 + v_5 - 360v_6)\\
c_2 &= -150v_0 + 80(v_1 + v_2) - 5(v_3 + v_4) + 480v_6 + 120\beta v_6\\
c_3 &= 10(5v_0 - 7v_1) - 5(v_2 - 7v_3 + v_4 + v_5) + 1800v_6,
\end{aligned}$$

which costs $23M_{\mathbb{Z}} + 32A + 3B$. Then the total cost for Toom-Cook-4x multiplication is $7M + 23M_{\mathbb{Z}} + 76A + 3B$, and the total cost for Toom-Cook-4x squaring is $7S + 23M_{\mathbb{Z}} + 54A + 3B$.

Table 7 shows the multiplicative costs for direct quartic extensions, and Table 8 shows the squaring costs for direct quartic extensions.

Table 7. Summary of multiplicative costs for quartic extensions as direct quartic

	> Linear	Linear
Schoolbook	$16M$	$12A + 3B$
Toom-Cook-4x	$7M$	$23M_{\mathbb{Z}} + 76A + 3B$

Table 8. Summary of squaring costs for quartic extensions as direct quartic

	> Linear	Linear
Schoolbook	$6M + 4S$	$10A + 3B$
Toom-Cook-4x	$7S$	$23M_{\mathbb{Z}} + 54A + 3B$

5.3 Isomorphism between the representations

Let $a \in \mathbb{F}_{p^4}$ be an element in the representation of the field as a quadratic over quadratic, and let $b \in \mathbb{F}_{p^4}$ be an element in the representation of the field as a direct quartic extension. We can write a, b as

$$\begin{aligned} a &= (a_{0,0} + a_{0,1}\sqrt{\beta}) + (a_{1,0} + a_{1,1}\sqrt{\beta})\sqrt[4]{\beta} \\ &= a_{0,0} + a_{0,1}\sqrt{\beta} + a_{1,0}\sqrt[4]{\beta} + a_{1,1}\sqrt[4]{\beta^3} \\ b &= b_0 + b_1\sqrt[4]{\beta} + b_2\sqrt{\beta} + b_3\sqrt[4]{\beta^3}, \end{aligned}$$

so the isomorphism between the representations is efficiently computed by the permutation

$$\begin{bmatrix} a_{0,0} & a_{0,1} & a_{1,0} & a_{1,1} \\ b_0 & b_2 & b_1 & b_3 \end{bmatrix}.$$

6 Sextic extensions

We consider three possibilities for building a sextic extension: quadratic over cubic, cubic over quadratic and direct sextic.

6.1 Quadratic over cubic

We build $\mathbb{F}_{p^6}$ as $\mathbb{F}_{p^3}[Y]/(Y^2 - \gamma)$, where $\mathbb{F}_{p^3} = \mathbb{F}_p[X]/(X^3 - \beta)$, β is a cubic non-residue in $\mathbb{F}_p$ and $\gamma = \sqrt[3]{\beta}$ is a quadratic non-residue in $\mathbb{F}_{p^3}$. An element $\alpha \in \mathbb{F}_{p^6}$ is represented as $\alpha_0 + \alpha_1 Y$, where $\alpha_i \in \mathbb{F}_{p^3}$.

Multiplication and squaring can be computed with the same methods for quadratic extensions described in Section 3, replacing multiplication by β with multiplication by γ. Notice that $\beta \in \mathbb{F}_p$ and $\gamma \in \mathbb{F}_{p^3}$. In fact, γ is represented by $0 + 1X + 0X^2 = X$, and the product $b = \gamma a$, with $a = a_0 + a_1 Y$ and $a_i = a_{i,0} + a_{i,1}X + a_{i,2}X^2$, is computed as

$$\begin{aligned} b_i &= \gamma a_i \\ &= X(a_{i,0} + a_{i,1}X + a_{i,2}X^2) \\ &= \beta a_{i,2} + a_{i,0}X + a_{i,1}X^2, \end{aligned}$$

amounting to six permutations and two multiplications by β. The corresponding costs based on $\mathbb{F}_p$ operations depend on the choice of multiplication methods for the cubic extension field. Table 9 shows the multiplicative costs for $\mathbb{F}_{p^6}$ as a quadratic over cubic, and Table 10 shows the squaring costs for $\mathbb{F}_{p^6}$ as a quadratic over cubic.

6.2 Cubic over quadratic

We construct $\mathbb{F}_{p^2} = \mathbb{F}_p[X]/(X^2 - \beta)$ and $\mathbb{F}_{p^6} = \mathbb{F}_{p^2}[Y]/(Y^3 - \gamma)$, where β is a quadratic non-residue in $\mathbb{F}_p$ and $\gamma = \sqrt{\beta}$ is a cubic non-residue in $\mathbb{F}_{p^2}$. An element $\alpha \in \mathbb{F}_{p^6}$ is represented as $\alpha_0 + \alpha_1 Y + \alpha_2 Y^2$, where $\alpha_i \in \mathbb{F}_{p^2}$.

Table 9. Summary of multiplicative costs for sextic extensions as quadratic over cubic

$\mathbb{F}_{p^6}$ method	Schoolbook		Karatsuba	
$\mathbb{F}_{p^3}$ method	> Linear	Linear	> Linear	Linear
Schoolbook	$36M$	$30A + 9B$	$27M$	$33A + 7B$
Karatsuba	$24M$	$58A + 9B$	$18M$	$54A + 7B$
Toom-Cook-3x	$20M$	$138A + 9B$	$15M$	$114A + 7B$

Table 10. Summary of squaring costs for sextic extensions as quadratic over cubic

$\mathbb{F}_{p^6}$ method	Schoolbook		Karatsuba	
$\mathbb{F}_{p^3}$ method	> Linear	Linear	> Linear	Linear
Schoolbook	$15M + 6S$	$24A + 7B$	$9M + 9S$	$30A + 8B$
Karatsuba	$6M + 12S$	$45A + 7B$	$18S$	$51A + 8B$
Karatsuba/CH-SQR1	$12M + 4S$	$41A + 7B$	$9M + 6S$	$45A + 8B$
Karatsuba/CH-SQR2	$10M + 6S$	$39A + 7B$	$6M + 9S$	$42A + 8B$
Karatsuba/CH-SQR3	$8M + 8S$	$41A + 7B + 2D_2$	$3M + 12S$	$45A + 8B + 3D_2$
Karatsuba/CH-SQR3x	$8M + 8S$	$47A + 7B$	$3M + 12S$	$54A + 8B$
Toom-Cook-3x	$5M + 10S$	$105A + 7B$	$15S$	$111A + 8B$
	Complex			
	> Linear	Linear		
Schoolbook	$18M$	$24A + 6B$		
Karatsuba	$12M$	$38A + 6B$		
Toom-Cook-3x	$10M$	$78A + 6B$		

Multiplication and squaring can be computed with the same methods for cubic extensions described in Section 4, replacing multiplication by β with multiplication by γ. Notice that $\beta \in \mathbb{F}_p$ and $\gamma \in \mathbb{F}_{p^2}$. In fact, γ is represented by $0 + 1X = X$, and the product $b = \gamma a$, with $a = a_0 + a_1 Y + a_2 Y^2$ and $a_i = a_{i,0} + a_{i,1}X$, is computed as

$$\begin{aligned} b_i &= \gamma a_i \\ &= X(a_{i,0} + a_{i,1}X) \\ &= \beta a_{i,1} + a_{i,0}X, \end{aligned}$$

amounting to permutations and three multiplications by β.

The corresponding costs based on $\mathbb{F}_p$ operations depend on the choice of multiplication and squaring methods for the quadratic extension field. Table 11 shows the multiplicative costs for $\mathbb{F}_{p^6}$ as a cubic over quadratic, and Table 12 shows the squaring costs for $\mathbb{F}_{p^6}$ as a cubic over quadratic.

Table 11. Summary of multiplicative costs for sextic extensions as cubic over quadratic

$\mathbb{F}_{p^2}$ method	Schoolbook		Karatsuba	
$\mathbb{F}_{p^6}$ method	$>$ Linear	Linear	$>$ Linear	Linear
Schoolbook	$36M$	$30A + 11B$	$27M$	$57A + 11B$
Karatsuba	$24M$	$38A + 8B$	$18M$	$56A + 8B$
Toom-Cook-3x	$20M$	$76A + 7B$	$15M$	$91A + 7B$

Table 12. Summary of squaring costs for sextic extensions as cubic over quadratic

$\mathbb{F}_{p^2}$ method	Schoolbook		Karatsuba	
$\mathbb{F}_{p^6}$ method	$>$ Linear	Linear	$>$ Linear	Linear
Schoolbook	$15M + 6S$	$24A + 8B$	$9M + 9S$	$39A + 11B$
Karatsuba	$6M + 12S$	$38A + 8B$	$18S$	$50A + 14B$
Toom-Cook-3x	$5M + 10S$	$76A + 7B$	$15S$	$86A + 12B$
CH-SQR1	$14M + 4S$	$32A + 7B$	$9M + 6S$	$45A + 9B$
CH-SQR2	$11M + 6S$	$30A + 7B$	$6M + 9S$	$42A + 10B$
CH-SQR3	$8M + 8S$	$32A + 7B + 2D_2$	$3M + 12S$	$43A + 11B + 2D_2$
CH-SQR3x	$8M + 8S$	$38A + 7B$	$3M + 12S$	$49A + 11B$
	Karatsuba/Complex			
	$>$ Linear	Linear		
Schoolbook	$15M$	$39A + 11B$		
Karatsuba	$12M$	$50A + 14B$		
Toom-Cook-3x	$10M$	$86A + 12B$		
CH-SQR1	$13M$	$45A + 9B$		
CH-SQR2	$12M$	$42A + 10B$		
CH-SQR3	$11M$	$43A + 11B + 2D_2$		
CH-SQR3x	$11M$	$49A + 11B$		

6.3 Direct sextic

We construct $\mathbb{F}_{p^6}$ as $\mathbb{F}_p[X]/(X^6 - \beta)$, where β is both a quadratic and cubic non-residue in $\mathbb{F}_p$. An element $\alpha \in \mathbb{F}_{p^6}$ is represented as $\alpha_0 + \alpha_1 X + \alpha_2 X^2 + \alpha_3 X^3 + \alpha_4 X^4 + \alpha_5 X^5$, where $\alpha_i \in \mathbb{F}_p$.

The Schoolbook method computes the product $c = ab$ as

$$\begin{aligned}
c_0 &= a_0 b_0 + \beta(a_1 b_5 + a_2 b_4 + a_3 b_3 + a_4 b_2 + a_5 b_1) \\
c_1 &= a_0 b_1 + a_1 b_0 + \beta(a_2 b_5 + a_3 b_4 + a_4 b_3 + a_5 b_2) \\
c_2 &= a_0 b_2 + a_1 b_1 + a_2 b_0 + \beta(a_3 b_5 + a_4 b_4 + a_5 b_3) \\
c_3 &= a_0 b_3 + a_1 b_2 + a_2 b_1 + a_3 b_0 + \beta(a_4 b_5 + a_5 b_4) \\
c_4 &= a_0 b_4 + a_1 b_3 + a_2 b_2 + a_3 b_1 + a_4 b_0 + \beta a_5 b_5 \\
c_5 &= a_0 b_5 + a_1 b_4 + a_2 b_3 + a_3 b_2 + a_4 b_1 + a_5 b_0,
\end{aligned}$$

thus requiring $36M + 30A + 5B$, and the square $c = a^2$ as

$$\begin{aligned}
c_0 &= a_0^2 + \beta(2(a_1a_5 + a_2a_4) + a_3^2)\\
c_1 &= 2(a_0a_1 + \beta(a_2a_5 + a_3a_4))\\
c_2 &= 2a_0a_2 + a_1^2 + \beta(2a_3a_5 + a_4^2)\\
c_3 &= 2(a_0a_3 + a_1a_2 + \beta a_4a_5)\\
c_4 &= 2(a_0a_4 + a_1a_3) + a_2^2 + \beta a_5^2\\
c_5 &= 2(a_0a_5 + a_1a_4 + a_2a_3),
\end{aligned}$$

which costs $15M + 6S + 22A + 5B$.

Montgomery in [17] presented a formula for multiplying quintic polynomials that needs one less multiplication than Karatsuba. The formula depends on the choice of a polynomial C; we have arbitrarily chosen $C = X^6$. For the sake of space, the detailed formula is presented in the appendix. The total cost for multiplication is $17M + 143A + 5B$, and the total cost for squaring is $17S + 123A + 5B$.

We proceed to analyse the Toom-Cook method for sextic extensions. The x_i points for evaluation are $\{0, \pm1, \pm2, \pm3, \pm4, 5, \infty\}$. The first step is to precompute the interpolation points $v_i = a(P_i)b(P_i)$, $0 \leq i \leq 10$. Notice that, in absolute value, 5 is the biggest evaluation point. As a, b are quintic polynomials, the biggest coefficient is $5^5 = 3125$. If we use addition chains to compute the coefficients of $a(P_i), b(P_i)$ except for 81, 243, 625 and 3125, where we use multiplication by (small) integers, the cost of computing the interpolation points is $11M + 8M_{\mathbb{Z}} + 146A$ for multiplication, and $11S + 4M_{\mathbb{Z}} + 73A$ for squaring. The interpolation formula is listed in the appendix. As in the case of cubic extensions, we compute the product $c = 362880ab$ (or the square $c = 362880a^2$) in order to completely avoid divisions, and the formula is listed in the appendix. Because the vast majority of the coefficients in the formula are too big to be considered for addition chains, we compute them as multiplication by small integers. The cost for the interpolation is $75M_{\mathbb{Z}} + 90A + 5B$. The total cost for multiplication is $11M + 93M_{\mathbb{Z}} + 236A + 5B$, and the total cost for squaring is $11S + 79M_{\mathbb{Z}} + 163A + 5B$.

Tables 13 and 14 summarise the costs of multiplication and squaring in $\mathbb{F}_{p^6}$ as direct sextic extensions.

Table 13. Summary of multiplicative costs for direct sextic extensions

	$>$ Linear	Linear
Schoolbook	$36M$	$30A + 5B$
Montgomery	$17M$	$143A + 5B$
Toom-Cook-6x	$11M$	$93M_{\mathbb{Z}} + 236A + 5B$

Table 14. Summary of squaring costs for direct sextic extensions

	> Linear	Linear
Schoolbook	$15M + 6S$	$22A + 5B$
Montgomery	$17S$	$123A + 5B$
Toom-Cook-6x	$11S$	$79M_{\mathbb{Z}} + 163A + 5B$

6.4 Isomorphisms among the representations

Let $a \in \mathbb{F}_{p^6}$ be an element in the representation of the field as a quadratic over cubic, let $b \in \mathbb{F}_{p^6}$ be an element in the representation of the field as a cubic over quadratic, and let $c \in \mathbb{F}_{p^6}$ be an element in the representation of the field as a direct sextic extension. We can write a, b, c as

$$\begin{aligned} a &= (a_{0,0} + a_{0,1}\sqrt[3]{\beta} + a_{0,2}\sqrt[3]{\beta^2}) + (a_{1,0} + a_{1,1}\sqrt[3]{\beta} + a_{1,2}\sqrt[3]{\beta^2})\sqrt[6]{\beta} \\ &= a_{0,0} + a_{0,1}\sqrt[3]{\beta} + a_{0,2}\sqrt[3]{\beta^2} + a_{1,0}\sqrt[6]{\beta} + a_{1,1}\sqrt{\beta} + a_{1,2}\sqrt[6]{\beta^5}, \\ b &= (b_{0,0} + b_{0,1}\sqrt{\beta}) + (b_{1,0} + b_{1,1}\sqrt{\beta})\sqrt[6]{\beta} + (b_{2,0} + b_{2,1}\sqrt{\beta})\sqrt[3]{\beta} \\ &= b_{0,0} + b_{0,1}\sqrt{\beta} + b_{1,0}\sqrt[6]{\beta} + b_{1,1}\sqrt[3]{\beta^2} + b_{2,0}\sqrt[3]{\beta} + b_{2,1}\sqrt[6]{\beta^5}, \\ c &= c_0 + c_1\sqrt[6]{\beta} + c_2\sqrt[3]{\beta} + c_3\sqrt{\beta} + c_4\sqrt[3]{\beta^2} + c_5\sqrt[6]{\beta^5}, \end{aligned}$$

so the isomorphisms among the representations are efficiently computed by the following permutations

$$\begin{bmatrix} a_{0,0} & a_{0,1} & a_{0,2} & a_{1,0} & a_{1,1} & a_{1,2} \\ b_{0,0} & b_{2,0} & b_{1,1} & b_{1,0} & b_{0,1} & b_{2,1} \\ c_0 & c_2 & c_4 & c_1 & c_3 & c_5 \end{bmatrix}.$$

7 Timings

The performance of multiplication and squaring was measured on a Pentium IV 2.8 GHz with 512 MB of RAM and a 512 KB cache, running GNU/Linux 2.6.15. We have used the MIRACL library for big number and modular arithmetic. Programs were coded in C and compiled with the GNU C Compiler version 4.1.2 (prerelease). Standard compiler (`-O2`) and IA-32/Intel SSE2 (`-msse2`) optimisations were used.

The MIRACL library implements multi-precision number arithmetic, and supports a number of powerful optional optimizations. In particular it supports completely unrolled assembly language support for fixed-size big number multiplication and modular reduction. Internally, prime field elements are in Montgomery representation [16], which allows for fast reduction without divisions. The memory for big numbers can be allocated from the heap, or more efficiently from the stack, which is the case for our experiments. When required to multiply big numbers by a small integer, multiplications by numbers less than or equal to 6 are instead carried out by up to 3 modular additions.

The pairing-friendly fields were constructed as follows. We generated five random primes p_i where $\beta = -2$ is both a quadratic and cubic non-residue in $\mathbb{F}_{p_i}$, and $\lceil \log_2 p_i \rceil \in \{160, 192, 224, 256, 512\}$.

We constructed the extensions with degree $k \in \{2, 3, 4, 6\}$ as:

$$\begin{aligned}
\mathbb{F}_{p^2} &= \mathbb{F}_p[X]/(X^2+2) \text{ (quadratic)}\\
\mathbb{F}_{p^3} &= \mathbb{F}_p[X]/(X^3+2) \text{ (cubic)}\\
\mathbb{F}_{p^4} &= \mathbb{F}_{p^2}[Y]/(Y^2-\sqrt{-2}) \text{ (quadratic over quadratic)}\\
\mathbb{F}_{p^4} &= \mathbb{F}_p[X]/(X^4+2) \text{ (direct quartic)}\\
\mathbb{F}_{p^6} &= \mathbb{F}_{p^3}[Y]/(Y^2-\sqrt[3]{-2}) \text{ (quadratic over cubic)}\\
\mathbb{F}_{p^6} &= \mathbb{F}_{p^2}[Y]/(Y^3-\sqrt{-2}) \text{ (cubic over quadratic)}\\
\mathbb{F}_{p^6} &= \mathbb{F}_p[X]/(X^6+2) \text{ (direct sextic).}
\end{aligned}$$

We implemented all the methods described in this paper; for extension towers, we implemented all the possible combinations. Methods that use divisions, such as CH-SQR3 and Toom-Cook-3, were implemented without divisions; instead, we multiplied the result by the least common multiple of the denominators as described in Section 2.

The timing method for multiplication starts by generating two random field elements $a, b \in \mathbb{F}_q$ and then computes $a \leftarrow ab$ repeatedly. For squaring, the method generates random $a \in \mathbb{F}_q$ and then computes $a \leftarrow a^2$ repeatedly. Each timing program was run several times in order to ensure randomness.

Timings are presented in the Appendix. Tables 15, 16, 17, 18, 19, 21, 22, 23 show the absolute timings for each method and field. The fastest method is highlighted with a star symbol, and other methods that are not more than 10% slower than the fastest method are highlighted with an equal sign. Tables 20, 24 summarise the methods that have the best performance for $k = 4, 6$ among the different construction possibilities.

8 Conclusion

We analysed different multiplication and squaring methods for finite extension fields $\mathbb{F}_{p^k}$ with $k \in \{2, 3, 4, 6\}$, detailing the cost of each method with regard to operations in $\mathbb{F}_p$. We also timed every method using the MIRACL library and primes p with bitlengths 160, 192, 224, 256, and 512.

Our first observation is a comparison about the time to compute a product and the time to compute a square. If we consider only the super-linear operations on the ground field, squaring costs $1M$ less than multiplication in $\mathbb{F}_{p^2}$, $3M$ less in $\mathbb{F}_{p^4}$, and $5M$ less in $\mathbb{F}_{p^6}$. Our experimental results show that multiplication in these fields is about 40% slower than squaring. In Miller's algorithm to compute the Tate pairing, the accumulating variable that is defined over $\mathbb{F}_{p^k}$ is normally squared each iteration of the loop, leading to a large amount of squarings in total. It is therefore important, when analysing the cost of pairing computation, to make a clear distinction between multiplication and squaring, a distinction that has been neglected in some analyses [7, 8, 10, 14].

A second observation is that, for composite k, constructing $\mathbb{F}_{p^k}$ as a high tower of extensions seems to yield the most efficient implementations for multiplication and squaring. There are very efficient formulæ for quadratic and cubic extensions, but not for higher degrees. Our best multiplication and squaring timings for $\mathbb{F}_{p^6}$ were obtained in the quadratic over cubic construction.

Another observation is that Toom-Cook is the method that asymptotically requires the least number of multiplication or squarings in the underlying field. However, both the computation of the evaluation points and the interpolation incur a number of additions and multiplications by word-size integers, as well as divisions, that result in an inefficient method. Toom-Cook-3 is an exception: its formula is simple and requires only one division by 3. Montgomery has presented formulæ that use less multiplication operations than a recursive application of Karatsuba, but the overhead of his method when applied to finite extension fields make it less efficient.

We also point out the recent development of efficient formulæ for squaring by Chung and Hasan [5], Montgomery's work on improving the number of multiplications required by Karatsuba [17], and the investigation of Bodrato and Zanoni on efficient ways to compute Toom-Cook [3]. They have used computer-assisted techniques to improve standard multiplication and squaring formulæ, and it is quite possible that there will be further improvements on methods for multiplication and squaring in the future.

Our final observation is a set of recommendations on the most efficient methods for each extension degree. For quadratic extensions, use Karatsuba for multiplication and Complex for squaring. For cubic extensions, if $M < 26A$ then use Karatsuba for multiplication; otherwise, use Toom-Cook-3x. For squaring, either Chung-Hasan SQR1 or SQR2. For quartic extensions, construct the extension as quadratic over quadratic, and use Karatsuba over Karatsuba for multiplication and Complex over Karatsuba for squaring. For sextic extensions, if $M < 20A$, construct the extension as quadratic over cubic, use Karatsuba over Karatsuba for multiplication, and use Complex over Karatsuba for squaring. If $M \geq 20A$, construct the extension as cubic over quadratic, use Toom-Cook-3x over Karatsuba for multiplication and either Complex, Chung-Hasan SQR3 or SQR3x over Karatsuba/Complex for squaring. Whenever it is necessary to map between different finite extension field constructions, there are very efficient isomorphisms available.

References

1. P. S. L. M. Barreto, B. Lynn, and M. Scott. Efficient Implementation of Pairing-Based Cryptosystems. *Journal of Cryptology*, 17(4):321–334, 2004.
2. Daniel J. Bernstein. Multidigit Multiplication for Mathematicians, 2001. Available from `http://cr.yp.to/arith.html#m3`.

3. Marco Bodrato and Alberto Zanoni. What About Toom-Cook Matrices Optimality? Technical Report 605, Centro Interdipartimentale Vito Volterra - Università di Roma "Tor Vergata", 2006. Available from `http://bodrato.it/papers/#CIVV2006`.
4. D. Boneh and M. Franklin. Identity-Based Encryption from the Weil Pairing. *SIAM Journal of Computing*, 32(3):586–615, 2003.
5. Jaewook Chung and M. Anwar Hasan. Asymmetric squaring formulae. Technical Report CACR 2006-24, CACR, Univ. of Waterloo, 2006. `http://www.cacr.math.uwaterloo.ca/techreports/2006/cacr2006-24.pdf`.
6. S. A. Cook. On the Minimum Computation Time of Functions. PhD Thesis, Harvard University Department of Mathematics, 1966.
7. R. Granger, D. Page, and N. P. Smart. High Security Pairing-Based Cryptography Revisited. In *Algorithmic Number Theory Symposium – ANTS VII*, volume 4076 of *Lecture Notes in Computer Science*, pages 480–494. Springer-Verlag, 2006.
8. R. Granger and N. P. Smart. On Computing Products of Pairings. Cryptology ePrint Archive, Report 2006/172, 2006. `http://eprint.iacr.org/2006/172`.
9. Darrel Hankerson, Alfred Menezes, and Scott Vanstone. *Guide to Elliptic Curve Cryptography.* Springer, New York, 2004.
10. F. Hess, N. P. Smart, and F. Vercauteren. The Eta Pairing Revisited. *IEEE Transactions on Information Theory*, 52(10):4595–4602, 2006.
11. A. Joux. A One-Round Protocol for Tripartite Diffie-Hellman. In *Algorithmic Number Theory Symposium – ANTS IV*, volume 1838 of *Lecture Notes in Computer Science*, pages 385–394. Springer-Verlag, 2000.
12. A. A. Karatsuba and Y. Ofman. Multiplication of Multidigit Numbers on Automata. *Soviet Physics Doklady*, 7:595–596, 1963.
13. D. E. Knuth. *The Art of Computer Programming, volume 2, Seminumerical Algorithms, 3rd edition.* Addision-Wesley, 1998.
14. N. Koblitz and A. Menezes. Pairing-Based Cryptography at High Security Levels. In *Cryptography and Coding – IMA 2005*, volume 3796 of *Lecture Notes in Computer Science*, pages 13–36. Springer-Verlag, 2005.
15. Chae Hoon Lim and Hyo Sun Hwang. Fast Implementation of Elliptic Curve Arithmetic in $GF(p^n)$. In *PKC 2000*, number 1751 in LNCS, pages 405–421, Berlin Heidelberg, 2000. Springer-Verlag.
16. Peter L. Montgomery. Modular Multiplication Without Trial Division. *Mathematics of Computation*, 44(170):519–521, April 1985.
17. Peter L. Montgomery. Five, six, and seven-term Karatsuba-like formulae. *IEEE Transactions on Computers*, 54(3):362–369, 2005.
18. Michael Scott. Implementing Cryptographic Pairings. preprint, 2006. `ftp://ftp.computing.dcu.ie/pub/resources/crypto/pairings.pdf`.
19. A. L. Toom. The Complexity of a Scheme of Functional Elements realizing the Multiplication of Integers. *Soviet Mathematics*, 4(3):714–716, 1963.
20. André Weimerskirch and Christof Paar. Generalizations of the Karatsuba Algorithm for Efficient Implementations. Cryptology ePrint Archive, Report 2006/224, 2006. Available from `http://eprint.iacr.org/`.

A Appendix

A.1 Montgomery-6

Montgomery in [17] presented a formula for multiplying quintic polynomials that needs one less multiplication than Karatsuba. The formula depends on the choice of a polynomial C; we have arbitrarily chosen $C = X^6$. We start by computing the 17 ground field products as follows (note that v_1 is omitted because of our choice of C):

$$\begin{aligned}
v_0 &= (a_0 + a_1 + a_2 + a_3 + a_4 + a_5)(b_0 + b_1 + b_2 + b_3 + b_4 + b_5)\\
v_2 &= (a_0 + a_1 + a_3 + a_4)(b_0 + b_1 + b_3 + b_4)\\
v_3 &= (a_0 - a_2 - a_3 + a_5)(b_0 - b_2 - b_3 + b_5)\\
v_4 &= (a_0 - a_2 - a_5)(b_0 - b_2 - b_5)\\
v_5 &= (a_0 + a_3 - a_5)(b_0 + b_3 - b_5)\\
v_6 &= (a_0 + a_1 + a_2)(b_0 + b_1 + b_2)\\
v_7 &= (a_3 + a_4 + a_5)(b_3 + b_4 + b_5)\\
v_8 &= (a_2 + a_3)(b_2 + b_3)\\
v_9 &= (a_1 - a_4)(b_1 - b_4)\\
v_{10} &= (a_1 + a_2)(b_1 + b_2)\\
v_{11} &= (a_3 + a_4)(b_3 + b_4)\\
v_{12} &= (a_0 + a_1)(b_0 + b_1)\\
v_{13} &= (a_4 + a_5)(b_4 + b_5)\\
v_{14} &= a_0 b_0\\
v_{15} &= a_1 b_1\\
v_{16} &= a_4 b_4\\
v_{17} &= a_5 b_5,
\end{aligned}$$

which costs $17M + 40A$. To compute the square $c = a^2$, we compute v_i as

$$\begin{aligned}
v_0 &= (a_0 + a_1 + a_2 + a_3 + a_4 + a_5)^2 \\
v_2 &= (a_0 + a_1 + a_3 + a_4)^2 \\
v_3 &= (a_0 - a_2 - a_3 + a_5)^2 \\
v_4 &= (a_0 - a_2 - a_5)^2 \\
v_5 &= (a_0 + a_3 - a_5)^2 \\
v_6 &= (a_0 + a_1 + a_2)^2 \\
v_7 &= (a_3 + a_4 + a_5)^2 \\
v_8 &= (a_2 + a_3)^2 \\
v_9 &= (a_1 - a_4)^2 \\
v_{10} &= (a_1 + a_2)^2 \\
v_{11} &= (a_3 + a_4)^2 \\
v_{12} &= (a_0 + a_1)^2 \\
v_{13} &= (a_4 + a_5)^2 \\
v_{14} &= a_0^2 \\
v_{15} &= a_1^2 \\
v_{16} &= a_4^2 \\
v_{17} &= a_5^2,
\end{aligned}$$

which costs $17S + 20A$. The product (or squaring) is computed as

$$\begin{aligned}
c_0 &= v_{14} + \beta(v_0 - v_2 + 2v_3 + v_4 + 2v_5 + 2v_6 + 3v_7 - 3v_8 - 3v_{10} - 3v_{11} - 2v_{12} \\
&\quad - 4v_{13} - 5v_{14} + 3v_{15} + 4v_{16} - 5v_{17}) \\
c_1 &= v_{12} - v_{14} - v_{15} + \beta(-v_3 - v_5 - v_6 - 2v_7 + v_8 + v_{10} + 3v_{11} + v_{12} + 2v_{13} \\
&\quad + 2v_{14} - v_{15} - 3v_{16} + 2v_{17}) \\
c_2 &= v_6 - v_{10} - v_{12} + 2v_{15} + \beta(v_7 - v_{11} - v_{13} + 2v_{16}) \\
c_3 &= -v_3 - v_4 - 2v_6 - v_7 + v_8 + 3v_{10} + v_{11} + 2v_{12} + v_{13} + 2v_{14} - 3v_{15} - v_{16} \\
&\quad + 2v_{17} + \beta(v_{13} - v_{16} - v_{17}) \\
c_4 &= v_2 + v_3 + v_4 + 2v_6 + v_7 - v_8 + v_9 - 2v_{10} - 2v_{11} - 3v_{12} - v_{13} - 2v_{14} + v_{15} \\
&\quad - 2v_{17} + \beta v_{17} \\
c_5 &= -v_3 - v_4 - v_5 - 2v_6 - 2v_7 + 2v_8 - v_9 + 2v_{10} + 2v_{11} + 2v_{12} + 2v_{13} + 3v_{14} \\
&\quad - v_{15} - v_{16} + 3v_{17},
\end{aligned}$$

which requires $103A+5B$. Then the total cost for multiplication is $17M+143A+5B$, and the total cost for squaring is $17S + 123A + 5B$.

A.2 Toom-Cook-6x

We proceed to analyse the Toom-Cook method for sextic extensions. The x_i points for evaluation are $\{0, \pm1, \pm2, \pm3, \pm4, 5, \infty\}$. The first step is to precompute the interpolation points $v_i = a(P_i)b(P_i)$, $0 \leq i \leq 10$. Notice that, in absolute value, 5 is the biggest evaluation point. As a, b are quintic polynomials, the biggest coefficient is $5^5 = 3125$. If we use addition chains to compute the coefficients of $a(P_i), b(P_i)$ except for 81, 243, 625 and 3125, where we use multiplication by (small) integers, the cost of computing the interpolation points is $11M + 8M_{\mathbb{Z}} + 146A$ for multiplication, and $11S + 4M_{\mathbb{Z}} + 73A$ for squaring. The interpolation is computed as

$$\begin{aligned}
c_0 &= v_0 + \beta(-(5/96)v_0 + (29/720)(v_1 + v_2) - (13/720)(v_3 + v_4) \\
&\quad + (1/240)(v_5 + v_6) - (1/2880)(v_7 + v_8) + 273v_{10}) \\
c_1 &= -(1/5)v_0 + v_1 - (2/3)v_2 - (1/3)v_3 + (1/7)v_4 + (2/21)v_5 - (1/42)v_6 \\
&\quad - (1/56)v_7 + (1/504)v_8 + (1/630)v_9 - 2880v_{10} + \beta((1/96)v_0 - (17/1440)v_1 \\
&\quad - (1/180)v_2 + (1/120)v_3 + (1/630)v_4 - (1/280)v_5 - (1/6720)v_6 \\
&\quad + (17/20160)v_7 - (1/60480)v_8 - (1/12096)v_9 + 150v_{10}) \\
c_2 &= -(205/144)v_0 + (4/5)(v_1 + v_2) - (1/10)(v_3 + v_4) + (4/315)(v_5 + v_6) \\
&\quad - (1/1120)(v_7 + v_8) + 576v_{10} + \beta((1/576)v_0 - (1/720)(v_1 + v_2) \\
&\quad + (1/1440)(v_3 + v_4) - (1/5040)(v_5 + v_6) + (1/40320)(v_7 + v_8) - 30v_{10}) \\
c_3 &= (41/144)v_0 - (449/720)v_1 + (161/1080)v_2 + (917/2160)v_3 \\
&\quad - (773/5040)v_4 - (331/2520)v_5 + (299/10080)v_6 + (127/5040)v_7 \\
&\quad - (59/22680)v_8 - (41/18144)v_9 + 4100v_{10} + \beta(-(1/2880)(v_0 + v_1) \\
&\quad + (1/4320)(v_2 - v_3) - (1/10080)(v_4 + v_5) + (1/40320)(v_6 - v_7) \\
&\quad - (1/362880)(v_8 + v_9) - 5v_{10}) \\
c_4 &= (91/192)v_0 - (61/180)(v_1 + v_2) + (169/1440)(v_3 + v_4) - (1/60)(v_5 + v_6) \\
&\quad + (7/5760)(v_7 + v_8) - 820v_{10} + \beta v_{10} \\
c_5 &= -(91/960)v_0 + (389/2880)v_1 + (1/480)(11v_2 + 19v_5) \\
&\quad - (1/1440)(143v_3 + 13v_4) - (11/1920)v_6 - (47/5760)v_7 \\
&\quad + (1/17280)(11v_8 + 13v_9) - 1365v_{10}.
\end{aligned}$$

We compute the product $c = 362880ab$ (or the square $c = 362880a^2$ in order to completely avoid divisions as follows:

$$\begin{aligned}
c_0 = {} & 362880v_0 + \beta(-18900v_0 + 14616(v_1 + v_2) - 6552(v_3 + v_4) + 1512(v_5 + v_6) \\
& - 126(v_7 + v_8) + 99066240v_{10}) \\
c_1 = {} & -72576v_0 + 362880v_1 - 241920v_2 - 120960v_3 + 51840v_4 + 34560v_5 \\
& - 8640v_6 - 6480v_7 + 720v_8 + 576v_9 - 1045094400v_{10} + \beta(3780v_0 - 4284v_1 \\
& - 2016v_2 + 3024v_3 + 576v_4 - 1296v_5 - 54v_6 + 306v_7 - 6v_8 - 30v_9 \\
& + 54432000v_{10}) \\
c_2 = {} & -516600v_0 + 290304(v_1 + v_2) - 36288(v_3 + v_4) + 4608(v_5 + v_6) \\
& - 324(v_7 + v_8) + 209018880v_{10} + \beta(630v_0 - 504(v_1 + v_2) + 252(v_3 + v_4) \\
& - 72(v_5 + v_6) + 9(v_7 + v_8) - 10886400v_{10}) \\
c_3 = {} & 103320v_0 - 226296v_1 + 54096v_2 + 154056v_3 - 55656v_4 - 47664v_5 \\
& + 10764v_6 + 9144v_7 - 944v_8 - 820v_9 + 1487808000v_{10} + \beta(-126(v_0 + v_1) \\
& + 84(v_2 - v_3) - 36(v_4 + v_5) + 9(v_6 - v_7) - (v_8 + v_9) - 1814400v_{10}) \\
c_4 = {} & 171990v_0 - 122976(v_1 + v_2) + 42588(v_3 + v_4) - 6048(v_5 + v_6) \\
& + 63(v_7 + v_8) - 297561600v_{10} + \beta(362880v_{10}) \\
c_5 = {} & -34398v_0 + 49014v_1 + 8316v_2 + 14364v_5 - 36036v_3 + 3276v_4 - 2079v_6 \\
& - 2961v_7 + 231v_8 + 273v_9 - 495331200v_{10}.
\end{aligned}$$

Because the vast majority of the coefficients in the formula are too large to be considered for addition chains, we compute them as multiplication by small integers. The cost for the interpolation is $75M_{\mathbb{Z}} + 90A + 5B$. The total cost for multiplication is $11M + 93M_{\mathbb{Z}} + 236A + 5B$, and the total cost for squaring is $11S + 79M_{\mathbb{Z}} + 163A + 5B$.

A.3 Timings

Table 15. Timings of $\mathbb{F}_p$ operations with Montgomery representation (microseconds)

	160-bit	192-bit	224-bit	256-bit	512-bit
Addition	0.04	0.04	0.04	0.05	0.08
Subtraction	0.04	0.04	0.05	0.05	0.09
Negation	0.03	0.03	0.03	0.04	0.06
Multiplication	0.29	0.37	0.42	0.54	1.41
Squaring	0.29	0.37	0.42	0.53	1.41
MUL/ADD ratio	7.81	8.90	9.36	10.92	16.73

Table 16. Timings of multiplication and squaring for $\mathbb{F}_{p^2}$ (microseconds)

	160-bit	192-bit	224-bit	256-bit	512-bit
Multiplication					
Schoolbook-2	1.33	1.64	1.87	2.32	5.86
Karatsuba-2	* 1.17	* 1.40	* 1.59	* 1.93	* 4.74
Squaring					
Schoolbook-2	1.10	1.25	1.46	1.74	4.43
Karatsuba-2	1.16	1.31	1.52	1.81	4.58
Complex	* 0.84	* 1.00	* 1.13	* 1.37	* 3.26
MUL/SQR ratio	1.39	1.40	1.41	1.41	1.45

Table 17. Timings of multiplication and squaring for $\mathbb{F}_{p^3}$ (microseconds)

	160-bit	192-bit	224-bit	256-bit	512-bit
Multiplication					
Schoolbook-3	3.01	3.66	4.26	5.53	13.45
Karatsuba-3	* 2.48	* 2.98	* 3.44	* 4.23	* 10.03
Toom-Cook-3x	2.99	3.47	3.95	4.72	= 10.28
Squaring					
Schoolbook-3	3.03	3.67	4.27	5.40	13.38
Karatsuba-3	2.48	2.83	3.25	3.86	9.53
CH-SQR1	* 2.02	* 2.38	* 2.75	= 3.33	= 8.15
CH-SQR2	= 2.05	= 2.39	= 2.76	* 3.31	* 8.13
CH-SQR3	2.30	2.66	= 3.02	= 3.61	= 8.69
CH-SQR3x	2.27	= 2.62	= 3.01	= 3.58	= 8.54
Toom-Cook-3x	2.80	3.14	3.58	4.22	9.49
MUL/SQR ratio	1.23	1.25	1.25	1.28	1.23

Table 18. Timings of multiplication and squaring for $\mathbb{F}_{p^4}$ as a quadratic over quadratic (microseconds). KA = Karatsuba

	160-bit	192-bit	224-bit	256-bit	512-bit
Multiplication					
Schoolbook over Schoolbook	5.67	7.04	7.94	9.80	24.31
Schoolbook over Karatsuba	5.19	6.09	6.88	8.28	19.75
Karatsuba over Schoolbook	= 4.88	5.87	6.52	7.97	19.28
Karatsuba over Karatsuba	* 4.47	* 5.17	* 5.75	* 6.90	* 15.90
Squaring					
Karatsuba over KA/Complex	= 3.15	* 3.60	* 4.03	* 4.82	= 10.96
Karatsuba over Karatsuba	4.08	4.55	5.24	6.24	14.90
Karatsuba over Schoolbook	3.86	4.37	5.01	5.97	14.50
Schoolbook over KA/Complex	= 3.32	= 3.81	= 4.33	= 5.22	= 12.02
Schoolbook over Karatsuba	3.97	4.46	5.15	6.18	14.80
Schoolbook over Schoolbook	3.94	4.57	5.24	6.33	15.56
Complex over Karatsuba	* 3.14	= 3.64	= 4.04	= 4.84	* 10.92
Complex over Schoolbook	= 3.44	4.10	4.59	5.57	13.15
MUL/SQR ratio	1.42	1.44	1.43	1.43	1.46

Table 19. Timings of multiplication and squaring for $\mathbb{F}_{p^4}$ as a direct quartic extension (microseconds)

	160-bit	192-bit	224-bit	256-bit	512-bit
Multiplication					
Schoolbook-4	* 5.40	* 6.52	* 7.58	* 9.23	* 23.84
Toom-Cook-4x	12.88	19.60	21.54	16.39	42.84
Squaring					
Schoolbook-4	* 3.62	* 4.31	* 4.99	* 6.02	* 15.18
Toom-Cook-4x	11.23	15.64	17.03	13.57	33.53
MUL/SQR ratio	1.49	1.51	1.52	1.53	1.57

Table 20. Best performance of multiplication and squaring for $\mathbb{F}_{p^4}$, all constructions (microseconds)

	Multiplication		Squaring		MUL/SQR
160-bit	KA-2 over KA-2	4.47	Complex over KA-2	3.14	1.42
192-bit	KA-2 over KA-2	5.17	KA-2 over KA-2/Complex	3.60	1.44
224-bit	KA-2 over KA-2	5.75	KA-2 over KA-2/Complex	4.03	1.43
256-bit	KA-2 over KA-2	6.90	KA-2 over KA-2/Complex	4.82	1.43
512-bit	KA-2 over KA-2	15.90	Complex over KA-2	10.92	1.46

Table 21. Timings of multiplication and squaring for $\mathbb{F}_{p^6}$ as a quadratic over cubic (microseconds)

	160-bit	192-bit	224-bit	256-bit	512-bit
Multiplication					
Schoolbook-2 over Schoolbook-3	12.64	15.42	17.97	22.88	54.95
Schoolbook-2 over Karatsuba-3	10.74	12.72	14.53	18.08	41.47
Schoolbook-2 over Toom-Cook-3x	12.74	14.74	16.69	19.95	42.51
Karatsuba-2 over Schoolbook-3	10.08	12.10	14.03	17.86	42.54
Karatsuba-2 over Karatsuba-3	* 8.58	* 10.17	* 11.55	* 14.44	* 32.11
Karatsuba-2 over Toom-Cook-3x	10.17	11.68	13.24	= 15.73	= 32.99
Squaring					
Complex-2 over Schoolbook-3	7.18	8.55	9.94	12.59	29.19
Complex-2 over Karatsuba-3	= 6.21	= 7.25	= 8.28	= 10.21	= 22.44
Complex-2 over Toom-Cook-3x	7.30	8.35	9.41	= 11.19	= 22.92
Karatsuba-2 over Karatsuba-3	= 6.19	= 7.26	= 8.31	* 10.20	= 22.45
Karatsuba-2 over Karatsuba-3/CH-SQR1	= 6.23	= 7.26	= 8.25	= 10.23	= 22.38
Karatsuba-2 over Karatsuba-3/CH-SQR2	= 6.25	* 7.24	= 8.26	= 10.28	= 22.46
Karatsuba-2 over Karatsuba-3/CH-SQR3	= 6.22	= 7.25	* 8.24	= 10.42	= 22.42
Karatsuba-2 over Karatsuba-3/CH-SQR3x	* 6.19	= 7.26	= 8.31	* 10.20	* 22.36
Schoolbook-2 over Schoolbook-3	9.81	11.80	13.72	17.09	41.48
Schoolbook-2 over Karatsuba-3	8.22	9.43	10.79	13.14	30.76
Schoolbook-2 over Karatsuba-3/CH-SQR1	7.30	8.59	9.82	12.13	27.97
Schoolbook-2 over Karatsuba-3/CH-SQR2	7.31	8.58	9.87	12.11	27.84
Schoolbook-2 over Karatsuba-3/CH-SQR3	7.93	9.18	10.42	12.78	29.57
Schoolbook-2 over Karatsuba-3/CH-SQR3x	7.88	9.07	10.35	12.62	28.91
Schoolbook-2 over Toom-Cook-3x	9.35	10.65	12.04	14.12	30.93
MUL/SQR ratio	1.39	1.40	1.40	1.42	1.44

Table 22. Timings of multiplication and squaring for $\mathbb{F}_{p^6}$ as a cubic over quadratic (microseconds)

	160-bit	192-bit	224-bit	256-bit	512-bit
Multiplication					
Schoolbook-3 over Karatsuba-2	11.70	13.84	15.85	19.17	45.47
Schoolbook-3 over Schoolbook-2	13.16	15.84	18.36	22.63	55.82
Karatsuba-3 over Karatsuba-2	* 9.24	* 10.68	* 12.20	* 14.49	* 33.06
Karatsuba-3 over Schoolbook-2	= 10.15	12.04	13.79	16.71	39.90
Toom-Cook-3x over Karatsuba-2	10.74	12.22	13.57	16.09	= 33.30
Toom-Cook-3x over Schoolbook-2	11.40	13.32	14.95	17.99	38.95
Squaring					
Schoolbook-3 over Karatsuba-2/Complex	= 7.02	= 8.22	9.47	11.48	26.17
Schoolbook-3 over Karatsuba-2	8.07	9.32	10.67	12.88	30.15
Schoolbook-3 over Schoolbook-2	8.23	9.68	11.24	13.74	33.17
Karatsuba-3 over Karatsuba-2/Complex	= 6.89	= 7.79	= 8.91	= 10.66	= 23.17
Karatsuba-3 over Karatsuba-2	8.93	9.94	11.40	13.34	30.89
Karatsuba-3 over Schoolbook-2	8.20	9.46	10.89	12.87	30.20
Toom-Cook-3x over Karatsuba-2/Complex	8.31	9.20	10.38	12.25	24.32
Toom-Cook-3x over Karatsuba-2	9.98	11.01	12.40	14.26	30.55
Toom-Cook-3x over Schoolbook-2	9.33	10.60	11.94	13.98	30.42
CH-SQR1 over Karatsuba-2/Complex	= 6.90	= 7.90	= 8.99	= 10.80	= 24.08
CH-SQR1 over Karatsuba-2	7.50	8.60	9.87	11.83	26.80
CH-SQR1 over Schoolbook-2	7.78	9.10	10.50	13.22	29.92
CH-SQR2 over Karatsuba-2/Complex	* 6.51	* 7.49	* 8.49	* 10.22	= 22.51
CH-SQR2 over Karatsuba-2	7.56	8.56	9.74	11.56	26.53
CH-SQR2 over Schoolbook-2	7.50	8.70	9.99	12.04	28.53
CH-SQR3 over Karatsuba-2/Complex	= 6.81	= 7.62	= 8.70	= 10.40	= 22.09
CH-SQR3 over Karatsuba-2	8.08	9.10	10.32	12.12	27.85
CH-SQR3 over Schoolbook-2	7.89	8.93	10.22	12.28	28.80
CH-SQR3x over Karatsuba-2/Complex	= 6.76	= 7.60	= 8.62	= 10.39	* 22.05
CH-SQR3x over Karatsuba-2	8.07	9.05	10.30	12.07	27.56
CH-SQR3x over Schoolbook-2	7.83	8.86	10.20	12.35	28.22
MUL/SQR ratio	1.42	1.42	1.44	1.42	1.50

Table 23. Timings of multiplication and squaring for $\mathbb{F}_{p^6}$ as a direct sextic extension (microseconds)

	160-bit	192-bit	224-bit	256-bit	512-bit
Multiplication					
Schoolbook-6	= 12.52	= 15.09	17.40	21.81	54.01
Montgomery-6	* 12.04	* 13.87	* 15.68	* 19.09	* 39.26
Toom-Cook-6x	39.34	75.47	82.81	51.59	156.39
Squaring					
Schoolbook-6	* 7.72	* 9.14	* 10.54	* 13.17	* 32.10
Montgomery-6	11.08	12.45	14.17	16.50	35.87
Toom-Cook-6x	35.45	67.83	74.34	45.23	144.22
MUL/SQR ratio	1.56	1.52	1.49	1.45	1.22

Table 24. Best performance of multiplication and squaring for $\mathbb{F}_{p^6}$, all constructions (microseconds)

	Multiplication		Squaring		MUL/SQR
160-bit	KA-2 over KA-3	8.58	KA-2 over KA-3/CH-SQR3x	6.19	1.39
192-bit	KA-2 over KA-3	10.17	KA-2 over KA-3/CH-SQR2	7.24	1.40
224-bit	KA-2 over KA-3	11.55	KA-2 over KA-3/CH-SQR3	8.24	1.40
256-bit	KA-2 over KA-3	14.44	KA-2 over KA-3/CH-SQR3x	10.20	1.42
512-bit	KA-2 over KA-3	32.11	CH-SQR3x over KA-2/Complex	22.05	1.46