# Choosing and generating parameters for low level pairing implementation on BN curves

S. Duquesne, N. El Mrabet, S. Haloui, F. Rondepierre

December 2, 2015

## Reste à faire

1. décider si pour changer de paragraphe on fait plutot sauts de ligne (et donc une indentation pour le nouveau paragraphe) ou des quadruples slash (et donc pas d'indentation). Puis uniformiser le document sur ce point car ca ne l'est pas du tout actuellement. Perso, je préfère éviter les indentations.

2. dans la biblio, il y a des prenom nom et des nom, prenom. A voir en fonctions du journal de soumission

## 1 Introduction

Pairing based cryptography has now many practical applications such as short signature schemes [12], identity based cryptography [10] or broadcast encryption [11]. Because of recent attacks on the discrete logarithm problem in small characteristic finite fields [30, 4], it is now clear that prime base fields should be used to define pairing friendly elliptic curves. Many implementations, both hardware and software can be found in the literature and some pairing friendly parameters are given. For example, at the 128-bit security level, the parameters given in [44] are almost always used because they have many good properties, in particular in terms of efficiency. However, depending on the situation, it could be useful to generate other nice parameters (e.g. resistance to subgroup attacks [38, 5], larger (or smaller) security levels, database of pairing friendly curves). This paper deals only with the case of BN curves because they are undoubtedly the best choice at the 128-bit security level, which is the most used today and in the near future, but also because they have many nice properties (e.g. maximal degree twists) that make them interesting even for higher security levels.

The main purpose of this paper is to describe explicitly and exhaustively what should be done to generate the best possible parameters and to make the best choices depending on the implementation context. We focus on low level implementation (mainly hardware but also assembly language), assuming that $\mathbb{F}_p$

additions have a significant cost compared to other $\mathbb{F}_p$ operations, whereas they are usually neglected in the literature. However, the results obtained are still valid in the case where $\mathbb{F}_p$ additions can be neglected. Most of the content of this paper already lies in the literature or in existing implementations but, even if it is not our initial purpose, we also give some new ideas to minimize the number of $\mathbb{F}_p$ additions during the pairing computation. We also explain why the best choice for the polynomials defining the tower field $\mathbb{F}_{p^{12}}$ is only depending on the value of the BN parameter $u$ modulo small integers like 12.

The paper is organized as follows. In Section 2 we recall how the optimal Ate pairing on BN curves is computed. In Section 3 we present the different options to build $\mathbb{F}_{p^{12}}$ and how to implement the basic operations. This includes the choice of the defining polynomials in terms of $u$, the choice of the tower structure and the choice of the algorithms for basic arithmetic, depending on the relative cost of $\mathbb{F}_p$ operations. In Section 4 we explain how to choose the BN parameter $u$. Sections 5 and 6 are devoted to the choices inherent to the curve (coefficients, generators, system of coordinates). Finally, in Section 7 we recall the other algorithms that must be used for efficient implementation and adapt them to our context and to the results obtained in the previous sections.

The following notations for $\mathbb{F}_{p^i}$ arithmetic will be used:

- $\mathbf{A}_i$ denotes an addition and $\mathbf{A'}_i$ denotes a multiplication by 2,
- $\mathbf{M}_i$ denotes a multiplication ($\mathbf{M}_i^M$ if the method $M$ is used),
- $\mathbf{sM}_i$ denotes a sparse multiplication,
- $\mathbf{m}_{i,c}$ denotes a multiplication by a constant $c$,
- $\mathbf{S}_i$ denotes a squaring ($\mathbf{S}_i^M$ if the method $M$ is used),
- $\mathbf{I}_i$ denotes an inversion.

# 2 Background

## 2.1 BN curves

A Barreto-Naehrig (BN) curve [8] is an elliptic curve $E$ over a finite field $\mathbb{F}_p$, $p \geq 5$, with order $r = \#E(\mathbb{F}_p)$, such that $p$ and $r$ are prime numbers given by

$$\begin{aligned} p &= 36u^4 + 36u^3 + 24u^2 + 6u + 1, \\ r &= 36u^4 + 36u^3 + 18u^2 + 6u + 1, \end{aligned}$$

for some $u$ in $\mathbb{Z}$. It has an equation of the form

$$y^2 = x^3 + b,$$

where $b \in \mathbb{F}_p^*$. Its neutral element is denoted by $O_E$.

BN curves have been designed to have an embedding degree equal to 12. This makes them particularly appropriate for the 128-bit security level. Indeed, a prime $p$ of size 256 bits leads to a BN curve whose group order is roughly 256 bits together with pairings taking values in $\mathbb{F}_{p^{12}}^*$, which is a 3072-bit multiplicative group. According to the NIST recommendations [45], both groups involved are matching the 128-bit security level. By the way, BN curves at this security level have been the object of numerous recent publications ([21, 2, 14, 24, 43, 27, 53]).

Finally, BN curves always have degree 6 twists. If $\xi$ is an element which is neither a square nor a cube in $\mathbb{F}_{p^2}$, the twisted curve $E'$ of $E$ is defined over $\mathbb{F}_{p^2}$ by the equation

$$E' : y^2 = x^3 + b',$$

with $b' = b/\xi$ or $b' = b\xi$. In order to simplify the computations, the element $\xi$ should also be used to represent $\mathbb{F}_{p^{12}}$ as a degree 6 extension of $\mathbb{F}_{p^2}$ ($\mathbb{F}_{p^{12}} = \mathbb{F}_{p^2}[\gamma]$ with $\gamma^6 = \xi$) [21], [37]. In this paper, we deal only with the case $b' = b/\xi$ as usually done in the literature but $b' = b/\xi^5$ can be also used with a very small additional cost [27].

## 2.2 Optimal Ate pairing

Let $a$ be an integer and $Q \neq O_E$ be a point on $E$. We denote by $f_{a,Q}$ the normalized function on the curve with divisor

$$\operatorname{div}(f_{a,Q}) = a[Q] - [aQ] - (a-1)\,[O_E]\,.$$

Such functions are the core of all known pairings. They are computed thanks to the Miller loop (described in Section 2.3), which is an adaptation of the classical scalar multiplication algorithm. For example, the (reduced) Tate pairing is defined by

$$e_T(P,Q) = f_{r,P}(Q)^{\frac{p^{12}-1}{r}}.$$

There are many variants of the Tate pairing allowing to use a smaller value of $a$ in order to shorten the length of the Miller loop ([6, 29, 28, 39, 36, 54]). It has been proven in [54] that the shortest possible loop has length $r/\varphi(12) = r/4$ and that this length is reached by the so-called optimal Ate pairing.

Let $\pi(x,y) = (x^p, y^p)$ be the Frobenius map on the curve. If $P$ is a rational point on $E$ and $Q$ is a point in $E\left(\mathbb{F}_{p^{12}}\right)$ which is in the $p$-eigenspace of $\pi$, the optimal Ate pairing [43] can be defined by

$$a_{opt}(Q,P) = \left(f_{v,Q}(P).\ell_{vQ,\pi(Q)}(P).\ell_{vQ+\pi(Q),-\pi^2(Q)}(P)\right)^{\frac{p^{12}-1}{r}},$$

where $v = 6u + 2$ and $\ell_{A,B}$ is the normalized line function arising in the sum of the points $A$ and $B$.

In this study, we are only considering this pairing because it makes no doubt that it is currently the most efficient for BN curves, but the same work can be easily done with other pairings. The computation of the optimal Ate pairing is done in four steps:

1. A Miller loop to compute $f_{|v|,Q}(P)$. The algorithmic choices for this step are discussed in Section 2.3.

2. If $v < 0$, the result $f$ of the Miller loop must be inverted to recover $f_{v,Q}(P)$. Such an inversion is potentially expensive but thanks to the final exponentiation, $f^{-1}$ can be replaced by $f^{p^6}$ [2] which is nothing but the conjugation in $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^6}$, thus it is for free.

3. Two line computations, $\ell_{vQ,\pi(Q)}(P)$ and $\ell_{vQ+\pi(Q),-\pi^2(Q)}(P)$ which are nothing but extra addition steps of the Miller loop.

4. A final exponentiation to the power of $\frac{p^{12}-1}{r}$. The algorithmic choices for this step are discussed in Section 2.4.

Since BN curves have twists of order 6, a twisted version of the optimal Ate pairing allows to take $Q$ in $E'\left(\mathbb{F}_{p^2}\right)$. Using the isomorphism between the curve and its twist, the point $Q$ in our definition of the optimal Ate pairing can then be chosen of the form $\left(x_Q\gamma^2, y_Q\gamma^3\right) \in E\left(\mathbb{F}_{p^{12}}\right)$, where $x_Q, y_Q \in \mathbb{F}_{p^2}$ $\left((x_Q, y_Q) \in E'\left(\mathbb{F}_{p^2}\right)\right)$. This means that elliptic curve operations lie in $\mathbb{F}_{p^2}$ instead of $\mathbb{F}_{p^{12}}$ (but the result remains in $\mathbb{F}_{p^{12}}$). Of course, this makes computations easier, but this also allows denominator elimination as in [7] because all the factors lying in a proper subfield of $\mathbb{F}_{p^{12}}$ (as $\mathbb{F}_{p^2}$) are wiped out by the final exponentiation.

## 2.3 The Miller loop

The first step of the pairing computation evaluates $f_{|v|,Q}(P)$ thanks to the Miller algorithm presented in Algorithm 1 and introduced in [40]. It is based on the double and add scheme used for the computation of $|v|Q$ by evaluating at $P$ the lines occurring in the doubling and addition steps of this computation. More precisely, it is based on Miller's formula given in the following Lemma, which can be proven by considering divisors.

**Lemma 1.** *For all $a$, $b \in \mathbb{Z}$, $Q$ a point of $E$, we have*

$$f_{a+b,Q} = f_{a,Q} f_{b,Q} \frac{\ell_{aQ,bQ}}{v_{(a+b)Q}}$$

*where $\ell_{aQ,bQ}$ is the equation of the line passing through the points $aQ$ and $bQ$ and $v_{(a+b)Q}$ is the equation of the vertical line passing through the point $(a+b)Q$.*

**Remark 1.** *In the case of the optimal Ate pairing, the point $Q$ comes from the twisted curve. Hence its $x$-coordinate lies in a proper subfield of $\mathbb{F}_{p^{12}}$ and $p \in E(\mathbb{F}_p)$ so that $v_{(a+b)Q}(P)$ also lies in a proper subfield of $\mathbb{F}_{p^{12}}$ and then* *is wiped out by the final exponentiation. This is known as the denominator elimination optimisation [33]. Then we do not take $v_{(a+b)Q}$ into consideration in the following.*

Miller's algorithm makes use of Lemma 1 with $b = a$ (for doubling steps) or $b = 1$ (for addition steps) and is described by the pseudocode in Algorithm 1 assuming Remark 1.

**Algorithm 1:** Miller$(P, Q, a)$

```
Data: $a = (a_n \dots a_0)_2$,
      $P \in E(\mathbb{F}_p)$,
      $Q \in E(\mathbb{F}_{p^{12}})$ having its $x$-coordinate in a proper subfield of $\mathbb{F}_{p^{12}}$;
Result: $\lambda f_{a,Q}(P) \in \mathbb{F}^*_{p^{12}}$ with $\lambda$ in a proper subfield of $\mathbb{F}_{p^{12}}$;
$T \leftarrow Q$ ;
$f \leftarrow 1$ ;
for $i = n - 1$ to $0$ do
    $f \longleftarrow f^2 \times \ell_{T,T}(P)$;
    $T \leftarrow 2T$;
    if $a_i = 1$ then
        $f \longleftarrow f \times \ell_{T,Q}(P)$;
        $T \leftarrow T + Q$ ;
    end
end
return $f$
```

Several choices are possible for the system of coordinates in order to perform the operations over the elliptic curve during the Miller loop. We discuss them in Section 6.

Since the Miller algorithm is based on the double and add algorithm, it is natural to try to improve it by using advanced exponentiation techniques like the sliding window method [17, Algo 9.10] or the NAF representation [17, Algo 9.14]. However, the interest is limited in practice for two reasons:

- In the context of pairing based cryptography, the exponent is not a secret. Then it is usually chosen sparse so these advanced exponentiation methods are useless.

- Such methods involve operations like $T \leftarrow T + 3Q$. We need to compute $f \leftarrow f \times f_{3,Q} \times \ell_{T,3Q}$ to obtain the corresponding function. Of course, $f_{3,Q}$ can be precomputed but such a step requires an additional $\mathbb{F}_{p^{12}}$ multiplication which is the most consuming operation in Algorithm 1.

The only interesting case is a signed binary representation of the exponent (*i.e.* a 2-NAF) because it can help to find a sparse exponent. In this case, the substraction step of Algorithm 1 is involving an additional division by the vertical line passing trough $Q$ and $-Q$ which could be expensive, but fortunately it is wiped out by the final exponentiation if $Q$ comes from the twisted curve.

## 2.4 The final exponentiation

As proposed in [33, 50] the cost of the final exponentiation can be reduced thanks to the integer factorization

$$\frac{p^{12}-1}{r} = \left(p^6-1\right)\left(p^2+1\right)\left(\frac{p^4-p^2+1}{r}\right).$$

Since $p$ is the characteristic of $\mathbb{F}_{p^{12}}$, it is easy to compute the $p^{\text{th}}-$power of any element in $\mathbb{F}_{p^{12}}$. More details about these Frobenius computations can be found in Section 7.1. Powering to the $\left(p^6-1\right)\left(p^2+1\right)$ is then called the easy part of the final exponentiation, even if an expensive inversion in $\mathbb{F}_{p^{12}}$ is required.

Powering to the $\frac{p^4-p^2+1}{r}$ is called the hard part of the final exponentiation. For this computation, the exponent is usually developed in base $p$ in order to use again cheap Frobenius computations ([49, 21, 50, 13, 23]), and the cost is around three times the cost of an exponentiation by $u$.

Moreover, as $f$ has been raised to the power of $\left(p^6-1\right)\left(p^2+1\right)$, it has order dividing $p^4-p^2+1$. Then, as noticed in [26], it lies in the cyclotomic group $G_{\Phi_6}(\mathbb{F}_{p^2})$. This has two important consequences for the efficiency of the computation :

- Squaring is less expensive than a classical squaring in $\mathbb{F}_{p^{12}}$ [26, 31] (more details are given in Section 7.2),
- Inversion is the same operation as raising to the power $p^6$, which is nothing but the conjugation in $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^6}$, thus it is for free [49, 52].

The most popular way to perform this hard part uses the following addition chain [50] :

$$f^{\frac{p^4-p^2+1}{r}} = y_0 y_1^2 y_2^6 y_3^{12} y_4^{18} y_5^{30} y_6^{36},$$

where
$$y_0 = f^p f^{p^2} f^{p^3}, \quad y_1 = \frac{1}{f}, \quad y_2 = \left(f^{u^2}\right)^{p^2}, \quad y_3 = \left(f^u\right)^p,$$
$$y_4 = \frac{\left(f^{u^2}\right)^p}{f^u}, \quad y_5 = \frac{1}{f^{u^2}}, \quad y_6 = \frac{\left(f^{u^3}\right)^p}{f^{u^3}}.$$

The cost of this method is $13\mathbf{M}_{12}, 4\mathbf{S}_{12}$ and 7 Frobenius maps in addition to the cost of 3 exponentiations by $u$. The method given in [13] is slightly more efficient but computes a power of the optimal Ate pairing. The main drawback of these methods is that they are memory consuming (up to 4Ko), which can be annoying in restricted environments. Some variants of these methods optimized in terms of memory consumption are given in [23].

# 3 Choosing the finite fields and their arithmetic

## 3.1 $\mathbb{F}_p$ arithmetic

In this paper, we assume that the basic operations in $\mathbb{F}_p$ (additions, subtractions, multiplications) are already implemented. This multiple precision arithmetic is usually performed by combining the schoolbook method or the Karatsuba method [32] and the Montgomery reduction [42] or the Barrett reduction [9].

In the case where $p$ is taken to be a generalized Mersene prime [51, 15], the field operations can be sped-up by exploiting the particular shape of $p$. However, no general method to produce ordinary pairing-friendly elliptic curves with such $p$ is known, and moreover, using low weight primes could introduce weakness in pairing based cryptosystems [48].

As an alternative, several authors (e.g. [22], [14]) proposed to use the Residue Number Systems (RNS) for pairing computation (an integer $a$ modulo $p$ is represented by $(a_1, \ldots, a_n)$, where $a_i = a \mod m_i$, for some well-chosen coprime integers $m_1, \ldots, m_n$). The RNS allows a rather low cost for additions and multiplications, but on the other hand, the reduction step is expensive, so it is recommended to accumulate several operations before performing a reduction (this is called lazy reduction), which requires some extra memory.

Anyway, the method used for $\mathbb{F}_p$ arithmetic does not have direct consequences on the choices to be made for extension field or elliptic curve arithmetic. The only important criterion is the relative cost between the $\mathbb{F}_p$ operations ($\mathbf{A}_1$, $\mathbf{A'}_1$, $\mathbf{M}_1$, $\mathbf{S}_1$, $\mathbf{I}_1$). One of the main purpose of this paper is to discuss the available choices depending on these ratios. We plan to cover most of the possible practical situations, so that we make the following assumptions:

- Addition can be relatively expensive ($\mathbf{A}_1 < 0.5\mathbf{M}_1$). They are often neglected in theoretical studies but this should not be the case for real life implementations, especially in low level implementation where the most common ratios at the 128-bit security level are between 0.2 and 0.3 [47, 25], but also in software implementation (for example, the ratio is 0.17 in the Microsoft ECC library [46] at the 128-bit security level). Therefore, the number of $\mathbb{F}_p$ additions involved in extension field arithmetic must be taken into account and may have an influence on the choices to be made.

- In order to stay in the most general case, we chose to use distinct symbols to denote an addition ($\mathbf{A}_1$) and a doubling ($\mathbf{A'}_1$). However, notice that we usually have $\mathbf{A'}_1 = \mathbf{A}_1$.

- If a specific algorithm for squaring is implemented, then $\mathbf{S}_1$ is usually assumed to be $0.8\mathbf{M}_1$, else, a square is computed by doing a multiplication and $\mathbf{S}_1 = \mathbf{M}_1$.

## 3.2 Arithmetic of $\mathbb{F}_{p^{2i}}/\mathbb{F}_{p^i}$

In theory, any irreducible quadratic polynomial can be used to build $\mathbb{F}_{p^{2i}}$ over $\mathbb{F}_{p^i}$, but non-zero coefficients of this polynomial imply extra operations for $\mathbb{F}_{p^{2i}}$ arithmetic. Therefore, $\mathbb{F}_{p^{2i}}$ is usually built with a polynomial of the form $X^2-\mu$, where $\mu$ is not a square in $\mathbb{F}_{p^i}$.

$$\mathbb{F}_{p^{2i}} = \mathbb{F}_{p^i}[\alpha] \text{ with } \alpha^2 = \mu.$$

### 3.2.1 $\mathbb{F}_{p^{2i}}$ addition

Whatever the choice for building $\mathbb{F}_{p^{2i}}$, an addition (resp. a multiplication by 2) always requires 2 $\mathbb{F}_{p^i}$ additions (resp. multiplications by 2). In any case

$$\mathbf{A}_{2i} = 2\mathbf{A}_i \text{ and } \mathbf{A'}_{2i} = 2\mathbf{A'}_i.$$

### 3.2.2 $\mathbb{F}_{p^{2i}}$ multiplication

We will consider only two methods since the other methods do not reduce the overall complexity, independently of the relative cost of $\mathbb{F}_{p^i}$ operations.

**Schoolbook method.** The method computes as follows:

$$(x_0 + x_1\alpha)(y_0 + y_1\alpha) = x_0y_0 + \mu x_1y_1 + (x_0y_1 + x_1y_0)\alpha$$

and requires $\mathbf{M}_{2i}^{\mathcal{SB}} = 4\mathbf{M}_i + \mathbf{m}_{i,\mu} + 2\mathbf{A}_i$.

**Karatsuba method.** The method is a standard variant of the schoolbook method. The evaluation is performed as follows:

$$(x_0 + x_1\alpha)(y_0 + y_1\alpha) = x_0y_0 + \mu x_1y_1 + ((x_0 + x_1)(y_0 + y_1) - x_0y_0 - x_1y_1)\alpha$$

and requires $\mathbf{M}_{2i}^{\mathcal{K}} = 3\mathbf{M}_i + \mathbf{m}_{i,\mu} + 5\mathbf{A}_i$.

**Remark 2.** *The schoolbook method should be preferred to the Karatsuba method when* $3\mathbf{A}_i > \mathbf{M}_i$.

### 3.2.3 $\mathbb{F}_{p^{2i}}$ squaring

In this case, we will additionally consider the complex method.

**Schoolbook method.** In the squaring case, we get

$$(x_0 + x_1\alpha)^2 = x_0^2 + \mu x_1^2 + 2x_0x_1\alpha$$

which requires $\mathbf{S}_{2i}^{\mathcal{SB}} = \mathbf{M}_i + 2\mathbf{S}_i + \mathbf{m}_{i,\mu} + \mathbf{A}_i + \mathbf{A'}_i$.

**Karatsuba method.** In the squaring case, we get

$$(x_0 + x_1\alpha)^2 = x_0^2 + \mu x_1^2 + ((x_0 + x_1)^2 - x_0^2 - x_1^2)\alpha$$

which requires $\mathbf{S}_{2i}^{K} = 3\mathbf{S}_i + \mathbf{m}_{i,\mu} + 4\mathbf{A}_i$.

**Complex method.** The method is particularly efficient if $\mu = -1$, which explains its name. It computes as follows:

$$(x_0 + x_1\alpha)^2 = (x_0 + \mu x_1)(x_0 + x_1) - (\mu + 1)x_0x_1 + 2x_0x_1\alpha$$

and requires $\mathbf{S}_{2i}^{\mathbb{C}} = 2\mathbf{M}_i + \mathbf{m}_{i,\mu} + \mathbf{m}_{i,\mu+1} + 3\mathbf{A}_i + \mathbf{A'}_i$.

**Remark 3.** *Determining which method is the best is not as easy as for multiplication in the general case because it depends on both the relative cost of* $\mathbf{M}_i$ *and* $\mathbf{A}_i$ *and of* $\mathbf{M}_i$ *and* $\mathbf{S}_i$*. We postpone this question to Section 3.5 which discusses specific choices of* $\mu$*.*

### 3.2.4 $\mathbb{F}_{p^{2i}}$ inversion

The $\mathbb{F}_{p^{2i}}$ inversion is classically done thanks to the norm of an element

$$N = (x_0 + x_1\alpha)\,(x_0 - x_1\alpha) = x_0^2 - \mu x_1^2 \in \mathbb{F}_p.$$

We easily get the inverse of $x_0 + x_1\alpha$ as $\frac{x_0}{N} - \frac{x_1}{N}\alpha$. This way of computing an inverse in $\mathbb{F}_{p^{2i}}$ requires

$$\mathbf{I}_{2i} = \mathbf{I}_i + 2\mathbf{M}_i + 2\mathbf{S}_i + \mathbf{A}_i + \mathbf{m}_{i,\mu}.$$

## 3.3 Arithmetic of $\mathbb{F}_{p^{3i}}/\mathbb{F}_{p^i}$

As in Section 3.2, it is preferable to chose a sparse polynomial to minimize the cost of $\mathbb{F}_{p^{3i}}$ arithmetic. Thus, $\mathbb{F}_{p^{3i}}$ is built as $\mathbb{F}_{p^i}[\alpha]$, where $\alpha^3 = \xi$ for some $\xi$ in $\mathbb{F}_{p^i}$. Of course, $\mathbb{F}_{p^{3i}}$ arithmetic will involve some multiplications by $\xi$, so that $\xi$ must be chosen carefully.

### 3.3.1 $\mathbb{F}_{p^{3i}}$ addition

Whatever the choice for building $\mathbb{F}_{p^{3i}}$, an addition (resp. a multiplication by 2) always requires 3 $\mathbb{F}_{p^i}$ additions (resp. multiplications by 2). So in any case

$$\mathbf{A}_{3i} = 3\mathbf{A}_i \text{ and } \mathbf{A'}_{3i} = 3\mathbf{A'}_i.$$

### 3.3.2 $\mathbb{F}_{p^{3i}}$ multiplication

Again, we give only the schoolbook and the Karatsuba methods because other methods (like Toom-Cook) require too many additions (and divisions by 2 or 3), which is contradictory with our assumptions (additions are not so negligible on real world devices).

**Schoolbook method.** The method uses the following equality

$$\begin{aligned}(x_0 + x_1\alpha + x_2\alpha^2)(y_0 + y_1\alpha + y_0\alpha^2) = {} & x_0y_0 + \xi(x_1y_2 + x_2y_1) \\ & + [x_0y_1 + x_1y_0 + \xi x_2y_2]\alpha \\ & + [x_0y_2 + y_2x_0 + x_1y_1]\alpha^2\end{aligned}$$

and requires $\mathbf{M}_{3i}^{\mathcal{SB}} = 9\mathbf{M}_i + 2\mathbf{m}_{i,\xi} + 6\mathbf{A}_i$.

**Karatsuba method.** As in the case of quadratic extensions, the method allows to compute sums of products like $x_1y_2+x_2y_1$ with only one multiplication, assuming that $x_1y_1$ and $x_2y_2$ are already computed. The equality in the cubic case is

$$\begin{aligned}(x_0 + x_1\alpha + x_2\alpha^2)(y_0 + y_1\alpha + y_2\alpha^2) = {} & x_0y_0 + \xi((x_1 + x_2)(y_1 + y_2) - x_1y_1 - x_2y_2) \\ & + [(x_0 + x_1)(y_0 + y_1) - x_0y_0 - x_1y_1 + \xi x_2y_2]\alpha \\ & + [(x_0 + x_2)(y_0 + y_2) - x_0y_0 - x_2y_2 + x_1y_1]\alpha^2\end{aligned}$$

This means that $2\mathbf{M}_i + \mathbf{A}_i$ is replaced by $\mathbf{M}_i + 4\mathbf{A}_i$ three times. As in the case of quadratic extensions, the Karatsuba method then becomes interesting when $\mathbf{M}_i \geq 3\mathbf{A}_i$ and requires $\mathbf{M}_{3i}^{\mathcal{K}} = 6\mathbf{M}_i + 15\mathbf{A}_i + 2\mathbf{m}_{i,\xi}$.

We will also use in the following the Karatsuba method when one of the operands is sparse. If two coefficients are zero, the Karatsuba method has no interest but it can be used if one coefficient, say $y_2$, is zero. In this case, we have

$$\begin{aligned}(x_0 + x_1\alpha + x_2\alpha^2)(y_0 + y_1\alpha) = {} & x_0y_0 + \xi x_2y_1 \\ & + [(x_0 + x_1)(y_0 + y_1) - x_0y_0 - x_1y_1]\alpha \\ & + [x_2y_0 + x_1y_1]\alpha^2\end{aligned}$$

which requires $\mathbf{sM}_{3i}^{\mathcal{K}} = 5\mathbf{M}_i + 6\mathbf{A}_i + \mathbf{m}_{i,\xi}$.

### 3.3.3 Squaring

The Schoolbook and Karatsuba squarings are deduced from the multiplication.

**Schoolbook method.** The method uses the equality

$$(x_0 + x_1\alpha + x_2\alpha^2)^2 \quad = \quad x_0^2 + 2\xi x_1x_2 + \left[2x_0x_1 + \xi x_2^2\right]\alpha + \left[x_1^2 + 2x_0x_2\right]\alpha^2.$$

Note that if we first compute $2x_1$, only 2 multiplications by 2 are necessary to evaluate this formula, so the cost is $\mathbf{S}_{3i}^{\mathcal{SB}} = 3\mathbf{M}_i + 3\mathbf{S}_i + 3\mathbf{A}_i + 2\mathbf{A}'_i + 2\mathbf{m}_{i,\xi}$.

**Karatsuba Method.** The method computes the double products involved in the schoolbook squaring using

$$2x_0x_1 \quad = \quad (x_0 + x_1)^2 - x_0^2 - x_1^2.$$

The complexity becomes $\mathbf{S}_{3i}^{\mathcal{K}} = 6\mathbf{S}_i + 12\mathbf{A}_i + 2\mathbf{m}_{i,\xi}$.

**Chung-Hasan Method.** We can also use the Chung-Hasan method [16] for squaring in degree 3 extensions. There are several variants but the most interesting in our context is to compute the term in $\alpha^2$ in the schoolbook method using the formula

$$x_1^2 + 2x_0x_2 = (x_0 + x_1 + x_2)^2 - (2x_0x_1 + 2x_1x_2 + x_0^2 + x_2^2).$$

Computing this term requires $\mathbf{S}_i + 6\mathbf{A}_i$ instead of $\mathbf{M}_i + \mathbf{S}_i + \mathbf{A}_i + \mathbf{A'}_i$, so the overall complexity is $\mathbf{S}_{3i}^{CH} = 2\mathbf{M}_i + 3\mathbf{S}_i + 8\mathbf{A}_i + \mathbf{A'}_i + 2\mathbf{m}_{i,\xi}$.

### 3.4 Building $\mathbb{F}_{p^{12}}$

In this section, we discuss the ways to build the extension tower $\mathbb{F}_{p^{12}}$ for pairings on BN curves. All the ways to build $\mathbb{F}_{p^{12}}$ are mathematically equivalent. However, we will use this extension in the specific case of pairings on BN curves, which implies some constraints in order to allow other improvements for pairing computation.

- As explained in Section 2.2, $\mathbb{F}_{p^{12}}$ must be built as an extension of $\mathbb{F}_{p^2}$ because of the use of a sextic twist. Indeed, the twisted curve is defined over $\mathbb{F}_{p^2}$.
- $\mathbb{F}_{p^{12}}$ must be built over $\mathbb{F}_{p^2}$ with a polynomial $X^6 - \xi$ where $\xi$, which is neither a square nor a cube, is the element used to define the twisted curve. This allows the line involved in the Miller algorithm to be a sparse element of $\mathbb{F}_{p^{12}}$ (see Section 6 for more details).

Then, $\mathbb{F}_{p^{12}}$ should be built

- Case $2, 2, 3$: as a cubic extension of a quadratic extension of $\mathbb{F}_{p^2}$,
- Case $2, 3, 2$: as a quadratic extension of a cubic extension of $\mathbb{F}_{p^2}$,
- Case $2, 6$: as a sextic extension of $\mathbb{F}_{p^2}$.

The latter case is proved to be less efficient [20], so we will only consider here the first two ones. In any case, we have

$$\mathbb{F}_{p^{12}} = \mathbb{F}_{p^2}[\gamma] \text{ with } \gamma^6 = \xi \in \mathbb{F}_{p^2}.$$

In the case $2, 2, 3$, we will use $\beta = \gamma^3$ to define $\mathbb{F}_{p^4}$ and in the case $2, 3, 2$, we will use $\beta = \gamma^2$ to define $\mathbb{F}_{p^6}$. These cases are studied in detail in Sections 3.6 and 3.7.

Of course, $\xi$ must be carefully chosen, since $\mathbb{F}_{p^{12}}$ arithmetic will involve multiplications by $\xi$ or $\beta$; this is the purpose of Section 3.8. But let us first give more details on the choice of $\mathbb{F}_{p^2}$ and its arithmetic.

## 3.5 Choice of $\mathbb{F}_{p^2}$ and its arithmetic

As explained in Section 3.2, $\mathbb{F}_{p^2}$ is built thanks to an element $\mu$ which is not a square in $\mathbb{F}_p$ so that

$$\mathbb{F}_{p^2} = \mathbb{F}_p[\alpha] \text{ with } \alpha^2 = \mu.$$

According to Remark 2, the Karatsuba method for $\mathbb{F}_{p^2}$ multiplication is better when $\mathbf{A}_1 < \frac{1}{3}\mathbf{M}_1$.

Determining the best squaring algorithm is less simple. Let us first compare the schoolbook and the Karatsuba method. The difference between the complexities is $\mathbf{M}_1 - \mathbf{S}_1 + \mathbf{A'}_1 - 3\mathbf{A}_1$. Assuming that $\mathbf{S}_1 = \mathbf{M}_1$, this difference becomes $\mathbf{A'}_1 - 3\mathbf{A}_1$ which is always negative. This means that the schoolbook method is always better. Assuming that $\mathbf{S}_1 = 0.8\mathbf{M}_1$, the difference becomes $0.2\mathbf{M}_1 + \mathbf{A'}_1 - 3\mathbf{A}_1$ which is negative if $\mathbf{A}_1 \geq 0.1\mathbf{M}_1$ (and even less if $\mathbf{A'}_1$ is assumed to be cheaper than $\mathbf{A}_1$). Thus the schoolbook method is also better in this case, unless if the addition in $\mathbb{F}_p$ is really very efficient ($\mathbf{A}_1 < 0.1\mathbf{M}_1$). We then assume that the Karatsuba method for $\mathbb{F}_{p^2}$ squaring has no (or very few) interest.

Let us now compare the Schoolbook method and the complex method. In this case, the difference between the complexities is $2\mathbf{S}_1 - \mathbf{M}_1 - 2\mathbf{A}_1 - \mathbf{m}_{1,\mu+1}$. This means that the schoolbook method is generally better if $\mu$ is randomly chosen ($\mathbf{m}_{1,\mu+1} = \mathbf{M}_1$). If $\mu$ is chosen to be a small number, the conclusion is depending on this choice (more precisely on the cost of the multiplication by $\mu + 1$). The choice of $\mu$ is the object of Section 3.5.1.

### 3.5.1 Choice of $\mu$ and consequences on $u$

We focus on multiplications and squarings in $\mathbb{F}_{p^2}$ because other operations such as inversions or additions are rare or independent of the choice of $\mu$. In order to have the best arithmetic, $\mu$ must be chosen such that $\mathbf{m}_{1,\mu}$ and $\mathbf{m}_{1,\mu+1}$ are as cheap as possible. Since $\mu$ has to be a non-square in $\mathbb{F}_p$, 0 and 1 must be avoided and the best choice is $\mu = -1$. If this choice is not possible, $\mu = \pm 2$ is also a good choice. Other choices are possible, even if they are of course more expensive. Let us now give more details on which values of $\mu$ should be used to define $\mathbb{F}_{p^2}$, depending on $u$. Further justifications are given in Appendix A.

**The case $\mathbb{F}_{p^2} = \mathbb{F}_p[\mathbf{i}]$.** If $\mu = -1$, then $\mathbb{F}_{p^2}$ can be seen as an analogue of the complex field and in this case $\alpha$ is usually denoted by $\mathbf{i}$. In this situation, $\mathbf{m}_{1,\mu}$ and $\mathbf{m}_{1,\mu+1}$ are both for free, but one can do even better because the complex method for squaring becomes

$$(a_0 + a_1\mathbf{i})^2 = (a_0 - a_1)(a_0 + a_1) + 2a_0a_1\mathbf{i}$$

and requires $2\mathbf{M}_1 + 2\mathbf{A}_1 + \mathbf{A'}_1$ which is faster than schoolbook or Karatsuba method in any case.

Choosing $\mu = -1$ is possible if and only if $-1$ is not a square in $\mathbb{F}_p$, which is equivalent to take $u$ odd, as proved in Proposition 1 of Appendix A.

**The case $\mathbb{F}_{p^2} = \mathbb{F}_p\left[\sqrt{-2}\right]$.** In this case, $\mathbf{m}_{1,\mu} = \mathbf{A'}_1$ and $\mathbf{m}_{1,\mu+1} = 0$. The complex method for computing a square in $\mathbb{F}_{p^2}$ then requires $2\mathbf{M}_1 + 3\mathbf{A}_1 + 2\mathbf{A'}_1$ whereas the schoolbook method needs $\mathbf{M}_1 + 2\mathbf{S}_1 + \mathbf{A}_1 + 2\mathbf{A'}_1$. The difference between the complexities is $2\mathbf{S}_1 - \mathbf{M}_1 - 2\mathbf{A}_1$. The complex method is then always better if $\mathbf{S}_1 = \mathbf{M}_1$. If $\mathbf{S}_1 = 0.8\mathbf{M}_1$, the schoolbook method is better only if $\mathbf{A}_1 > 0.3\mathbf{M}_1$.

According to Proposition 1 of Appendix A, choosing $\mu = -2$ is possible if and only if $u = 1$ or 2 modulo 4. However, if $u = 1$ modulo 4, choosing $\mu = -1$ is more appropriate in terms of efficiency.

**The case $\mathbb{F}_{p^2} = \mathbb{F}_p\left[\sqrt{-5}\right]$.** In this case, $\mathbf{m}_{1,\mu} = 2\mathbf{A'}_1 + \mathbf{A}_1$ and $\mathbf{m}_{1,\mu+1} = 2\mathbf{A'}_1$. However, the complex method can be rewritten in a more efficient way

$$(a_0 + a_1\alpha)^2 = (a_0 + a_1 + 2a_1')(a_0 + a_1) + 2a_0a_1' + a_0a_1'\alpha \text{ with } a_1' = 2a_1$$

and then requires $2\mathbf{M}_1 + 4\mathbf{A}_1 + 2\mathbf{A'}_1$ which is almost as good as the case $\mu = -2$ and better than the schoolbook method (except if $\mathbf{A'}_1$ is very small compared to $\mathbf{A}_1$ and $\mathbf{S}_1 = 0.8\mathbf{M}_1$).

If $u$ is odd, it is better to choose $\mu = -1$, and if $u = 2$ modulo 4 it is better to choose $\mu = \pm 2$, so we assume that $u = 0$ modulo 4. In this case, according to Proposition 1 of Appendix A, 5 and $-5$ are squares together when $u = 0$ or 4 modulo 5. Therefore, $\mu = -5$ should be chosen when $u = 8, 12$ or 16 modulo 20.

### Other choices for building $\mathbb{F}_{p^2}$

- If $\mu = -1$ and $\mu = -2$ cannot be chosen to build $\mathbb{F}_{p^2}$, then 2 cannot be chosen either. Moreover, choosing $\mu = 2$ instead of $-2$ is less efficient for the complex method since $\mu + 1 = 3$ instead of $-1$. Therefore, choosing $\mu = 2$ has no interest.
- The same remark holds for $\mu = 5$.
- The case $\mu = 3$ is interesting at first glance because the complexity of the complex method is as good as when $\mu = -2$. However, this choice can be done if and only if $u$ is odd, and in this case, choosing $\mu = -1$ is more appropriate in terms of efficiency.
- Since $p = 1$ modulo 3 for BN primes, $-3$ is always a square in $\mathbb{F}_p$, so it cannot be chosen to build $\mathbb{F}_{p^2}$.

Finally, the only interesting small values for $\mu$ are $-1, -2$ and $-5$ and one of them can be used whenever $u \neq 0$ or 4 modulo 20.

### 3.5.2 Summary of choices and complexities for $\mathbb{F}_{p^2}$ arithmetic

We saw that we trivially have $\mathbf{A}_2 = 2\mathbf{A}_1$ and $\mathbf{A}'_2 = 2\mathbf{A}'_1$ whatever the choice made to build $\mathbb{F}_{p^2}$. The situation is more complicated for $\mathbf{M}_2$ and $\mathbf{S}_2$; it is summarized in Tables 1 and 2. In these tables SB, K and $\mathbb{C}$ denote the method used.

| $\mu$ | Assuming $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$ | Assuming $\mathbf{A}_1 > 0.33\mathbf{M}_1$ | Condition |
|---|---|---|---|
| $-1$ | $3\mathbf{M}_1 + 5\mathbf{A}_1$ (K) | $4\mathbf{M}_1 + 2\mathbf{A}_1$ (SB) | $u = 1 \bmod 2$ |
| $-2$ | $3\mathbf{M}_1 + 5\mathbf{A}_1 + \mathbf{A}'_1$ (K) | $4\mathbf{M}_1 + 2\mathbf{A}_1 + \mathbf{A}'_1$ (SB) | $u = 2 \bmod 4$ |
| $-5$ | $3\mathbf{M}_1 + 6\mathbf{A}_1 + 2\mathbf{A}'_1$ (K) | $4\mathbf{M}_1 + 3\mathbf{A}_1 + 2\mathbf{A}'_1$ (SB) | $u = 8, 12, 16 \bmod 20$ |
| any | $4\mathbf{M}_1 + 5\mathbf{A}_1$ (K) | $5\mathbf{M}_1 + 2\mathbf{A}_1$ (SB) | $u = 0, 4 \bmod 20$ |

Table 1: Complexities of $\mathbf{M}_2$ depending on the way to build $\mathbb{F}_{p^2}$

| | Assuming $\mathbf{S}_1 = \mathbf{M}_1$ or | Assuming | Condition |
|---|---|---|---|
| $\mu$ | $\mathbf{S}_1 = 0.8\mathbf{M}_1, \mathbf{A}_1 \leq 0.3\mathbf{M}_1$ | $\mathbf{S}_1 = 0.8\mathbf{M}_1, \mathbf{A}_1 > 0.3\mathbf{M}_1$ | |
| $-1$ | $2\mathbf{M}_1 + 2\mathbf{A}_1 + \mathbf{A}'_1$ ($\mathbb{C}$) | | $u = 1 \bmod 2$ |
| $-2$ | $2\mathbf{M}_1 + 3\mathbf{A}_1 + 2\mathbf{A}'_1$ ($\mathbb{C}$) | $\mathbf{M}_1 + 2\mathbf{S}_1 + \mathbf{A}_1 + 2\mathbf{A}'_1$ (SB) | $u = 2 \bmod 4$ |
| $-5$ | $2\mathbf{M}_1 + 4\mathbf{A}_1 + 2\mathbf{A}'_1$ ($\mathbb{C}$) | | $u = 8, 12, 16 \bmod 20$ |
| any | $2\mathbf{M}_1 + 2\mathbf{S}_1 + \mathbf{A}_1 + \mathbf{A}'_1$ (SB) | | $u = 0, 4 \bmod 20$ |

Table 2: Complexities of $\mathbf{S}_2$ depending on the way to build $\mathbb{F}_{p^2}$

**Remark 4.** *The influence of the relative cost of $\mathbf{A}'_1$ compared to $\mathbf{A}_1$ is small and even negligible, even if $-2$ is chosen to define $\mathbb{F}_{p^2}$*

Thanks to Tables 1 and 2, it is easy to choose the best algorithm for $\mathbb{F}_{p^2}$ multiplication and squaring depending on the context (relative cost of $\mathbb{F}_p$ operations, choice of $u$).

## 3.6 Choice of $\mathbb{F}_{p^{12}}$ arithmetic in the case $2, 3, 2$

We assume that $\mathbb{F}_{p^{12}}$ is built over $\mathbb{F}_{p^2}$ via $\mathbb{F}_{p^6}$, using some $\xi$ which is neither a square nor a cube in $\mathbb{F}_{p^2}$:

$$\mathbb{F}_{p^6} = \mathbb{F}_{p^2}[\beta] \text{ where } \beta^3 = \xi \text{ and } \mathbb{F}_{p^{12}} = \mathbb{F}_{p^6}[\gamma] \text{ with } \gamma^2 = \beta.$$

### 3.6.1 $\mathbb{F}_{p^6}$ arithmetic

We saw in Section 3.3.2 that the Karatsuba method becomes interesting when $\mathbf{M}_2 \geq 3\mathbf{A}_2$. Regarding Table 1, this condition is clearly always satisfied. Therefore, the Karatsuba method should always be used for $\mathbb{F}_{p^6}/\mathbb{F}_{p^2}$ multiplications and

$$\mathbf{M}_6 = 6\mathbf{M}_2 + 15\mathbf{A}_2 + 2\mathbf{m}_{2,\xi}.$$

It is also easy to study the difference between the complexities of the schoolbook, the Karastuba and the Chung-Hasan methods and to conclude that the latter is better in each case of Tables 1 and 2. Therefore, the Chung-Hasan method should always be use for $\mathbb{F}_{p^6}/\mathbb{F}_{p^2}$ squaring so

$$\mathbf{S}_6 = 2\mathbf{M}_2 + 3\mathbf{S}_2 + 8\mathbf{A}_2 + \mathbf{A'}_2 + 2\mathbf{m}_{2,\xi}.$$

### 3.6.2 $\mathbb{F}_{p^{12}}$ arithmetic

We obviously have $\mathbf{A}_{12} = 2\mathbf{A}_6 = 12\mathbf{A}_1$ and $\mathbf{A'}_{12} = 2\mathbf{A'}_6 = 12\mathbf{A'}_1$.

**$\mathbb{F}_{p^{12}}$ multiplication.** According to Section 3.2, the Karatsuba method is better than the schoolbook method when $3\mathbf{A}_6 < \mathbf{M}_6$, which is obviously always the case according to 3.6.1. The $\mathbb{F}_{p^6}$ multiplication by $\beta$ involved in Karatsuba formulas is given by

$$\left(b_0 + b_1\beta + b_2\beta^2\right)\beta = \xi b_2 + b_0\beta + b_1\beta^2,$$

thus $\mathbf{m}_{6,\beta} = \mathbf{m}_{2,\xi}$. Finally, the Karatsuba method should always be used for $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^6}$ multiplications and

$$\mathbf{M}_{12} = 18\mathbf{M}_2 + 60\mathbf{A}_2 + 7\mathbf{m}_{2,\xi}.$$

**$\mathbb{F}_{p^{12}}$ sparse multiplication.** During the Miller loop for the optimal Ate pairing, one of the operands (the line $\ell$) is sparse but the conclusion remains the same: the Karatsuba method should be preferred for multiplications at all levels, assuming that at least two coefficients are non-zero. More precisely, as explained in Section 6, $\ell$ is of the form $b_0 + b_1\gamma + b_3\gamma^3$, where $b_i \in \mathbb{F}_{p^2}$. With our notations, it can be written $\ell = b_0 + (b_1 + b_3\beta)\gamma$. The Karatsuba multiplication in $\mathbb{F}_{p^{12}}$ between $\ell$ and $c_0 + c_1\gamma$ is given by

$$b_0c_0 + (b_1 + b_3\beta)c_1\beta + \left((b_0 + b_1 + b_3\beta)(c_0 + c_1) - b_0c_0 - (b_1 + b_3\beta)c_1\right) \qquad (1)$$

and requires

- one multiplication of $c_0 \in \mathbb{F}_{p^6}$ by $b_0 \in \mathbb{F}_{p^2}$ which trivially costs $3\mathbf{M}_2$,
- one multiplication of $c_1 \in \mathbb{F}_{p^6}$ by $b_1 + b_3\beta$. It is done thanks to the sparse Karatsuba multiplication given in Section 3.3 and costs $5\mathbf{M}_2 + 6\mathbf{A}_2 + \mathbf{m}_{2,\xi}$,
- $4\mathbf{A}_2$ to compute $b_0 + b_1$ and $c_0 + c_1$,
- one multiplication of $c_0 + c_1$ by $b_0 + b_1 + b_3\beta$ which is the same as the one of $c_1$ by $b_1 + b_3\beta$,
- one $\mathbf{m}_{6,\beta} = \mathbf{m}_{2,\xi}$ and $3\mathbf{A}_6$ to compute the final result.

Hence a $\mathbb{F}_{p^{12}}$ sparse multiplication requires

$$\mathbf{sM}_{12} = 13\mathbf{M}_2 + 25\mathbf{A}_2 + 3\mathbf{m}_{2,\xi}.$$

$\mathbb{F}_{p^{12}}$ **squaring.** Let us now compare the schoolbook and the Karatsuba methods for squaring in $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^6}$. Using the complexities obtained for $\mathbf{M}_6$ and $\mathbf{S}_6$ in 3.6.1, the difference between the complexities is

$$\mathbf{S}_{12}^{K} - \mathbf{S}_{12}^{SB} = 3\mathbf{S}_2 - 4\mathbf{M}_2 + 2\mathbf{A}_2 - 2\mathbf{A'}_2 + \mathbf{m}_{2,\xi}.$$

Using Tables 1 and 2, we can easily verify that this difference is negative in all the cases we have considered even if $\mathbf{m}_{2,\xi} = \mathbf{M}_2$ (which is obviously the worst case for $\mathbf{m}_{2,\xi}$). Therefore, the schoolbook method should not be used.

We have now to compare the Karatsuba and the complex methods for squaring in $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^6}$. Using the complexities obtained for $\mathbf{M}_6$ and $\mathbf{S}_6$ in 3.6.1 and assuming that $\mathbf{m}_{6,\beta+1} = \mathbf{A}_6 + \mathbf{m}_{6,\beta} = \mathbf{A}_6 + \mathbf{m}_{2,\xi}$, the difference between the complexities is

$$\Delta = \mathbf{S}_{12}^{K} - \mathbf{S}_{12}^{\mathbb{C}} = 9\mathbf{S}_2 - 6\mathbf{M}_2 - 6\mathbf{A}_2 + \mathbf{m}_{2,\xi}$$

The sign of $\Delta$ depends on the value of $\mu$. Again, we do not give all the details but thanks to Tables 1 and 2, we can determine that

- The Karatsuba method should be used if $\mu = -1, -5$ and if $\mu = -2$ with $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$ or $\mathbf{S}_1 = 0.8\mathbf{M}_1$ and in this case

  $$\mathbf{S}_{12} = 6\mathbf{M}_2 + 9\mathbf{S}_2 + 36\mathbf{A}_2 + 3\mathbf{A'}_2 + 7\mathbf{m}_{2,\xi}.$$

- The complex method should be used if $\mu$ is not small and if $\mu = -2$ with $\mathbf{A}_1 > 0.33\mathbf{M}_1$ and $\mathbf{S}_1 = \mathbf{M}_1$ and in this case

  $$\mathbf{S}_{12} = 12\mathbf{M}_2 + 42\mathbf{A}_2 + 3\mathbf{A'}_2 + 6\mathbf{m}_{2,\xi}.$$

## 3.7 Choice of $\mathbb{F}_{p^{12}}$ arithmetic in the case $2, 2, 3$

We assume that $\mathbb{F}_{p^{12}}$ is built over $\mathbb{F}_{p^2}$ via $\mathbb{F}_{p^4}$, using some $\xi$ which is neither a square nor a cube in $\mathbb{F}_{p^2}$:

$$\mathbb{F}_{p^4} = \mathbb{F}_{p^2}[\beta] \text{ where } \beta^2 = \xi \text{ and } \mathbb{F}_{p^{12}} = \mathbb{F}_{p^4}[\gamma] \text{ with } \gamma^3 = \beta.$$

### 3.7.1 $\mathbb{F}_{p^4}$ arithmetic

We saw in Section 3.2.2 that the Karatsuba method becomes interesting when $\mathbf{M}_2 \geq 3\mathbf{A}_2 = 6\mathbf{A}_1$. Regarding Table 1, this condition is clearly always satisfied.

Therefore, the Karatsuba method should always be used for $\mathbb{F}_{p^4}/\mathbb{F}_{p^2}$ multiplications and

$$\mathbf{M}_4 = 3\mathbf{M}_2 + 5\mathbf{A}_2 + \mathbf{m}_{2,\xi}.$$

Concerning the squaring, all the methods are close in terms of complexity. We do not give details here, but thanks to Tables 1 and 2, we can decide which method is preferable to use, depending on the context. The Karatsuba method is usually the best and this is summarized in Table 3.

| $\mu$ | condition | method | complexity |
|---|---|---|---|
| assuming $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$ | | | |
| $-1, -2$ or $-5$ | | K | $3\mathbf{S}_2 + \mathbf{m}_{2,\xi} + 4\mathbf{A}_2$ |
| any | | $\mathbb{C}$ | $2\mathbf{M}_2 + \mathbf{m}_{2,\xi} + \mathbf{m}_{2,\xi+1} + 3\mathbf{A}_2 + \mathbf{A'}_2$ |
| assuming $\mathbf{A}_1 > 0.33\mathbf{M}_1$ | | | |
| $-1$ or $-5$ | | K | $3\mathbf{S}_2 + \mathbf{m}_{2,\xi} + 4\mathbf{A}_2$ |
| $-2$ | $\mathbf{S}_1 = 0.8\mathbf{M}_1$ | | |
| any | | SB | $\mathbf{M}_2 + 2\mathbf{S}_2 + \mathbf{m}_{2,\xi} + \mathbf{A}_2 + \mathbf{A'}_2$ |
| $-2$ or any | $\mathbf{S}_1 = \mathbf{M}_1$ | $\mathbb{C}$ | $2\mathbf{M}_2 + \mathbf{m}_{2,\xi} + \mathbf{m}_{2,\xi+1} + 3\mathbf{A}_2 + \mathbf{A'}_2$ |

Table 3: Complexities of $\mathbf{S}_4$ depending on the context

### 3.7.2 $\mathbb{F}_{p^{12}}$ arithmetic

We also have $\mathbf{A}_{12} = 12\mathbf{A}_1$ and $\mathbf{A'}_{12}12\mathbf{A'}_1$ in this case.

**$\mathbb{F}_{p^{12}}$ multiplication.** According to Section 3.3, the Karatsuba method is better than the schoolbook method for multiplications in $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^4}$ when $3\mathbf{A}_4 < \mathbf{M}_4$, which is obviously always the case according to Section 3.7.1. The $\mathbb{F}_{p^4}$ multiplication by $\beta$ involved in Karatsuba formulas is given by

$$(b_0 + b_1\beta)\,\beta = \xi b_1 + b_0\beta,$$

thus $\mathbf{m}_{4,\beta} = \mathbf{m}_{2,\xi}$. Therefore, the Karatsuba method should always be used for $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^4}$ multiplications and

$$\mathbf{M}_{12} = 18\mathbf{M}_2 + 60\mathbf{A}_2 + 8\mathbf{m}_{2,\xi}.$$

**$\mathbb{F}_{p^{12}}$ sparse multiplication.** Again, the Karatsuba method should be preferred for multiplications at all levels when one of the operands is sparse. With our notations, the sparse element involved in the Miller loop can be written $\ell = b_0 + b_3\beta + b_1\gamma$. We use the sparse Karatsuba multiplication given in Section 3.3 to compute the product of $\ell$ and $c_0 + c_1\gamma + c_2\gamma^2 \in \mathbb{F}_{p^{12}}$

$$\begin{aligned}(c_0 + c_1\gamma + c_2\gamma^2)((b_0 + b_3\beta) + b_1\gamma) = {} & c_0(b_0 + b_3\beta) + \beta c_2 b_1 \\ & + [(c_0 + c_1)(b_0 + b_3\beta + b_1) - c_0(b_0 + b_3\beta) - c_1 b_1]\gamma \\ & + [c_2(b_0 + b_3\beta) + c_1 b_1]\gamma^2 \end{aligned} \tag{2}$$

It requires

- Two multiplications of $c_1$ (or $c_2$) $\in \mathbb{F}_{p^4}$ by $b_1 \in \mathbb{F}_{p^2}$ which trivially cost $2\mathbf{M}_2$ each.
- Three $\mathbb{F}_{p^4}$ multiplications which are done thanks to the Karatsuba method and cost $3\mathbf{M}_2 + 5\mathbf{A}_2 + \mathbf{m}_{2,\xi}$.
- $3\mathbf{A}_2$ to compute $b_0 + b_1$ and $c_0 + c_1$.

- One $\mathbf{m}_{4,\beta} = \mathbf{m}_{2,\xi}$ and $4\mathbf{A}_4$ to compute the final result.

Hence a $\mathbb{F}_{p^{12}}$ sparse multiplication requires

$$\mathbf{sM}_{12} = 13\mathbf{M}_2 + 26\mathbf{A}_2 + 4\mathbf{m}_{2,\xi}.$$

**$\mathbb{F}_{p^{12}}$ squaring.** We do not give details here because it would be repetitive after previous sections but it is easy to study the difference between the complexities of the schoolbook, the Karastuba and the Chung-Hasan methods and to conclude that the latter is always better. Therefore, the Chung-Hasan method should always be used for $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^4}$ squaring and

$$\mathbf{S}_{12} = 3\mathbf{S}_4 + 6\mathbf{M}_2 + 26\mathbf{A}_2 + 2\mathbf{A'}_2 + 4\mathbf{m}_{2,\xi}.$$

### 3.8 Choice of $\xi$ and consequences on $u$

The choice of $\xi$ has no consequence on the way of choosing the arithmetic of $\mathbb{F}_{p^{12}}$, but it must be done such that multiplication by $\xi$ is as efficient as possible. However, the choice of $\xi$ has less influence on $\mathbf{M}_{12}$ and $\mathbf{S}_{12}$ complexities than the choice of $\mu$. So an efficient $\mu$ should be chosen in priority.

The most natural choice is $\xi = \alpha$, so that a multiplication by $\xi$ in $\mathbb{F}_{p^2}$ is

$$(a_0 + a_1\alpha)\xi = \mu a_1 + a_0\alpha$$

and thus $\mathbf{m}_{2,\xi} = \mathbf{m}_{1,\mu}$. It means that $\mu$ must be neither a fourth power nor a cube in $\mathbb{F}_p$. If $p \equiv 1 \pmod 6$, which is always the case for a BN prime, such a $\mu$ always exists.

In practice, it is easy to find one by trial and error, by factoring $x^{12} - \mu$ modulo $p$ until finding an irreducible polynomial. However, the obtained value is not necessarily optimal for $\mathbb{F}_{p^2}$ arithmetic and $\mu$ should be chosen in priority to define $\mathbb{F}_{p^2}$ as efficiently as possible. For example, the best choice for $\mathbb{F}_{p^2}$ is $\mu = -1$ but it is obviously always a cube. Then, if $\mu$ is already fixed to define $\mathbb{F}_{p^2}$ and is a cube, $\xi$ must be chosen as an element of $\mathbb{F}_{p^2}$ with small coefficients. At first glance, the best choices are $\xi = 2$ and $\xi = 1 + \alpha$. However, as proved in Corollary 1 of Appendix A, 2 (as any other element of $\mathbb{F}_p$) is always a square in $\mathbb{F}_{p^2}$, so it cannot be chosen to define $\mathbb{F}_{p^{12}}$. Other choices can be made as $2 + \alpha$ but they are of course less interesting in terms of efficiency.

**Remark 5.** *The condition to choose $\xi$ is that it is neither a square nor a cube in $\mathbb{F}_{p^2}$. Since $-1$ is always a square and a cube in $\mathbb{F}_{p^2}$, there is no interest to consider the sign of $\xi$. In the same way, there is no interest to consider the conjugates of $\xi$ because they are squares and cubes at the same times as $\xi$.*

Let us now precise the possible choices for $\xi$ depending on the choice of $\mu$ and their consequences on the choice of the parameter $u$. We will use the following proposition which is proved in Appendix A:

**Proposition 3.** *Let $p$ be an odd prime number which is equal to 1 modulo 3 (this is always the case for BN primes). Then an element is a square (resp. a cube) in $\mathbb{F}_{p^2}$ if and only if its norm is a square (resp. a cube) in $\mathbb{F}_p$.*

### 3.8.1 The case $\mu = -1$

In this case, the smallest complexity for $\mathbf{m}_{2,\xi}$ is reached by $\xi = 1 + \mathbf{i}$ and equals $2\mathbf{A}_1$ since

$$(1 + \mathbf{i})(a_0 + a_1\mathbf{i}) = a_0 - a_1 + (a_0 + a_1)\mathbf{i}.$$

Let us now study the conditions on $u$ ensuring that $1 + \mathbf{i}$ is neither a square nor a cube in $\mathbb{F}_{p^2}$.

**Theorem 1.** *Let $p = 36u^4 + 36u^3 + 24u^2 + 6u + 1$ be a BN prime with $u$ odd (thus $\mathbb{F}_{p^2}$ can be defined by $\mathbf{i} = \sqrt{-1}$). Then $1 + \mathbf{i}$ is neither a square nor a cube in $\mathbb{F}_{p^2}$ if and only if $u = 7$ or $11$ modulo $12$. In this case, $\mathbb{F}_{p^{12}}$ can be defined over $\mathbb{F}_{p^2}$ by a sixth root of $1 + \mathbf{i}$.*

*Proof.* By Proposition 3, $1 + \mathbf{i}$ is a square (resp. a cube) in $\mathbb{F}_{p^2}$ if and only if $2 = N_{\mathbb{F}_{p^2}/\mathbb{F}_p}(1 + \mathbf{i})$ is a square (resp. a cube) in $\mathbb{F}_p$.

- According to Proposition 1 of Appendix A, 2 is a square in $\mathbb{F}_p$ if and only if $u = 0$ or $1$ modulo 4. Thus, in our situation, $1 + \mathbf{i}$ is not a square in $\mathbb{F}_{p^2}$ if and only if $u = 3$ modulo 4.

- In the same way, $1 + \mathbf{i}$ is not a cube if and only if 2 is not a cube which means, according to Proposition 2 of Appendix A, that $u \neq 0$ mod 3. Thus $1 + \mathbf{i}$ is not a cube if and only if $u = 1$ or $5$ modulo 6.

Combining these two congruences ends the proof. □

If $1+\mathbf{i}$ cannot be chosen, $\xi$ should be sought in the form $a+\mathbf{i}b$ with $a > b > 0$ and can be chosen if and only if $a^2 + b^2 = N_{\mathbb{F}_{p^2}/\mathbb{F}_p}(a+\mathbf{i}b)$ is neither a square nor a cube in $\mathbb{F}_p$. The next candidate in terms of efficiency is $\xi = 2 + \mathbf{i}$ for which $\mathbf{m}_{2,\xi} = 2\mathbf{A}_1 + 2\mathbf{A'}_1$ and whose norm is 5. Combining the condition ensuring that 5 is neither a square nor a cube given in Appendix A and the fact that $u \neq 7$ or $11$ modulo $12$ (otherwise $\xi = 1 + \mathbf{i}$ can be chosen), we get that $2 + \mathbf{i}$ should be chosen to define $\mathbb{F}_{p^{12}}$ if $u = 1, 3, 13, 27, 33, 37, 41$ or $57$ modulo $60$ but not in the 12 remaining cases ($u = 5, 9, 15, 17, 21, 25, 29, 39, 45, 49, 51$ or $53$ modulo 60). In 3 of these cases ($u = 17, 39$ and $53$ modulo 60), one can choose $3 + \mathbf{i}$, whose norm is 10. Of course, this study can be easily continued with "larger" values of $\xi$ $(4 + \mathbf{i}, 3 + 2\mathbf{i}, ...)$ to be able to deal with the 9 remaining cases. But the interest is limited here and the reader can easily do it by himself if necessary.

### 3.8.2 The case $\mu = -2$

Taking in account the results from Section 3.5, this choice should be made only when $-1$ cannot be chosen to build $\mathbb{F}_{p^2}$, that is, when $u = 2$ modulo 4 ($-1$ is a square and $-2$ is not a square). In this case, the smallest complexity for $\mathbf{m}_{2,\xi}$ is reached by $\xi = \alpha = \sqrt{-2}$. Let us now study the conditions on $u$ ensuring that $\sqrt{-2}$ is neither a square nor a cube in $\mathbb{F}_{p^2}$.

**Theorem 2.** *Let $p = 36u^4 + 36u^3 + 24u^2 + 6u + 1$ be a BN prime with $u = 2$ modulo 4 (thus $\mathbb{F}_{p^2}$ can be defined by $\sqrt{-2}$ but not by $\sqrt{-1}$). Then $\alpha = \sqrt{-2}$ is neither a square nor a cube in $\mathbb{F}_{p^2}$ if and only if $u = 2$ or 10 modulo 12. In this case, $\mathbb{F}_{p^{12}}$ can be defined over $\mathbb{F}_{p^2}$ by a sixth root of $\alpha$.*

*Proof.*

- Since $-1$ is a square and $-2$ is not, $N_{\mathbb{F}_{p^2}/\mathbb{F}_p}(\alpha) = 2$ is never a square in $\mathbb{F}_p$ and thus, by 3, $\alpha$ is never a square in $\mathbb{F}_{p^2}$. Note that this proof is not specific to the case $\mu = -2$; it is only using the fact that $-1$ is a square in $\mathbb{F}_p$ and $\mu$ is not.

- Since $-1$ is a cube, $\alpha$ is a cube in $\mathbb{F}_{p^2}$ if and only if $2 = -N_{\mathbb{F}_{p^2}/\mathbb{F}_p}(\alpha)$ is a cube in $\mathbb{F}_p$. According to Proposition 2 of Appendix A, 2, and thus $\alpha$, is not a cube if and only if $u = 1$ or 2 modulo 3. Taking in account that $-1$ is a square and $-2$ is not a square, this last condition can be rewritten $u = 2$ or 10 modulo 12.

□

The next interesting candidate is $\xi = 1 + \alpha$ but it cannot be chosen since its norm equals 3 which is always a square in $\mathbb{F}_p$ in the remaining case ($u = 6$ modulo 12). The situation is better if $\xi = 2 + \alpha$ whose norm is 6. Indeed, since 2 is not a square and 3 is always a square, 6 is never a square in $\mathbb{F}_p$. Moreover, since it was not possible to choose $\xi = \alpha$, 2 is a cube in $\mathbb{F}_p$ and thus 6 is not a cube in $\mathbb{F}_p$ if and only if 3 is not a cube in $\mathbb{F}_p$. According to Proposition 2 of Appendix A, $\xi = 2 + \alpha$ can then be chosen if $u = 6$ or 30 modulo 36. In this case $\mathbf{m}_{2,\xi} = 2\mathbf{A}_1 + 2\mathbf{A}'_1$. Again, this study can be easily continued by the interested reader with "larger" values of $\xi$ like $3 + \alpha$ to deal with the last remaining case ($u = 18$ modulo 36).

### 3.8.3 The case $\mu = -5$

Again, taking in account the results from Section 3.5, this case should be considered only if $u = 8, 12$ or 16 modulo 20 ($-1, \pm 2$ are squares and $\pm 5$ are not). The smallest complexity for $\mathbf{m}_{2,\xi}$ is reached by $\xi = \alpha = \sqrt{-5}$. Let us study the conditions on $u$ ensuring that $\sqrt{-5}$ is neither a square nor a cube in $\mathbb{F}_{p^2}$.

**Theorem 3.** *Let $p = 36u^4 + 36u^3 + 24u^2 + 6u + 1$ be a BN prime with $u = 8, 12$ or 16 modulo 20. Then $\alpha = \sqrt{-5}$ is neither a square nor a cube in $\mathbb{F}_{p^2}$ if and only if $u = 12, 16, 28, 48, 52, 56$ modulo 60. In this case, $\mathbb{F}_{p^{12}}$ can be defined over $\mathbb{F}_{p^2}$ by a sixth root of $\alpha$.*

*Proof.* Similarly to the proof of Theorem 2, we prove that $\alpha$ is never a square in $\mathbb{F}_{p^2}$. Moreover, $\alpha$ is a cube in $\mathbb{F}_{p^2}$ if and only if $5 = -N_{\mathbb{F}_{p^2}/\mathbb{F}_p}(\alpha)$ is a cube in $\mathbb{F}_p$. Proposition 2 of Appendix A gives a condition ensuring that 5, and thus $\alpha$, is not a cube and allows to conclude. □

The next interesting candidate is $\xi = 1 + \alpha$ but it cannot be chosen since its norm equals 6 which is always a square in this case. The situation is the same for $2 + \alpha$ since its norm is 9. The "smallest" usable candidates are $1 + 2\alpha$ and $3 + \alpha$ whose norms are respectively 21 and 14. This will give conditions on $u$ modulo 7 but we leave it to the interested reader, since there are very few concerned cases ($u = 8, 32$ or 36 modulo 60) and the cost of $\mathbf{m}_{2,\xi}$ becomes high.

### 3.8.4 Recapitulative table

Finally, we gave almost all the best choices for $\xi$, depending on the value of $u$. This is summarized in Table 4.

| $\mu$ | Value of $u$ modulo | | | | | $\xi$ | $\mathbf{m}_{2,\xi}$ |
|---|---|---|---|---|---|---|---|
| | 4 | 12 | 20 | 36 | 60 | | |
| $-1$ | $1, 3$ | $7, 11$ | | | | $1 + \mathbf{i}$ | $2\mathbf{A}_1$ |
| | | $1, 3, 5, 9$ | | | $1, 3, 13, 27, 33, 37, 41, 57$ | $2 + \mathbf{i}$ | $2\mathbf{A}_1 + 2\mathbf{A'}_1$ |
| | | | | | $17, 39, 53$ | $3 + \mathbf{i}$ | $4\mathbf{A}_1 + 2\mathbf{A'}_1$ |
| | | | | | $5, 9, 15, 21, 25, 29, 45, 49, 51$ | – | – |
| $-2$ | $2$ | $2, 10$ | | | | $\sqrt{-2}$ | $\mathbf{A'}_1$ |
| | | $6$ | | $6, 30$ | | $2 + \sqrt{-2}$ | $2\mathbf{A}_1 + 2\mathbf{A'}_1$ |
| | | | | $18$ | | – | – |
| $-5$ | $0$ | $0, 4, 8$ | $8, 12, 16$ | | $12, 16, 28, 48, 52, 56$ | $\sqrt{-5}$ | $\mathbf{A}_1 + 2\mathbf{A'}_1$ |
| | | | | | $8, 32, 36$ | – | – |
| any | | | $0, 4$ | | | $\sqrt{\mu}$ | $\mathbf{M}_1$ |

Table 4: Best choices of $\mu$ and $\xi$ in terms of $u$

## 3.9 New improvements of $\mathbb{F}_{p^{12}}$ arithmetic

Usually, the cost of additions is not taken in account in complexity studies because it is assumed to be negligible. However, it is not always the case in hardware or assembly language where the ratio between additions and multiplications in $\mathbb{F}_p$ can be as large as 0.2 to 0.5, as explained in Section 3.1. In this case, we get some new improvements (saving of course only additions). The first one is just a computational trick and is probably used in practical implementations but it is usually not explicitly written. The second one consists in precomputing the traces of some elements of the intermediate levels of the extension tower which are involved in several operations.

Since these are new results, we chose to present them in a dedicated section. These results could be included in our previous discussion on the choices for building $\mathbb{F}_{p^{12}}$. We did not do it for two reasons. The first one is that they require some precomputations storage which is not always desirable, depending on the context. The second one is that it has no influence on the choices to be made.

### 3.9.1 Multiplications by $\xi - 1$ in Karatsuba operations

If $\mathbf{m}_{i,\xi-1} \leq \mathbf{m}_{i,\xi}$, then the Karatsuba multiplication of $x_0 + x_1\beta$ by $y_0 + y_1\beta$ in $\mathbb{F}_{p^{2i}}$ can be evaluated as

$$x_0y_0 + x_1y_1 + (\xi - 1)x_1y_1 + ((x_0 + x_1)(y_0 + y_1) - x_0y_0 - x_1y_1)\beta,$$

instead of

$$x_0y_0 + \xi x_1y_1 + ((x_0 + x_1)(y_0 + y_1) - x_0y_0 - x_1y_1)\beta.$$

Then it requires $3\mathbf{M}_i + \mathbf{m}_{i,\xi-1} + 5\mathbf{A}_i$ instead of $3\mathbf{M}_i + \mathbf{m}_{i,\xi} + 5\mathbf{A}_i$ because $x_0y_0 + x_1y_1$ is used twice.

In the same way, the Karatsuba multiplication of $x_0 + x_1\beta + x_2\beta^2$ by $y_0 + y_1\beta + y_0\beta^2$ in $\mathbb{F}_{p^{3i}}$ can be evaluated as

$$\begin{aligned} & x_0y_0 + \xi((x_1 + x_2)(y_1 + y_2) - x_1y_1 - x_2y_2) \\ & + [(x_0 + x_1)(y_0 + y_1) - x_0y_0 - x_1y_1 + x_2y_2 + (\xi - 1)x_2y_2]\beta \\ & + [(x_0 + x_2)(y_0 + y_2) - x_0y_0 - (-x_1y_1 + x_2y_2)]\beta^2. \end{aligned}$$

One of the multiplications by $\xi$ is then replaced by a multiplication by $\xi - 1$ compared to the formula given in 3.3.2. Of course, both in the case of quadratic and cubic extensions, this trick also applies to Karatsuba squaring.

In the cases considered in Section 3.8, this improvement is only interesting in the intermediate fields ($\mathbb{F}_{p^4}/\mathbb{F}_{p^2}$ in the $2, 2, 3$ case and $\mathbb{F}_{p^6}/\mathbb{F}_{p^2}$ in the $2, 3, 2$ case) and allows to save some $\mathbb{F}_p$ additions for each Karatsuba multiplication or squaring in $\mathbb{F}_{p^4}$ or $\mathbb{F}_{p^6}$ in the cases given in Table 5.

| $\xi$ | $m_{2,\xi-1}$ | $m_{2,\xi}$ | Saving |
|---|---|---|---|
| $1 + \mathbf{i}$ | $0$ | $2\mathbf{A}_1$ | $2\mathbf{A}_1$ |
| $2 + \mathbf{i}$ | $2\mathbf{A}_1$ | $2\mathbf{A}_1 + 2\mathbf{A'}_1$ | $2\mathbf{A'}_1$ |
| $3 + \mathbf{i}$ | $2\mathbf{A}_1 + 2\mathbf{A'}_1$ | $4\mathbf{A}_1 + 2\mathbf{A'}_1$ | $2\mathbf{A}_1$ |
| $2 + \sqrt{-2}$ | $2\mathbf{A}_1 + \mathbf{A'}_1$ | $2\mathbf{A}_1 + 2\mathbf{A'}_1$ | $\mathbf{A'}_1$ |

Table 5: Savings provided by the $\xi - 1$ trick

**Remark 6.** *This trick is more interesting in the case $2, 2, 3$ that in the case $2, 3, 2$. Indeed, a Karatsuba multiplication in $\mathbb{F}_{p^{12}}$ requires $6$ multiplications at the middle level in the case $2, 2, 3$ (and then $6m_{2,\xi}$ are replaced by $6m_{2,\xi-1}$) but only $3$ in the case $2, 3, 2$. Of course, this remark also applies to sparse $\mathbb{F}_{p^{12}}$ multiplications and to $\mathbb{F}_{p^{12}}$ squarings.*

### 3.9.2 Precomputed traces

In the case where memory is large enough, we can perform some pre-computations to speed up the different operations (*i.e.* $\mathbf{M}_i$ or $\mathbf{S}_i$). In particular, it could be interesting to precompute the trace of an element if this element is used in several operations (requiring its trace) in the following.

For example, assuming that $x_0 + x_1 = tr_{\mathbb{F}_{p^{2i}}/\mathbb{F}_{p^i}}(x_0 + x_1\alpha)$ is precomputed, computing $(x_0 + x_1\alpha)^2$ in $\mathbb{F}_{p^{2i}}$ using the Karatsuba method as in 3.2.3 requires only $3\mathbf{S}_i + \mathbf{m}_{i,\mu} + 3\mathbf{A}_i$ instead of $3\mathbf{S}_i + \mathbf{m}_{i,\mu} + 4\mathbf{A}_i$. The same remark holds for all the methods for squaring or multiplying in $\mathbb{F}_{p^{2i}}$ or $\mathbb{F}_{p^{3i}}$ involving traces (or more precisely sums of coordinates). It is summarized in Table 6 (notations are those of Sections 3.2 and 3.3).

| Operation | Method | Precomputations | Saving |
|---|---|---|---|
| $(x_0 + x_1\alpha)(y_0 + y_1\alpha)$ | $\mathbf{M}_{2i}^{K}$ | $x_0 + x_1, y_0 + y_1$ | $2\mathbf{A}_i$ |
| $(x_0 + x_1\alpha)^2$ | $\mathbf{S}_{2i}^{K}$ or $\mathbf{S}_{2i}^{\mathbb{C}}$ | $x_0 + x_1$ | $\mathbf{A}_i$ |
| $(x_0 + x_1\xi + x_2\xi^2)(y_0 + y_1\xi + y_2\xi^2)$ | $\mathbf{M}_{3i}^{K}$ | $x_0 + x_1, x_0 + x_2, x_1 + x_2$<br>$y_0 + y_1, y_0 + y_2, y_1 + y_2$ | $6\mathbf{A}_i$ |
| $(x_0 + x_1\xi + x_2\xi^2)^2$ | $\mathbf{S}_{3i}^{K}$ | $x_0 + x_1, x_0 + x_2, x_1 + x_2$ | $3\mathbf{A}_i$ |
|  | $\mathbf{S}_{3i}^{CH}$ | $x_0 + x_1 + x_2$ | $2\mathbf{A}_i$ |

Table 6: Precomputing traces

Of course, we get savings only if the precomputations are used in several operations. This may hold in the arithmetic of the extension tower when an element is used twice. For example, a schoolbook multiplication in $\mathbb{F}_{p^{2i}}$ will use $x_0$ both for $x_0y_0$ and for $x_0y_1$. If these multiplications are performed with the Karatsuba method in $\mathbb{F}_{p^i}$, precomputing the trace of $x_0$ is interesting because it is used twice. However, in Karatsuba operations or in complex squarings, no element is used twice. As a consequence, this is not interesting for multiplications in $\mathbb{F}_{p^{12}}$ because the Karatsuba method is always used at higher levels of the extension tower according to Sections 3.6 and 3.7. But it can be applied for squarings when the Chung-Hasan method is used over Karatsuba or complex arithmetic. It can also be applied for sparse multiplications since it involves schoolbook steps if operands have only one non-zero coefficient. Finally, we can also precompute traces if one $\mathbb{F}_{p^{12}}$ element is used for several multiplications, which is usually the case in the final exponentiation step. Let us now give more details about these three situations.

### 3.9.3 Use of precomputed traces in $\mathbb{F}_{p^{12}}$ squarings

Some coefficients are used several times in $\mathbb{F}_{p^{12}}$ squaring if the Chung-Hasan method is used over Karatsuba or complex arithmetic (*i.e.* when $\mathbf{A}_1$ is not very expensive compared to $\mathbf{M}_1$). Indeed, we know from Section 3.3 that $x_0, 2x_1$ and $x_2$ are used twice to compute $(x_0 + x_1\alpha + x_2\alpha^2)^2$ using the Chung-Hasan method in $\mathbb{F}_{p^{3i}}/\mathbb{F}_{p^i}$. Thus $3\mathbf{A}_i$ can be saved in $\mathbf{S}_{3i}^{CH}$ if traces are precomputed at the level $i$. We saw in Sections 3.6 and 3.7 that this method is always used but one can do even better, depending on the way to build $\mathbb{F}_{p^{12}}$.

**Case 2,3,2.** We saw in Section 3.6 that the Karatsuba method is usually used for the $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^6}$ squaring, so that the Chung-Hasan squaring in $\mathbb{F}_{p^6}$ is used 3 times to compute $(c_0 + c_1\gamma)^2$ (for $c_0^2, c_1^2$ and $(c_0 + c_1)^2$). Of course each of them is used only once but precomputing traces is nonetheless interesting because $tr_{\mathbb{F}_{p^6}/\mathbb{F}_{p^2}}(c_0 + c_1) = tr_{\mathbb{F}_{p^6}/\mathbb{F}_{p^2}}(c_0) + tr_{\mathbb{F}_{p^6}/\mathbb{F}_{p^2}}(c_1)$ which requires only one $\mathbf{A}_2$ instead of 2 if it is computed directly. Hence, in this case, if the Karatsuba method is used in $\mathbb{F}_{p^2}$, $11\mathbf{A}_1$ can be saved in $\mathbf{S}_{12}$ thanks to trace precomputations (9 from $\mathbb{F}_{p^6}/\mathbb{F}_{p^2}$ Chung-Hasan over Karatsuba squaring and 2 from $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^6}$ Karatsuba over Chung-Hasan squaring). If the Karatsuba method is not used in $\mathbb{F}_{p^2}$, which means that $\mathbf{A}_1 > 0.33\mathbf{M}_1$, only $2\mathbf{A}_1$ can be saved (from $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^6}$ Karatsuba over Chung-Hasan squaring).

**Case 2,2,3.** We saw that when we compute $(c_0 + c_1\gamma + c_2\gamma^2)^2$ with the Chung-Hasan method $c_0, c_1$ and $c_2 \in \mathbb{F}_{p^4}$ are used twice. For example $c_0$ is used in $c_0^2$ and in $2c_0c_1$, so that precomputing $t_0 = tr_{\mathbb{F}_{p^4}/\mathbb{F}_{p^2}}(c_0)$ saves one $\mathbf{A}_2$. But, assuming that $c_0 = b_0 + b_1\beta$, $b_0 \in \mathbb{F}_{p^2}$ is also used twice (also in $c_0^2$ and in $2c_0c_1$ whatever the methods used) and precomputing its trace is then interesting if the Karatsuba/complex method is used in $\mathbb{F}_{p^2}$. Moreover for $\mathbb{F}_{p^4}$ Karatsuba or complex operations, $t_0$ plays the same role as $b_0$ so that precomputing its trace is also interesting. Finally, in this case and if the Karatsuba method is used in $\mathbb{F}_{p^2}$, $15\mathbf{A}_1$ can be saved in $\mathbf{S}_{12}$ thanks to trace precomputations (6 from the $\mathbb{F}_{p^4}/\mathbb{F}_{p^2}$ traces of the $c_i$, 6 from the $\mathbb{F}_{p^2}/\mathbb{F}_p$ traces of the $\mathbb{F}_{p^2}$ components of the $c_i$ and 3 from the $\mathbb{F}_{p^2}/\mathbb{F}_p$ traces of the $\mathbb{F}_{p^4}/\mathbb{F}_{p^2}$ traces of the $c_i$). If the Karatsuba method is not used in $\mathbb{F}_{p^2}$, which means that $\mathbf{A}_1 > 0.33\mathbf{M}_1$, only $6\mathbf{A}_1$ can be saved (from the $\mathbb{F}_{p^4}/\mathbb{F}_{p^2}$ traces of the $c_i$).

### 3.9.4 Use of precomputed traces in $\mathbb{F}_{p^{12}}$ sparse multiplications

The sparse multiplication involved in the Miller loop for the optimal Ate pairing involves schoolbook steps if operands have only one non-zero coefficient. Again, the savings are depending on the way to build $\mathbb{F}_{p^{12}}$

**Case 2,3,2.** Looking at formula (1) given in Section 3.6.2, we can see that

- $b_0$ is used in 3 $\mathbb{F}_{p^2}$ multiplications. Precomputing its trace then saves $2\mathbf{A}_1$,
- $b_1$ and the third component of $c_1$ are used in 2 $\mathbb{F}_{p^2}$ multiplications during the sparse product $(b_1 + b_3\beta)c_1$, so $2\mathbf{A}_1$ can be saved,
- The same holds for the sparse product $(b_0 + b_1 + b_3\beta)(c_0 + c_1)$,
- $b_3$ is used twice in each of these sparse products, so $3\mathbf{A}_1$ can be saved by precomputing $tr_{\mathbb{F}_{p^2}/\mathbb{F}_p}(b_3)$.

Finally, $9\mathbf{A}_1$ can be saved in the sparse multiplication if traces are precomputed (assuming that the Karatsuba method is used for $\mathbb{F}_{p^2}$ multiplications, *i.e.* that $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$)

**Case** $2, 2, 3$. Looking at formula (2) given in Section 3.7.2, we can see that

- $b_0 + b_3\beta$ is used in 2 $\mathbb{F}_{p^4}$ multiplication. Precomputing its trace $(b_0 + b_3)$ saves $\mathbf{A}_2$,
- As a consequence of the previous point, $b_0, b_3$ and $b_0 + b_3$ are used in 2 $\mathbb{F}_{p^2}$ multiplications, so $3\mathbf{A}_1$ can be saved,
- $b_3$ is also used in the $\mathbb{F}_{p^4}$ product $(c_0 + c_1)(b_0 + b_1 + b_3\beta)$ so one additional $\mathbf{A}_1$ can be saved,
- $b_1$ is used in 4 $\mathbb{F}_{p^2}$ multiplications ($c_2b_1$ and $c_1b_1$) so $3\mathbf{A}_1$ can be saved by precomputing $tr_{\mathbb{F}_{p^2}/\mathbb{F}_p}(b_1)$,
- $c_2$ is used twice (in $c_2b_1$ and in $c_2(b_0 + b_3\beta)$) so its $\mathbb{F}_{p^2}$ coefficients are used twice each which saves $2\mathbf{A}_1$.

Finally, $11\mathbf{A}_1$ can be saved in the sparse multiplication if traces are precomputed (assuming that the Karatsuba method is used for $\mathbb{F}_{p^2}$ multiplications, *i.e.* that $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$, otherwise only $2\mathbf{A}_1$ are saved).

In all considered cases, the saving obtained is around 10% of the total number of additions in $\mathbb{F}_{p^{12}}$ operations which is not negligible if the relative cost of an addition compared to a multiplication in $\mathbb{F}_p$ is not small.

### 3.9.5 Use of precomputed traces in the final exponentiation

Full multiplications in $\mathbb{F}_{p^{12}}$ are only used in the final exponentiation (in the Miller loop, sparse multiplications are used). If the implemented exponentiation parses the exponent from left to right (which is usually the case), then the multiplication steps are performed with one constant term $c$. Hence, we can precompute and store all the traces depending only on $c$. Since the Karatsuba method is used at all levels of the extension tower (except in $\mathbb{F}_{p^2}$ if $\mathbf{A}_1 > 0.33\mathbf{M}_1$), we will significantly reduce the number of required additions, whatever the way to build $\mathbb{F}_{p^{12}}$.

**Case 2,3,2.**

- $\mathbf{M}_{12}^{K}$ requires the $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^6}$ trace of $c$, thus one $\mathbf{A}_6$ can be saved if this trace is already precomputed. It also requires $3\mathbf{M}_6^{K}$ with one constant term (the 2 coordinates of $c$ and its trace).
- Each $\mathbf{M}_6^{K}$ involving a constant term $b$ requires 3 sums of 2 coordinates of $b$, thus $3\mathbf{A}_2$ can be saved if these sums are precomputed. Hence $9\mathbf{A}_2$ are saved at this level. Each $\mathbf{M}_6^{K}$ also requires $6\mathbf{M}_2$ with one constant term (the 3 coordinates of $b$ and the 3 sums of 2 coordinates).

- Each $\mathbf{M}_2^K$ involving a constant term $a$ requires the $\mathbb{F}_{p^2}/\mathbb{F}_p$ trace of $a$, thus one $\mathbf{A}_1$ can be saved if this trace is precomputed. Hence $18\mathbf{A}_1$ can be saved at this level when the Karatsuba method is used for $\mathbf{M}_2$ (*i.e.* if $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$ according to Section 3.5).

**Case 2,2,3.**

- $\mathbf{M}_{12}^K$ requires 3 sums of 2 coordinates of $c$, thus $3\mathbf{A}_4$ can be saved if these sums are precomputed. It also requires $6\mathbf{M}_4^K$ with one constant term (the 3 coordinates of $c$ and the 3 sums of 2 coordinates).
- Each $\mathbf{M}_4^K$ involving a constant term $b$ requires the $\mathbb{F}_{p^4}/\mathbb{F}_{p^2}$ trace of $b$, thus one $\mathbf{A}_2$ can be saved if this trace is precomputed. Hence $6\mathbf{A}_2$ are saved at this level. Each $\mathbf{M}_4^K$ also requires $3\mathbf{M}_2$ with one constant term (the 2 coordinates of $b$ and its trace).
- Again, one $\mathbf{A}_1$ can be saved for each $\mathbf{M}_2^K$ involving a constant term if its trace is precomputed. Hence $18\mathbf{A}_1$ can be saved at this level when the Karatsuba method is used for $\mathbf{M}_2$ (*i.e.* if $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$ according to Section 3.5).

In both cases, $42\mathbf{A}_1$ can be saved for each multiplication in $\mathbb{F}_{p^{12}}$ involving a constant term if its traces are precomputed. This is about 20% of the total number of additions in $\mathbf{M}_{12}$ which is significant if the relative cost of an addition compared to a multiplication in $\mathbb{F}_p$ is not small. If $\mathbf{A}_1 > 0.33\mathbf{M}_1$, the schoolbook method is used for $\mathbb{F}_{p^2}$ multiplication and only $24\mathbf{A}_1$ are saved in this case.

## 3.10 Summary of $\mathbb{F}_{p^{12}}$ arithmetic

In the previous sections, we explained how to choose the tower field depending on $u$ and on the relative cost of $\mathbb{F}_p$ operations. We also saw that the arithmetic choices to be made are essentially depending on the $\mathbb{F}_{p^2}$ arithmetic. Let us now recapitulate these choices.

### 3.10.1 The case $\mu = -1$

This choice can be made if $u$ is odd and $\xi$ can be chosen "small" in most cases :

- $\xi = 1 + \mathbf{i}$ if $u = 7$ or 11 modulo 12 and $\mathbf{m}_{2,\xi} = 2\mathbf{A}_1$,
- $\xi = 2+\mathbf{i}$ if $u = 1, 3, 13, 27, 33, 37, 41, 57$ modulo 60 and $\mathbf{m}_{2,\xi} = 2\mathbf{A}_1 + 2\mathbf{A}'_1$,
- $\xi = 3 + \mathbf{i}$ if $u = 17, 39$ or 53 modulo 60 and $\mathbf{m}_{2,\xi} = 4\mathbf{A}_1 + 2\mathbf{A}'_1$.

Let us first assume that $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$. In this case, the full $\mathbb{F}_{p^{12}}$ multiplication and the sparse multiplication should be done using Karatsuba arithmetic at all levels of the tower. Concerning the $\mathbb{F}_{p^{12}}$ squaring, the Chung Hasan method should be used for squaring in the degree 3 extension ($\mathbb{F}_{p^6}/\mathbb{F}_{p^2}$ or $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^4}$) but all the other multiplications should use the Karatsuba method. The overall complexities are summarized in Table 7. For the complexities using our new improvements given in Section 3.9, we assumed that $\mathbf{M}_{12}$ is only used in the final exponentiation and $\mathbf{sM}_{12}$ is only used in the Miller loop (which is always the case in practice). We also assumed for simplifying that the $\xi - 1$ trick saves $2\mathbf{A}_1$ whatever the choice of $\xi$ (in fact it saves $2\mathbf{A'}_1$ instead of $2\mathbf{A}_1$ if $\xi = 2 + \mathbf{i}$). An interesting point is that, if our improvements are used, the results given in Table 7 also hold if $\mathbf{A}_1 > 0.33\mathbf{M}_1$ (because less additions are required). Finally, in this table (as well as in the following ones), K, CH, SB and $\mathbb{C}$ denote the methods used for arithmetic and SB/$\mathbb{C}$ means that the schoolbook method is used for multiplications and the complex method is used for squarings.

| Operation | Case | Arithmetic used | | | | Number of operations without/with improvement | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | $\mathbb{F}_{p^2}$ | $\mathbb{F}_{p^4}$ | $\mathbb{F}_{p^6}$ | $\mathbb{F}_{p^{12}}$ | $\mathbf{M}_1$ | $\mathbf{A}_1$ | $\mathbf{A'}_1$ | $\mathbf{m}_{2,\xi}$ |
| $\mathbf{M}_{12}$ | $2,3,2$ | K | | K | K | 54 | 210/162 | 0 | 7 |
| | $2,2,3$ | | K | | | | 210/156 | | 8 |
| $\mathbf{sM}_{12}$ | $2,3,2$ | K | | K | K | 39 | 115/106 | 0 | 3 |
| | $2,2,3$ | | K | | | | 117/100 | | 4 |
| $\mathbf{S}_{12}$ | $2,3,2$ | K/$\mathbb{C}$ | | CH | K | 36 | 120/109 | 15 | 7 |
| | $2,2,3$ | | K | | CH | | 124/99 | 13 | |

Table 7: $\mathbb{F}_{p^{12}}$ complexities if $\mu = -1$ (assuming $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$ if our new improvements are not used)

**Remark 7.** *We can see that, if our new improvements are not used, it is slightly better to choose a $2,3,2$ extension tower. This is in accordance with the state of the art (see [20, 14, 3, 27] for example). However, our improvements take more advantage of a $2,2,3$ extension tower and then this way to build $\mathbb{F}_{p^{12}}$ becomes the one to choose.*

The other cases are quite similar so we are only giving the final complexities in tables. Let us first give in Table 8 the $\mathbb{F}_{p^{12}}$ complexities if $\mathbf{A}_1 > 0.33\mathbf{M}_1$.

| Operation | Case | Arithmetic used | | | | Number of operations | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | $\mathbb{F}_{p^2}$ | $\mathbb{F}_{p^4}$ | $\mathbb{F}_{p^6}$ | $\mathbb{F}_{p^{12}}$ | $\mathbf{M}_1$ | $\mathbf{A}_1$ | $\mathbf{A'}_1$ | $\mathbf{m}_{2,\xi}$ |
| $\mathbf{M}_{12}$ | $2,3,2$ | SB | | K | K | 72 | 156 | 0 | 7 |
| | $2,2,3$ | | K | | | | 156 | | 8 |
| $\mathbf{sM}_{12}$ | $2,3,2$ | SB | | K | K | 52 | 76 | 0 | 3 |
| | $2,2,3$ | | K | | | | 78 | | 4 |
| $\mathbf{S}_{12}$ | $2,3,2$ | SB/$\mathbb{C}$ | | CH | K | 42 | 102 | 15 | 7 |
| | $2,2,3$ | | K | | CH | | 106 | 13 | |

Table 8: $\mathbb{F}_{p^{12}}$ complexities if $\mu = -1$ assuming our improvements are not used and $\mathbf{A}_1 > 0.33\mathbf{M}_1$

### 3.10.2 The case $\mu = -2$

This choice should be made if the previous one cannot, in the case where $u = 2$ modulo 4. Tables 9 and 10 give the complexities of $\mathbb{F}_{p^{12}}$ operations when $\xi = \sqrt{-2}$ (which means, according to Section 3.8, that $u = 2$ or 10 modulo 12) and when $\xi = 2 + \sqrt{-2}$ (which means that $u = 6$ or 30 modulo 36 but not 18).

| Oper-ation | Case | Arithmetic used | | | | Number of operations without/with improvement | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | $\mathbf{M}_1$ | if $\xi = \sqrt{-2}$ | | if $\xi = 2 + \sqrt{-2}$ | |
| | | $\mathbb{F}_{p^2}$ | $\mathbb{F}_{p^4}$ | $\mathbb{F}_{p^6}$ | $\mathbb{F}_{p^{12}}$ | | $\mathbf{A}_1$ | $\mathbf{A'}_1$ | $\mathbf{A}_1$ | $\mathbf{A'}_1$ |
| $\mathbf{M}_{12}$ | $2, 3, 2$ | K | | K | K | 54 | 210/168 | 25 | 224/182 | 32/29 |
| | $2, 2, 3$ | | K | | | | 210/168 | 26 | 226/184 | 34/28 |
| $\mathbf{sM}_{12}$ | $2, 3, 2$ | K | | K | K | 39 | 115/106 | 16 | 121/112 | 19 |
| | $2, 2, 3$ | | K | | | | 117/106 | 17 | 125/114 | 21/18 |
| $\mathbf{S}_{12}$ | $2, 3, 2$ | K/$\mathbb{C}$ | | CH | K | 36 | 129/118 | 37 | 143/132 | 44 |
| | $2, 2, 3$ | | K | | CH | | 133/118 | 35 | 147/132 | 42/37 |

Table 9: $\mathbb{F}_{p^{12}}$ complexities if $\mu = -2$ (assuming $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$ if our improvements are not used)

| Oper-ation | Case | Arithmetic used | | | | Number of operations | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | $\mathbf{M}_1$ | if $\xi = \sqrt{-2}$ | | if $\xi = 2 + \sqrt{-2}$ | |
| | | $\mathbb{F}_{p^2}$ | $\mathbb{F}_{p^4}$ | $\mathbb{F}_{p^6}$ | $\mathbb{F}_{p^{12}}$ | | $\mathbf{A}_1$ | $\mathbf{A'}_1$ | $\mathbf{A}_1$ | $\mathbf{A'}_1$ |
| $\mathbf{M}_{12}$ | $2, 3, 2$ | SB | | K | K | 72 | 156 | 25 | 170 | 32 |
| | $2, 2, 3$ | | K | | | | 156 | 26 | 172 | 34 |
| $\mathbf{sM}_{12}$ | $2, 3, 2$ | SB | | K | K | 52 | 76 | 16 | 82 | 19 |
| | $2, 2, 3$ | | K | | | | 78 | 17 | 86 | 21 |
| $\mathbf{S}_{12}$ | $2, 3, 2$ | SB | | K | $\mathbb{C}$ | 48 | 108 | 24 | 120 | 30 |
| $\mathbf{s}_1 = \mathbf{M}_1$ | $2, 2, 3$ | | K/$\mathbb{C}$ | | CH | | 106 | 32 | 126 | 42 |
| $\mathbf{S}_{12}$ | $2, 3, 2$ | SB | | CH | K | 47.4 | 93 | 37 | 107 | 44 |
| $\mathbf{s}_1 = 0.8\mathbf{M}_1$ | $2, 2, 3$ | | K | | CH | | 97 | 35 | 111 | 40 |

Table 10: $\mathbb{F}_{p^{12}}$ complexities if $\mu = -2$ assuming our improvements are not used and $\mathbf{A}_1 > 0.33\mathbf{M}_1$

### 3.10.3 The case $\mu = -5$

This choice should be made if the previous ones cannot, in the case where $u = 8, 12$ or 16 modulo 20. Tables 11 and 12 give the complexities of $\mathbb{F}_{p^{12}}$ operations when $\xi = \sqrt{-5}$ which means, according to Section 3.8, that $u = 12, 16, 28, 48, 52$ or 56 modulo 60 (but not $8, 32$ or 36).

| Oper-ation | Case | Arithmetic used | | | | Number of operations without/with improvement | | |
|---|---|---|---|---|---|---|---|---|
| | | $\mathbb{F}_{p^2}$ | $\mathbb{F}_{p^4}$ | $\mathbb{F}_{p^6}$ | $\mathbb{F}_{p^{12}}$ | $\mathbf{M}_1$ | $\mathbf{A}_1$ | $\mathbf{A'}_1$ |
| $\mathbf{M}_{12}$ | $2,3,2$ | K | | K | K | 54 | 225/183 | 50 |
| | $2,2,3$ | | K | | | | 226/184 | 52 |
| $\mathbf{sM}_{12}$ | $2,3,2$ | K | | K | K | 39 | 131/122 | 32 |
| | $2,2,3$ | | K | | | | 134/123 | 34 |
| $\mathbf{S}_{12}$ | $2,3,2$ | K/$\mathbb{C}$ | | CH | K | 36 | 151/140 | 50 |
| | $2,2,3$ | | K | | CH | | 155/140 | 48 |

Table 11: $\mathbb{F}_{p^{12}}$ complexities if $\mu = -5$ (assuming $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$ if our improvements are not used)

| Oper-ation | Case | Arithmetic used | | | | Number of operations | | |
|---|---|---|---|---|---|---|---|---|
| | | $\mathbb{F}_{p^2}$ | $\mathbb{F}_{p^4}$ | $\mathbb{F}_{p^6}$ | $\mathbb{F}_{p^{12}}$ | $\mathbf{M}_1$ | $\mathbf{A}_1$ | $\mathbf{A'}_1$ |
| $\mathbf{M}_{12}$ | $2,3,2$ | SB | | K | K | 72 | 181 | 50 |
| | $2,2,3$ | | K | | | | 182 | 52 |
| $\mathbf{sM}_{12}$ | $2,3,2$ | SB | | K | K | 52 | 92 | 32 |
| | $2,2,3$ | | K | | | | 95 | 34 |
| $\mathbf{S}_{12}$ | $2,3,2$ | SB/$\mathbb{C}$ | | CH | K | 42 | 133 | 50 |
| | $2,2,3$ | | K | | CH | | 137 | 48 |

Table 12: $\mathbb{F}_{p^{12}}$ complexities if $\mu = -5$ assuming our improvements are not used and $\mathbf{A}_1 > 0.33\mathbf{M}_1$

### 3.10.4 The case $\mu$ large

This choice should be made if $u = 0$ or $4$ modulo $20$ and we assume that $\xi = \sqrt{\mu}$ since it is always possible to choose such a $\mu$. In this case, we have $\mathbf{m}_{1,\mu} = \mathbf{m}_{2,\xi} = \mathbf{M}_1$ and the complexities of $\mathbb{F}_{p^{12}}$ operations are given in Tables 13 and 14.

| Oper-ation | Case | Arithmetic used | | | | Number of operations without/with improvement | | |
|---|---|---|---|---|---|---|---|---|
| | | $\mathbb{F}_{p^2}$ | $\mathbb{F}_{p^4}$ | $\mathbb{F}_{p^6}$ | $\mathbb{F}_{p^{12}}$ | $\mathbf{M}_1$ | $\mathbf{A}_1$ | $\mathbf{A'}_1$ |
| $\mathbf{M}_{12}$ | $2,3,2$ | K | | K | K | 79 | 210/168 | 0 |
| | $2,2,3$ | | K | | | 80 | 210/168 | |
| $\mathbf{sM}_{12}$ | $2,3,2$ | K | | K | K | 55 | 115/106 | 0 |
| | $2,2,3$ | | K | | | 56 | 117/106 | |
| $\mathbf{S}_{12}$ | $2,3,2$ | K | | K | $\mathbb{C}$ | 54 | 144/133 | 6 |
| | $2,2,3$ | | K/$\mathbb{C}$ | | CH | 58 | 142/127 | 10 |

Table 13: $\mathbb{F}_{p^{12}}$ complexities if $\mu$ is large (assuming $\mathbf{A}_1 \leq 0.33\mathbf{M}_1$ if our improvements are not used)

| Oper-<br>ation | Case | Arithmetic used<br>$\mathbb{F}_{p^2}$ | <br>$\mathbb{F}_{p^4}$ | <br>$\mathbb{F}_{p^6}$ | <br>$\mathbb{F}_{p^{12}}$ | Number of operations<br>$\mathbf{M}_1$ | <br>$\mathbf{A}_1$ | <br>$\mathbf{A'}_1$ |
|---|---|---|---|---|---|---|---|---|
| $\mathbf{M}_{12}$ | $2,3,2$ | SB | | K | K | 97 | 156 | 0 |
| | $2,2,3$ | | K | | | 98 | | |
| $\mathbf{sM}_{12}$ | $2,3,2$ | SB | | K | K | 68 | 76 | 0 |
| | $2,2,3$ | | K | | | 69 | 78 | |
| $\mathbf{S}_{12}$ | $2,3,2$ | SB | | K | $\mathbb{C}$ | 66 | 108 | 6 |
| $\mathrm{s}_1{=}\mathrm{M}_1$ | $2,2,3$ | | K/$\mathbb{C}$ | | CH | 70 | 106 | 10 |
| $\mathbf{S}_{12}$ | $2,3,2$ | SB | | K | $\mathbb{C}$ | 66 | 108 | 6 |
| $\mathrm{s}_1{=}\mathrm{M}_1$ | $2,2,3$ | | K/SB | | CH | 73.6 | 82 | 16 |

Table 14: $\mathbb{F}_{p^{12}}$ complexities if $\mu$ is large assuming our improvements are not used and $\mathbf{A}_1 > 0.33\mathbf{M}_1$

### 3.10.5 Some remarks

Looking at these tables, we can make some interesting remarks:

- As noticed in Remark 7 building $\mathbb{F}_{p^{12}}$ with a $2, 3, 2$ tower remains preferable in most cases to a $2, 2, 3$ tower if our new improvements are not used but this is no longer the case if there are used. Anyway, the difference between the complexities of the two choices for building $\mathbb{F}_{p^{12}}$ is negligible.

- As already mentioned, the choice of $\mu$ has a great influence on the number of additions involved in $\mathbb{F}_{p^{12}}$ arithmetic, so it should be chosen first, even if the choice of $\xi$ is not optimal in this case. For example, it is better to take $\mu = -2, \xi = 2 + \sqrt{-2}$ than $\mu = -5, \xi = \sqrt{-5}$.

- Our new improvements have a significant impact on $\mathbb{F}_{p^{12}}$ arithmetic if $\mathbb{F}_p$ additions are costly which, as explained in Section 3.1, is often the case in low level implementations. For example if $\mathbf{A}_1 = 0.25\mathbf{M}_1$ (which is probably not far from the average cost of additions in hardware implementations), our improvements are providing a gain between 6 and 13% on $\mathbb{F}_{p^{12}}$ arithmetic (and then on pairing computation). In fact the gain is even not so negligible if $\mathbb{F}_p$ additions are cheap (like in software implementations). For example, if $\mathbf{A}_1 = 0.1\mathbf{M}_1$, it is between 3 and 7%.

- The cost of $\mathbb{F}_p$ squaring (with respect to $\mathbb{F}_p$ multiplication) and $\mathbb{F}_p$ doubling (with respect to $\mathbb{F}_p$ addition) have very few impact on the choices to be made for $\mathbb{F}_{p^{12}}$ arithmetic.

## 4 Choosing $u$

The parameter $u$ is involved at several levels of the pairing computation, so that the best choice is not trivial to do. Let us summarize the constraints on $u$ that we have to deal with in order to make a good choice.

- The parameter $u$ is defining the security level. Indeed, it is both parametrizing the size of the elliptic curve (whose prime order is $36u^4+36u^3+18u^2+6u+1$) and the number of elements of the target finite field (which is $(36u^4+36u^3+24u^2+6u+1)^{12}$).

- It is involved as an exponent in the Miller loop. More precisely, in the case of an optimal Ate pairing, the exponent of the Miller loop is $6u+2$. In order to optimize this step, $u$ should be chosen such that $6u+2$ is sparse.

- It is involved as an exponent in the final exponentiation. If the addition chain given in Section 2.4 is used, $u$ is directly used (three times) as an exponent, so it should be sparse to ensure a fast final exponentiation. Other final exponentiation methods may involve exponentiations by $6u+5$ and $6u^2+1$ [21, 23] or $6u+4$ [23] but these quantities are usually sparse at the same time as $u$.

- The sign of $u$ has no consequence in terms of complexities of the algorithms involved. Indeed, changing $u$ in $-u$ costs an $\mathbb{F}_{p^{12}}$ inversion, but this inversion can be replaced by a conjugation in $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^6}$ thanks to the final exponentiation.

- Choosing $u$ with a signed binary representation (to facilitate the research of a sparse $u$) is possible if the exponentiation algorithms are adapted.

- The choice of $u$ has a great impact on $\mathbb{F}_{p^{12}}$ arithmetic. The work done in Section 3 shows that the choice of the extension tower is a direct consequence of the value of $u$ modulo $12, 20, 36$ or $60$. It is summarized in Table 4. The best choice is $u = 7$ or $11$ modulo $12$ so that we can use $\mu = -1$ and $\xi = 1 + \mathbf{i}$. Depending on the situation, it could be better to choose a sparser $u \neq 7, 11$ modulo 12 or reciprocally a $u$ of higher Hamming weight but congruent to 7 or 11 modulo 12.

Hence $u$ should be chosen as sparse as possible and with the best possible way to build $\mathbb{F}_{p^{12}}$. Moreover, its size must ensure the right security level. For example, at the 128-bit security level, $u$ should be a 63-bit integer. A 95-bit integer provides a 192-bit security level on the elliptic curve but not in $\mathbb{F}_{p^{12}}$. To get this level of security, a 169 or 170-bit integer $u$ should be chosen.

Thanks to the results of Section 3.8, finding an appropriate value of $u$ can be easily done by an exhaustive search with any software which is able to check integer's primality. Unfortunately, only few values of $u$ with very low Hamming weight can be found. The best choice at the 128-bit security level is given by $u = -2^{61} - 2^{55} - 1$ [44], even if it is ensuring a slightly smaller security level than 128. It has weight 3 and is congruent to 11 modulo 12, so that $\mu = -1$ and $\xi = 1 + \mathbf{i}$ can be used to build $\mathbb{F}_{p^{12}}$ (and to twist the curve). For these reasons, it is widely used in the literature. However, relaxing the constraint on the weight of $u$ allows to generate many good values of $u$ (of weight $4, 5$ or $6$ for example) that can be used in a database of pairing friendly parameters or for higher (or smaller) security levels.

# 5 Choosing the groups involved

## 5.1 Choosing the elliptic curve

Choosing the curve is not difficult once $u$ is chosen. Indeed, the base field $\mathbb{F}_p$ and the cardinality of the curve $r$ are fixed by the BN parametrisation given in Section 2.1. Then, we only have to find $b \in \mathbb{F}_p$ such that the curve defined by the equation

$$E : y^2 = x^3 + b$$

has cardinality $r$. Because of the form of the equation and of the existence of a sextic twist, there are only 6 isomorphism classes over $\mathbb{F}_p$. So, if $b$ is randomly chosen, there is about one chance in 6 that $\#E\left(\mathbb{F}_p\right) = r$. Checking the cardinality is easily done by verifying that $rP = O_E$ for some $P \in E(\mathbb{F}_p)$.

**Remark 8.** *We will see in Section 6 that the coefficient $b$ is involved in the elliptic curve arithmetic, so that it is better to choose it small.*

However, $E$ is not the only curve to choose, we also have to choose its twist $E'$. If $\xi$ is the $\mathbb{F}_{p^2}$ element chosen to define $\mathbb{F}_{p^{12}}$, $E'$ is defined over $\mathbb{F}_{p^2}$ by the equation

$$y^2 = x^3 + b',$$

where $b' = b/\xi$ or $b' = b\xi$, such that $r$ divides the cardinality of $E'$. This is the case for exactly one of the two choices for $b'$ [2]. Again, the correct choice for $b'$ is done by checking the cardinality of $E'$, thanks to some point in $E'(\mathbb{F}_{p^2})$.

Since $E(\mathbb{F}_p)$ has prime order, it is naturally protected against subgroup attacks which exploit small prime divisors of the cofactor [38]. However, this is not the case of $E'(\mathbb{F}_{p^2})$ whose order equals $r(2p - r)$. For example, the value of $u$ given in Section 4, and usually used in the literature for the 128-bit security level, is not naturally protected against subgroup attacks. This can be prevented by using (possibly expensive) membership tests. If we want to avoid these tests, the parameter $u$ should be chosen such that both $r$ and $2p - r$ are prime numbers [5].

## 5.2 Choosing the generators

In this section, we present an explicit method to construct the points $P$ and $Q$ potentially involved in the optimal Ate pairing computations. Since $E(\mathbb{F}_p)$ has prime order $r$, it is trivial to find a suitable candidate for $P$: any point $P \neq O_E$ has order $r$.

However, generating a suitable point $Q$ seems less easy because it must be of order $r$ in $E\left(\mathbb{F}_{p^{12}}\right)$, come from the twisted curve and be an eigenvector for the eigenvalue $p$ of the Frobenius map. In fact, it is not so difficult because the last condition is a consequence of the other ones [29]. Finding a suitable point $Q$ is then done in the following way:

1. Choose a random point $Q'$ in $E'(\mathbb{F}_{p^2})$.

2. If $Q'$ has not order $r$ (which is statistically always the case), replace it by $(p-2r)Q'$ which has order $r$ since $\#E'(\mathbb{F}_{p^2}) = (p-2r)r$.
3. If $Q' = O_{E'}$ then repeat steps 1 and 2.
4. Map $Q'$ to $E\left(\mathbb{F}_{p^{12}}\right)$ thanks to the twist isomorphism between $E$ and $E'$ over $\mathbb{F}_{p^{12}}$. If $b' = b/\xi$ this is nothing but
$$Q = \left(x_{Q'}\gamma^2, y_{Q'}\gamma^3\right) \text{ if } \gamma^6 = \xi.$$

Then $Q$ is a valuable candidate, in particular it lies in the $p$-eigenspace of $\pi$.

# 6 Choosing the system of coordinates

In this section, we give the formulas for adding and doubling points on BN curves (with the line computation) and their complexities in the affine and the projective cases (it is now well known that Jacobian coordinates are always less efficient than projective coordinates for pairing computations [18]). This allows to determine which system of coordinates should be chosen, depending on the context. Assuming the previous choices, the two operations involved in the Miller loop are

**The doubling step.** In this step, we have to

- double a temporary point $T = \left(x_T\gamma^2, y_T\gamma^3\right) \in E\left(\mathbb{F}_{p^{12}}\right)$ with $x_T, y_T \in \mathbb{F}_{p^2}$,
- compute the tangent line to $E$ at $T$,
- evaluate it at $P = (x_P, y_P) \in E\left(\mathbb{F}_p\right)$.

**The addition step** In this step, we have to

- add $Q = \left(x_Q\gamma^2, y_Q\gamma^3\right)$ and $T = \left(x_T\gamma^2, y_T\gamma^3\right)$ in $E\left(\mathbb{F}_{p^{12}}\right)$ with $x_Q, y_Q, x_T, y_T \in \mathbb{F}_{p^2}$,
- compute the line passing through $T$ and $Q$,
- evaluate it at $P = (x_P, y_P) \in E\left(\mathbb{F}_p\right)$.

## 6.1 Affine coordinates

The slope of the line passing through $T$ and $Q$ (or the tangent line at $T$ if $T = Q$) is $\lambda\gamma$, with

$$\lambda = \frac{y_T - y_Q}{x_T - x_Q} \quad \left(\text{or} \quad \lambda = \frac{3x_T^2}{2y_T}\right).$$

Then $T+Q$ (or $2T$) can be written in the form $\left(x_{T+Q}\gamma^2, y_{T+Q}\gamma^3\right)$ with

$$x_{T+Q} = \lambda^2 - x_T - x_Q \quad \text{and} \quad y_{T+Q} = \lambda\left(x_T - x_{T+Q}\right) - y_T.$$

The equation of the line involved in the operation is $y = \lambda\gamma\left(x - x_T\gamma^2\right) - y_T\gamma^3$, thus the $\mathbb{F}_{p^{12}}$ element involved in the update of $f$ in Algorithm 1 is

$$\ell = y_P - \lambda x_P\gamma + \left(\lambda x_T - y_T\right)\gamma^3.$$

Assuming that $-x_P$ is precomputed, the cost of the addition step (including the line computation) is then $\mathbf{I}_2 + 3\mathbf{M}_2 + \mathbf{S}_2 + 2\mathbf{M}_1 + 7\mathbf{A}_2$ and the cost of the doubling step is $\mathbf{I}_2 + 3\mathbf{M}_2 + 2\mathbf{S}_2 + 2\mathbf{M}_1 + 5\mathbf{A}_2 + 2\mathbf{A'}_2$.

**Remark 9.** *Since $\lambda$ is used three times in $\mathbb{F}_{p^2}$ operations ($\lambda^2, \lambda\left(x_T - x_{T+Q}\right)$ and $\lambda x_T$), $2\mathbf{A}_1$ can be saved using our idea of precomputing its trace if the Karatsuba/complex methods are used for $\mathbb{F}_{p^2}$ arithmetic. In the same way, $x_T$ is used twice in the doubling step so that an additional $\mathbf{A}_1$ can be saved in this case.*

## 6.2 Projective coordinates

In order to avoid inversions in $\mathbb{F}_{p^2}$, projective coordinates are used for the point $T$, so that $T = \left(X_T\gamma^2, Y_T\gamma^3, Z_T\right)$ with $X_T, Y_T$ and $Z_T \in \mathbb{F}_{p^2}$. However, the point $Q$ is kept in affine coordinates (mixed addition method). According to [18], $2T = \left(X_{2T}\gamma^2, Y_{2T}\gamma^3, Z_{2T}\right)$ with

$$\begin{aligned} X_{2T} &= 2X_TY_T(Y_T^2 - 9bZ_T^2) \\ Y_{2T} &= \left(Y_T^2 + 9bZ_T^2\right)^2 - 12(3bZ_T^2)^2 \\ Z_{2T} &= 8Y_T^3Z_T \end{aligned}$$

and the equation of the tangent to the curve at $T$ is (up to some subfield multiple)

$$\ell = 2y_PY_TZ_T - 3x_PX_T^2\gamma + \left(Y_T^2 - 3bZ_T^2\right)\gamma^3.$$

Assuming that $-3x_P$ is precomputed, the doubling step (including the line computation) then requires $2\mathbf{M}_2 + 7\mathbf{S}_2 + 4\mathbf{M}_1 + 13\mathbf{A}_2 + 5\mathbf{A'}_2 + 2\mathbf{m}_{1,b}$. In order to obtain this complexity, the double products like $2X_TY_T$ are computed by $(X_T + Y_T)^2 - X_T^2 - Y_T^2$. This trick is not always interesting over $\mathbb{F}_p$ (e.g. if $\mathbf{M}_1 = \mathbf{S}_1$) but it is always interesting over $\mathbb{F}_{p^2}$ because $\mathbf{S}_2$ is clearly cheaper than $\mathbf{M}_2$ according to Section 3.5.

In the same way, if

$$\begin{aligned} N &= Y_T - y_QZ_T, \\ D &= X_T - x_QZ_T \quad \text{(so that } \lambda = \tfrac{N}{D}\text{)}, \\ X &= N^2Z_T - X_TD^2 - x_QD^2Z_T, \end{aligned}$$

we compute the addition step with

$$\begin{aligned} X_{T+Q} &= DX \\ Y_{T+Q} &= N(x_Q D^2 Z_T - X) - y_Q D^3 Z_T \\ Z_{T+Q} &= D^3 Z_T \\ \ell &= y_P D - N x_P \gamma + (N x_Q - D y_Q)\,\gamma^3. \end{aligned}$$

Assuming that $-x_P$ is precomputed, this requires $12\mathbf{M}_2 + 2\mathbf{S}_2 + 4\mathbf{M}_1 + 7\mathbf{A}_2$.

**Remark 10.** *Again, many $\mathbb{F}_{p^2}$ operands are used several times during the computation, so that precomputing the traces saves additions in $\mathbb{F}_p$. We do not give details here because the addition step is rarely used in the Miller loop but it is not difficult to see that $16\mathbf{A}_1$ can be saved if Karatsuba/complex arithmetic is used for $\mathbb{F}_{p^2}$ arithmetic.*

## 6.3 Consequences of formulas

Several remarks can be made looking at these formulas.

The first one is that the influence of $b$ is small since it is just involved in two multiplications by $\mathbb{F}_p$ elements. Hence a sparse $u$ or a value of $u$ enabling a nice choice of $\mu$ (and $\xi$) should be preferred, even if a very small value of $b$ is not available.

The second one is that, as mentioned in Sections 3.6 and 3.7, the line $\ell$ is of the form $b_0 + b_1\gamma + b_3\gamma^3$ with $b_i \in \mathbb{F}_{p^2}$, thus it is sparse in $\mathbb{F}_{p^{12}}$ and a multiplication by $\ell$ is faster than a full multiplication in $\mathbb{F}_{p^{12}}$.

The third one is that, as already mentioned in [34, 1, 35, 27], it can be better to use affine coordinates than projective coordinates, depending on the context. Indeed, using the complexity formula for $\mathbf{I}_2$ given in Section 3.2.4 and the complexities obtained for the doubling step in affine and projective coordinates, it is easy to verify that affine coordinates become interesting for this step (and then for the full Miller loop) as soon as

$$\mathbf{I}_1 < 5\mathbf{S}_2 - \mathbf{M}_2 - 2\mathbf{S}_1 + 15\mathbf{A}_1 + 6\mathbf{A'}_1 + 2\mathbf{m}_{1,b} - \mathbf{m}_{1,\mu}.$$

For example, in the case $\mu = -1, \mathbf{A}_1 \leq 0.33\mathbf{M}_1$, Tables 1 and 2 show that affine coordinates are interesting as soon as

$$\mathbf{I}_1 < 7\mathbf{M}_1 - 2\mathbf{S}_1 + 20\mathbf{A}_1 + 11\mathbf{A'}_1 + 2\mathbf{m}_{1,b}. \tag{3}$$

Depending on the way to implement $\mathbb{F}_p$ inversion, this inequality may hold in practice, especially if $\mathbb{F}_p$ addition are not negligible. In Table 15, we give the maximum cost of $\mathbf{I}_1$ for which affine coordinates should be chosen, depending on the context. To make the results more readable, we assumed that $\mathbf{S}_1 = \mathbf{M}_1, \mathbf{A}_1 = \mathbf{A'}_1$ and $b = 2$ (which is the least advantageous value for affine coordinates). In any case, the non-simplified result is very similar to (3).

| $\mu$ | Use affine coordinates if |
|---|---|
| $-1$ | $\mathbf{I}_1 < 5\mathbf{M}_1 + 33\mathbf{A}_1$ |
| $-2$ | $\mathbf{I}_1 < 5\mathbf{M}_1 + 41\mathbf{A}_1$ |
| $-5$ | $\mathbf{I}_1 < 5\mathbf{M}_1 + 42\mathbf{A}_1$ |
| any | $\mathbf{I}_1 < 13\mathbf{M}_1 + 28\mathbf{A}_1$ |

Table 15: Affine coordinates versus projective ones

# 7 Algorithms that must be used for efficient implementation

We describe here some algorithms which should not influence the choice of the parameters but which are important for the efficiency of pairing computation.

## 7.1 Frobenius computation

Let

$$a = b_0 + b_1\gamma + b_2\gamma^2 + b_3\gamma^3 + b_4\gamma^4 + b_5\gamma^5$$

be an arbitrary element of $\mathbb{F}_{p^{12}}$, where $b_j \in \mathbb{F}_{p^2}$, $j = 0, \ldots, 5$ and $\gamma^6 = \xi \in \mathbb{F}_{p^2}$. The Frobenius map, which consists in computing $a^p$, can be easily written in terms of $a$. In this section, we give a way to compute it efficiently, as well as $a^{p^i}$ for $i = 1, \ldots, 11$.

### 7.1.1 Computation of $a^p$

Since the Frobenius map is linear, we have

$$a^p = b_0^p + b_1^p\gamma^p + b_2^p\left(\gamma^2\right)^p + b_3^p\left(\gamma^3\right)^p + b_4^p\left(\gamma^4\right)^p + b_5^p\left(\gamma^5\right)^p,$$

and $b_j^p$ is just the conjugate $\overline{b_j}$ of $b_j$. Therefore, we only have to study the $\left(\gamma^j\right)^p$. For this, we define the constant

$$\delta = \xi^{\frac{p-1}{6}} \in \mathbb{F}_{p^2},$$

so that $\gamma^p = (\gamma^6)^{(p-1)/6}\gamma = \delta\gamma$ and we have

$$a^p = \overline{b_0} + \overline{b_1}\delta\gamma + \overline{b_2}\delta^2\gamma^2 + \overline{b_3}\delta^3\gamma^3 + \overline{b_4}\delta^4\gamma^4 + \overline{b_5}\delta^5\gamma^5.$$

The constants $\delta^j$ are only depending on the way to build $\mathbb{F}_{p^{12}}$, so they can be precomputed. Finally, computing $a^p$ requires $5\mathbf{M}_2$. If precomputing the $\delta^j$ is a problem in terms of memory resources, they can of course all be recovered from $\delta$ using only $2\mathbf{S}_2 + 2\mathbf{M}_2$. But in any case, $\delta$ should be precomputed since it is very expensive to compute.

Let us finally note that some particular choices of $\mu$ and $\xi$ allow to improve this computation by using a precomputed constant in $\mathbb{F}_p$ instead of $\delta \in \mathbb{F}_{p^2}$. Some details for the most interesting cases are given in Section 7.1.4.

### 7.1.2 Computation of $a^{p^2}$

In this case

$$a^{p^2} = b_0 + b_1\gamma^{p^2} + b_2\left(\gamma^2\right)^{p^2} + b_3\left(\gamma^3\right)^{p^2} + b_4\left(\gamma^4\right)^{p^2} + b_5\left(\gamma^5\right)^{p^2}.$$

We need the new precomputed constant

$$\omega = \xi^{\frac{p^2-1}{6}} = N_{\mathbb{F}_{p^2}/\mathbb{F}_p}(\delta) \in \mathbb{F}_p.$$

Then we have $\gamma^{p^2} = (\gamma^6)^{(p^2-1)/6}\gamma = \omega\gamma$. Notice that $\omega$ is a primitive 6th root of unity (because $\xi$ is neither a square nor a cube in $\mathbb{F}_{p^2}$). In particular, we have

$$\omega^2 - \omega + 1 = 0. \tag{4}$$

Then

$$a^{p^2} = b_0 + b_1\omega\gamma + b_2\left(\omega - 1\right)\gamma^2 - b_3\gamma^3 - b_4\omega\gamma^4 - b_5\left(\omega - 1\right)\gamma^5,$$

and computing $a^{p^2}$ requires one addition in $\mathbb{F}_p$ to compute $\omega - 1$ (it can be precomputed but the interest is really limited) and 5 multiplications of an element of $\mathbb{F}_{p^2}$ by an element of $\mathbb{F}_p$, thus the total cost is $10\mathbf{M}_1 + \mathbf{A}_1$.

### 7.1.3 Generalization to the computation of $a^{p^i}$

The results of the two previous sections can be generalized as follows : for $i = 1, \ldots, 11$, $j = 0, \ldots, 5$, let $c_{i,j} = \omega^{j\lfloor i/2 \rfloor}$. Notice that it is easy to deduce from (4) that $c_{i,j} = 1$ (resp. $\omega$, $\omega - 1$, $-1$, $-\omega$, $-\omega + 1$) if $j\lfloor i/2 \rfloor = 0 \mod 6$ (resp. 1, 2, 3, 4, 5). We find that for $i = 1, \ldots, 11$,

$$a^{p^i} = \overline{b_0} + \overline{b_1}c_{i,1}\delta\gamma + \overline{b_2}c_{i,2}\delta^2\gamma^2 + \overline{b_3}c_{i,3}\delta^3\gamma^3 + \overline{b_4}c_{i,4}\delta^4\gamma^4 + \overline{b_5}c_{i,5}\delta^5\gamma^5$$

if $i$ is odd, and

$$a^{p^i} = b_0 + b_1c_{i,1}\gamma + b_2c_{i,2}\gamma^2 + b_3c_{i,3}\gamma^3 + b_4c_{i,4}\gamma^4 + b_5c_{i,5}\gamma^5$$

if $i$ is even.

As explained for $i = 2$, the $c_{i,j}$ can be easily deduced from $\omega$, so that only $\omega, \delta$ and possibly the $c_{i,j}\delta^j$ need to be precomputed. We also note that computing $a^{p^6}$ is for free, since it is nothing but the conjugation in $\mathbb{F}_{p^{12}}/\mathbb{F}_{p^6}$.

### 7.1.4 Simplification of the constant $\delta$ for odd powers

For some particular choices of $\mu$ and $\xi$, the precomputed constants for odd powers of the Frobenius map can be simplified.

**The case $\mu = -1$ and $\xi = 1 + \mathbf{i}$.** We proved in Section 3.8 that in this case, $u = 7$ or $11$ modulo $12$, so $p$ equals $7$ modulo $12$. Then

$$\delta = (1+\mathbf{i})^{\frac{p-1}{6}} = (1+\mathbf{i})^{2\frac{p-7}{12}}(1+\mathbf{i}) = (2\mathbf{i})^{\frac{p-7}{12}}(1+\mathbf{i}) = \mathbf{i}^{\frac{p-7}{12}}(1+\mathbf{i})2^{\frac{p-7}{12}},$$

thus we only need to precompute $d_1 = 2^{\frac{p-7}{12}} \in \mathbb{F}_p$ instead of $\delta \in \mathbb{F}_{p^2}$. Moreover, in this case, $\overline{\delta} = \mathbf{i}\delta$ so that $\omega = \mathbf{i}\delta^2$ and by (4), we get $\delta^4 = 1 - \mathbf{i}\delta^2$. Finally, if $d_2 = -2d_1^2$ and $d_3 = d_1 d_2$, it is easy to find that

$$\begin{aligned}
\left(\delta, \delta^2, \delta^3, \delta^4, \delta^5\right) &= \left(\overline{\xi}d_1, \mathbf{i}d_2, \xi d_3, d_2 + 1, \overline{\xi}(d_1 + d_3)\right) \text{ if } u = 11, 19 \mod 24, \\
&= \left(-\overline{\xi}d_1, \mathbf{i}d_2, -\xi d_3, d_2 + 1, -\overline{\xi}(d_1 + d_3)\right) \text{ if } u = 7, 23 \mod 24.
\end{aligned}$$

It is not necessary to precompute $d_2$ and $d_3$ because their computation requires only $\mathbf{M}_1 + \mathbf{S}_1 + \mathbf{A'}_1$. Once they are computed, computing $a^p$ requires 5 multiplications of an element of $\mathbb{F}_p$ by an element of $\mathbb{F}_{p^2}$, 2 additions ($1 + d_2$ and $d_1 + d_3$) and 3 multiplications by $\xi$ or $\overline{\xi}$. Finally, assuming that $d_1$ is precomputed, computing $a^p$ requires $11\mathbf{M}_1 + \mathbf{S}_1 + 8\mathbf{A}_1 + \mathbf{A'}_1$.

**The case $\xi = \sqrt{\mu}$.** We assume that $u$ is even (see Section 3.5.1), so $p = 1 \mod 12$. We have

$$\delta = \sqrt{\mu}^{\frac{p-1}{6}} = \mu^{\frac{p-1}{12}} \in \mathbb{F}_p.$$

Therefore, multiplications by $\delta$ and its powers by $\mathbb{F}_{p^2}$ elements cost $2\mathbf{M}_1$ and $\overline{\delta} = \delta$ so that $\omega = \delta^2$ and by (4), $\delta^4 = \delta^2 - 1$ (and $\delta^5 = \delta^3 - \delta$). Finally, if $\delta$ is precomputed, computing $a^p$ requires $11\mathbf{M}_1 + \mathbf{S}_1 + 2\mathbf{A}_1$ ($\mathbf{M}_1 + \mathbf{S}_1 + 2\mathbf{A}_1$ to compute the $\delta^j$ and $10\mathbf{M}_1$ to compute the $\overline{b_j}\delta^j$).

## 7.2 Cyclotomic squaring

As mentioned in Section 2.4, the final exponentiation requires exponentiations in the subgroup $G_{\Phi_6(p^2)}$ of $\mathbb{F}_{p^{12}}^*$ of elements of order dividing $\Phi_6(p^2)$. Since around $3\log_2 u$ squarings are performed during the final exponentiation, saving few field operations for each squaring leads to a significant improvement. Given $a \in \mathbb{F}_{p^{12}}$, the condition $a^{\Phi_6(p^2)} = 1$ combined with a description of the Frobenius action gives algebraic relations on the coordinates of $a$ (where $\mathbb{F}_{p^{12}}$ is seen as a $\mathbb{F}_{p^2}$-vector space). Granger and Scott [26] proposed to exploit these relations to simplify the expression of $a^2$.

Furthermore, these relations can be used to compress the representation of $a$ in such a way that the square of $a$ can be directly computed in its compressed form (multiplications have to be performed on the non-compressed forms). In particular, Karabina's method for compressed squaring [31, 2] costs $3\mathbf{S}_2 + 10\mathbf{A}_2 + \mathbf{m}_{2,\xi}$ less than Granger and Scott's method (in most cases). On the other hand, the decompression step is relatively expensive and requires an inversion in $\mathbb{F}_{p^2}$. The number of decompressions required for an exponentiation is at most the Hamming weight of the exponent, which is assumed to be small in our situation (see

Sections 2.4 and 4), and moreover, Montgomery's simultaneous inversion trick [41] can be used (we refer to [2] for more details). Therefore, if $\mathbb{F}_{p^2}$ inversions are available (at a reasonable cost), Karabina's method should be used for final exponentiation, otherwise, Granger and Scott's method should be chosen.

### 7.2.1 Karabina's method

This method is due to Karabina [31] and was slightly improved in [2]. Let $a = b_0 + b_1\gamma + b_2\gamma^2 + b_3\gamma^3 + b_4\gamma^4 + b_5\gamma^5 \in G_{\Phi_6(p^2)} \setminus \{1\}$, with $b_j \in \mathbb{F}_{p^2}$. The compressed representation of $a$ is $[b_1, b_2, b_4, b_5]$. The full representation (decompression) of $a$ is obtained via the formulae

$$\begin{array}{lll} b_3 = \dfrac{b_5^2\xi + 3b_2^2 - 2b_4}{4b_1}, & b_0 = (2b_3^2 + b_1b_5 - 3b_4b_2)\xi + 1, & \text{if } b_1 \neq 0, \\ b_3 = \dfrac{2b_2b_5}{b_4}, & b_0 = (2b_3^2 - 3b_4b_2)\xi + 1, & \text{if } b_1 = 0, \end{array}$$

for a cost of $\mathbf{I}_2 + 2\mathbf{M}_2 + 3\mathbf{S}_2 + 4\mathbf{A'}_2 + 6\mathbf{A}_2 + 2\mathbf{m}_{2,\xi} + \mathbf{A}_1$ (or $\mathbf{I}_2 + 2\mathbf{M}_2 + \mathbf{S}_2 + 2\mathbf{A'}_2 + 2\mathbf{A}_2 + \mathbf{m}_{2,\xi} + \mathbf{A}_1$ if $b_1 = 0$). The compressed representation $[B_1, B_2, B_4, B_5]$ of $a^2$ can be computed using the formulae

$$\begin{array}{ll} B_1 = 2b_1 + 3(S_{2,5} - S_2 - S_5)\xi & B_4 = 3(S_2 + S_5\xi) - 2b_4 \\ B_2 = 3(S_1 + S_4\xi) - 2b_2 & B_5 = 2b_5 + 3(S_{1,4} - S_1 - S_4), \end{array}$$

where $S_{i,j} = (b_i + b_j)^2$ and $S_i = b_i^2$, for a cost of $6\mathbf{S}_2 + 4\mathbf{A'}_2 + 16\mathbf{A}_2 + 3\mathbf{m}_{2,\xi}$.

### 7.2.2 Granger and Scott's method

This method is due to Granger and Scott [26]. They worked with a tower $2, 2, 3$ but their results can be easily adapted to a tower $2, 3, 2$, either by using the same technique and the formulae from Section 7.1, or simply by expanding the formulae for a tower $2, 2, 3$.

**Case** $2, 2, 3$. With the notations of Section 3.7, let $a = c_0 + c_1\gamma + c_2\gamma^2 \in \mathbb{F}_{p^{12}} = \mathbb{F}_{p^2}[\gamma]$, where $c_0, c_1, c_2 \in \mathbb{F}_{p^4}$ and write

$$a^2 = C_0 + C_1\gamma + C_2\gamma^2,$$

where $C_0, C_1, C_2 \in \mathbb{F}_{p^4}$. If $a \in G_{\Phi_6(p^2)}$, then we have the following formulae, given in [26]

$$\begin{array}{rcl} C_0 & = & 3c_0^2 - 2\overline{c_0} \\ C_1 & = & 3\beta c_2^2 + 2\overline{c_1} \\ C_2 & = & 3c_1^2 - 2\overline{c_2}, \end{array}$$

where $\overline{u + v\beta} = u - v\beta$, for $u, v \in \mathbb{F}_{p^2}$. Therefore, the square of $a \in G_{\Phi_6(p^2)}$ can be computed for a cost of $3\mathbf{S}_4 + 3\mathbf{A'}_4 + 6\mathbf{A}_4 + \mathbf{m}_{4,\beta}$ (using that an expression of the form $3u + 2v = 2(u + v) + u$ can be computed with one doubling and 2 additions).

**Case** 2, 3, 2. With the notations of Section 3.6, let $a = (b_0 + b_2\beta + b_4\beta^2) + (b_1 + b_3\beta + b_5\beta^2)\gamma \in \mathbb{F}_{p^2}[\gamma]$, where $b_i \in \mathbb{F}_{p^2}$, $i = 0, \ldots, 5$, and write

$$a^2 = (B_0 + B_2\beta + B_4\beta^2) + (B_1 + B_3\beta + B_5\beta^2)\gamma$$

where $B_i \in \mathbb{F}_{p^2}$, $i = 0, \ldots, 5$. If $a \in G_{\Phi_6(p^2)}$, then we have

$$\begin{array}{lcl} B_0 & = & 3(b_3^2\xi + b_0^2) - 2b_0 \\ B_2 & = & 3(b_4^2\xi + b_1^2) - 2b_2 \\ B_4 & = & 3(b_5^2\xi + b_2^2) - 2b_4 \\ B_1 & = & 3\xi((b_2 + b_5)^2 - b_2^2 - b_5^2)) + 2b_1 \\ B_3 & = & 3((b_0 + b_3)^2 - b_0^2 - b_3^2)) + 2b_3 \\ B_5 & = & 3((b_4 + b_1)^2 - b_4^2 - b_1^2)) + 2b_5. \end{array}$$

We find that the square of $a \in G_{\Phi_6(p^2)}$ can be computed for a cost of $9\mathbf{S}_2 + 6\mathbf{A}'_2 + 24\mathbf{A}_2 + 4\mathbf{m}_{2,\xi}$.

**Remark 11.** *According to Table 3,* $\mathbf{S}_4 = 3\mathbf{S}_2 + \mathbf{m}_{2,\xi} + 4\mathbf{A}_2$ *in the most common case and then it is easy to verify that the cost of squaring in the case* 2, 2, 3 *is exactly the same as in the case* 2, 3, 2

# 8 Conclusion

In this paper, we explained in details how to choose and generate the parameters for implementing a pairing on BN curves. More precisely, we explained the choices which should be made be made in terms of

- pairing algorithm (Miller loop, final exponentiation),
- ways to build the tower field depending on the value of the BN parameter $u$ as a nice application of old elementary arithmetic results,
- $\mathbb{F}_{p^{12}}$ arithmetic depending on the relative cost of $\mathbb{F}_p$ operations on the targeted device,
- groups involved (base elliptic curve, its twist and their generators),
- system of coordinates to be used depending on the context.

Of course, the best choices are already well-known but there are many situations where generating other parameters is necessary (pairing parameters database, other security levels than 126-bit, resistance to subgroups attacks, trusted parameters, ...) and this paper should help anyone interested by generating pairing parameters depending on the target device (security level, relative cost of $\mathbb{F}_p$ operations, memory resources, ...). Moreover, we used this opportunity to give some new improvements on $\mathbb{F}_{p^{12}}$ arithmetic (in a pairing context) in terms of $\mathbb{F}_p$-addition because they are usually not so negligible for small devices. The gain obtained on $\mathbb{F}_{p^{12}}$ arithmetic is around 10%, depending on the context and the $\mathbb{F}_{p^{12}}$ operation considered.

# References

[1] Tolga Acar, Kristin E. Lauter, Michael Naehrig, and Daniel Shumow. Affine pairings on ARM. In Michel Abdalla and Tanja Lange, editors, *Pairing-Based Cryptography - Pairing 2012*, volume 7708 of *Lecture Notes in Computer Science*, pages 203–209. Springer, 2012.

[2] D. Aranha, K. Karabina, P. Longa, C. H. Gebotys, and J López. Faster explicit formulas for computing pairings over ordinary curves. In *Advances in Cryptology EUROCRYPT 2011*, volume 6632 of *LNCS*, pages 48–68. Springer, 2011.

[3] Dieg oF. Aranha, Paulo S.L.M. Barreto, Patrick Longa, and Jefferson E. Ricardini. The realm of the pairings. In Tanja Lange, Kristin Lauter, and Petr LisonÄŻk, editors, *Selected Areas in Cryptography – SAC 2013*, volume 8282 of *Lecture Notes in Computer Science*, pages 3–25. Springer Berlin Heidelberg, 2014.

[4] Razvan Barbulescu, Pierrick Gaudry, Antoine Joux, and Emmanuel Thomé. A heuristic quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic.

[5] Paulo S. L. M. Barreto, Craig Costello, Rafael Misoczki, Michael Naehrig, Geovandro C. C. F. Pereira, and Gustavo Zanon. Subgroup security in pairing-based cryptography. In *LATINCRYPT 2015*, volume 9230 of *Lecture Notes in Computer Science*, pages 245–265. Springer-Verlag, 2015.

[6] Paulo S. L. M. Barreto, Steven D. Galbraith, Colm O. hEigeartaigh, and Michael Scott. Efficient pairing computation on supersingular abelian varieties. *IACR Cryptology ePrint Archive*, 2004:375, 2004.

[7] Paulo S. L. M. Barreto, Hae Yong Kim, Ben Lynn, and Michael Scott. Efficient algorithms for pairing-based cryptosystems. In *CRYPTO 2002*, volume 2442 of *LNCS*, pages 354–368. Springer, 2002.

[8] Paulo S. L. M. Barreto and Michael Naehrig. Pairing-friendly elliptic curves of prime order. In *Proc. of SAC 2005*, volume 3897 of *LNCS*, pages 319 –331. Springer-Verlag, 2006.

[9] Paul Barrett. Implementing the rivest shamir and adleman public key encryption algorithm on a standard digital signal processor. In *Proceedings on Advances in cryptology—CRYPTO '86*, pages 311–323, London, UK, UK, 1987. Springer-Verlag.

[10] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the weil pairing. In *CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer, 2001.

[11] Dan Boneh, Craig Gentry, and B. Waters.

[12] Dan Boneh, Ben Lynn, and Hovav Shacham. Short signatures from the weil pairing. *Journal of cryptology*, 17(4):297–319, 2004.

[13] Laura Fuentes Castaneda, Edward Knapp, and Francisco Rodrguez Henrquez. Faster hashing to ${\mathbb G}_2$. In *Selected Areas in Cryptography - 18th International Workshop, 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers*, pages 412–430, 2011.

[14] Ray C. C. Cheung, Sylvain Duquesne, Junfeng Fan, Nicolas Guillermin, Ingrid Verbauwhede, and Gavin Xiaoxu Yao. FPGA implementation of pairings using residue number system and lazy reduction. In *Cryptographic Hardware and Embedded Systems - CHES 2011*, volume 6917 of *LNCS*, pages 421–441. Springer, 2011.

[15] Jaewook Chung and M. Anwar Hasan. More generalized mersenne numbers: (extended abstract). In *Selected Areas in Cryptography, 10th Annual International Workshop, SAC 2003, Ottawa, Canada, August 14-15, 2003, Revised Papers*, pages 335–347, 2003.

[16] Jaewook Chung and M Anwar Hasan. Asymmetric squaring formulae. In *18$^{th}$ symposium on Computer Arithmetic, Montpellier, France*, IEEE conference publications, pages 113–122, 2007.

[17] Henri Cohen, Gerhard Frey, Roberto Avanzi, Christophe Doche, Tanja Lange, Kim Nguyen, and Frederik Vercauteren. *Handbook of Elliptic and Hyperelliptic Curve Cryptography, Second Edition*. Chapman & Hall/CRC, 2nd edition, 2012.

[18] Craig Costello, Tanja Lange, and Michael Naehrig. Faster pairing computations on curves with high-degree twists. In PhongQ. Nguyen and David Pointcheval, editors, *Public Key Cryptography -ĂŞ PKC 2010*, volume 6056 of *Lecture Notes in Computer Science*, pages 224–242. Springer Berlin Heidelberg, 2010.

[19] David A. Cox. *Primes of the form $x^2 + ny^2$*. 1989.

[20] Augusto Jun Devegili, Colm O'Eigeartaigh, Michael Scott, and Ricardo Dahab. Multiplication and squaring on pairing-friendly fields. *IACR Cryptology ePrint Archive*, page 471, 2006.

[21] Augusto Jun Devegili, Michael Scott, and Ricardo Dahab. Implementing cryptographic pairings over barreto-naehrig curve. pages 197–207.

[22] Sylvain Duquesne. RNS arithmetic in pk and application to fast pairing computation. *J. Mathematical Cryptology*, 5(1):51–88, 2011.

[23] Sylvain Duquesne and Loubna Ghammam. Memory-saving computation of the pairing final exponentiation on bn curves. *Groups, Complexity, Cryptology*, 2015. To appear.

[24] C. C. F. Pereira Geovandro, Marcos A. Simplício Jr., Michael Naehrig, and Paulo S. L. M. Barreto. A family of implementation-friendly bn elliptic curves. *Journal of Systems and Software*, 84(8):1319–1326, 2011.

[25] Christophe Giraud and Vincent Verneuil. Atomicity improvement for elliptic curve scalar multiplication. In Dieter Gollmann, Jean-Louis Lanet, and Julien Iguchi-Cartigny, editors, *Smart Card Research and Advanced Application*, volume 6035 of *Lecture Notes in Computer Science*, pages 80–101. Springer Berlin Heidelberg, 2010.

[26] Robert Granger and Michael Scott. Faster squaring in the cyclotomic subgroup of sixth degree extensions. In *Public Key Cryptography - PKC 2010, 13th International Conference on Practice and Theory in Public Key Cryptography, Paris, France, May 26-28, 2010. Proceedings*, pages 209–223, 2010.

[27] Gurleen Grewal, Reza Azarderakhsh, Patrick Longa, Shi Hu, and David Jao. Efficient implementation of bilinear pairings on arm processors. In LarsR. Knudsen and Huapeng Wu, editors, *Selected Areas in Cryptography*, volume 7707 of *Lecture Notes in Computer Science*, pages 149–165. Springer Berlin Heidelberg, 2013.

[28] Florian Heß. Pairing lattices. In *Proc. of Pairing 2008*, volume 5209 of *LNCS*, pages 18–38, 2008.

[29] Florian Hess, Nigel P. Smart, and Frederik Vercauteren. The eta pairing revisited. *IEEE Transactions on Information Theory*, 52(10):4595–4602, 2006.

[30] Antoine Joux. A new index calculus algorithm with complexity l(1/4+o(1)) in small characteristic. In Tanja Lange, Kristin Lauter, and Petr LisonÄŻk, editors, *Selected Areas in Cryptography – SAC 2013*, volume 8282 of *Lecture Notes in Computer Science*, pages 355–379. Springer Berlin Heidelberg, 2014.

[31] Koray Karabina. Squaring in cyclotomic subgroups. *Math. Comput.*, 82(281), 2013.

[32] Donald E. Knuth. *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1997.

[33] N. Koblitz and A. Menezes. Pairing-based cryptography at high security levels. *Cryptography and coding*, 3796:13–36, 2005.

[34] Kristin E. Lauter, Peter L. Montgomery, and Michael Naehrig. An analysis of affine coordinates for pairing computation. In Marc Joye, Atsuko Miyaji, and Akira Otsuka, editors, *Pairing-Based Cryptography - Pairing 2010 - 4th International Conference, Yamanaka Hot Spring, Japan, December 2010.*

*Proceedings*, volume 6487 of *Lecture Notes in Computer Science*, pages 1–20. Springer, 2010.

[35] Duc-Phong Le and ChikHow Tan. Speeding up ate pairing computation in affine coordinates. In Taekyoung Kwon, Mun-Kyu Lee, and Daesung Kwon, editors, *Information Security and Cryptology -ĂŞ ICISC 2012*, volume 7839 of *Lecture Notes in Computer Science*, pages 262–277. Springer Berlin Heidelberg, 2013.

[36] Eunjeong Lee, Hyang-Sook Lee, and Cheol-Min Park. Efficient and generalized pairing computation on abelian varieties. *Information Theory, IEEE Transactions on*, 55(4):1793–1803, April 2009.

[37] Rudolf Lidl and Harald Niederreiter. *Finite Fields.* Cambridge University Press, 1997.

[38] Chae Hoon Lim and Pil Joong Lee. A key recovery attack on discrete log-based schemes using a prime order subgroup. In *CRYPTO 1997*, pages 249–263. Springer-Verlag, 1997.

[39] Seiichi Matsuda, Naoki Kanayama, Florian Heß, and Eiji Okamoto. Optimised versions of the Ate and twisted Ate pairings. *IEICE Transactions*, 92-A(7):1660–1667, 2009.

[40] S.V. Miller. The weil pairing, and its efficient calculation. *Journal of cryptology*, vol. 17(4), pp. 235-261, 2004.

[41] P. L. Montgomery. Speeding the pollard and elliptic curve methods of factorization. *Mathematics of Computation*, 48(177):243–264, 1987.

[42] Peter L. Montgomery. Modular multiplication without trial division. *Mathematics of Computation*, 44(170):519–521, 1985.

[43] Michael Naehrig, Ruben Niederhagen, and Peter Schwabe. New software speed records for cryptographic pairings. In *LATINCRYPT 2010*, volume 6212 of *LNCS*, pages 109–123. Springer, 2010.

[44] Yasuyuki Nogami, Masataka Akane, Yumi Sakemi, Hidehiro Katou, and Yoshitaka Morikawa. Integer variable chi-based ate pairing. In *Pairing-Based Cryptography - Pairing 2008*, pages 178–191, 2008.

[45] National Institute of Standard and technology. Key management, 2007.

[46] Microsoft Research. Msr ecclib v2.0. 2015.

[47] Franck Rondepierre. Revisiting atomic patterns for scalar multiplications on elliptic curves. In Aurélien Francillon and Pankaj Rohatgi, editors, *Smart Card Research and Advanced Applications*, volume 8419 of *Lecture Notes in Computer Science*, pages 171–186. Springer International Publishing, 2014.

[48] Oliver Schirokauer. The number field sieve for integers of low weight. *Math. Comput.*, 79(269):583–602, 2010.

[49] Michael Scott and Paulo S. L. M. Barreto. Compressed pairings. In *Advances in cryptology—CRYPTO 2004*, volume 3152 of *Lecture Notes in Comput. Sci.*, pages 140–156. Springer, Berlin, 2004.

[50] Michael Scott, Naomi Benger, Manuel Charlemagne, Luis J. Dominguez Perez, and Ezekiel J. Kachisa. On the final exponentiation for calculating pairings on ordinary elliptic curves. In *Pairings 2009*, volume 5671 of *LNCS*, pages 78–88. Springer, 2009.

[51] Jerome A. Solinas. Generalized mersenne numbers. Technical report, 1999.

[52] Martijn Stam and Arjen K. Lenstra. Efficient subgroup exponentiation in quadratic and sixth degree extensions. In *Cryptographic Hardware and Embedded Systems - CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002, Revised Papers*, pages 318–332, 2002.

[53] Thomas Unterluggauer and Erich Wenger. Efficient pairings and ecc for embedded systems. In Lejla Batina and Matthew Robshaw, editors, *Cryptographic Hardware and Embedded Systems ĂŞ CHES 2014*, volume 8731 of *Lecture Notes in Computer Science*, pages 298–315. Springer Berlin Heidelberg, 2014.

[54] Frederik Vercauteren. Optimal pairings. *IEEE Transactions of Information Theory*, 56:455–461, 2009.

# A Appendix

In this appendix, we prove some results on the required conditions for an element to be a square or a cube in $\mathbb{F}_p$ or $\mathbb{F}_{p^2}$, if $p$ is a BN prime.

**Proposition 1.** *Let $p = 36u^4 + 36u^3 + 24u^2 + 6u + 1$ be a BN prime, we have*

*i)* *$-1$ is a square in $\mathbb{F}_p$ if and only if $u$ is even.*

*ii)* *$2$ is a square in $\mathbb{F}_p$ if and only if $u = 0$ or $1$ modulo $4$.*

*iii)* *$3$ is a square in $\mathbb{F}_p$ if and only if $u$ is even.*

*iv)* *$5$ is a square in $\mathbb{F}_p$ if and only if $u = 0$ or $4$ modulo $5$.*

*Proof.*

i) Since $p = 1 + 2u \bmod 4$, this is a direct consequence of the classical result stating that $-1$ is a square in $\mathbb{F}_p$ if and only if $p = 1$ modulo 4.

ii) It is well known that 2 is a square in $\mathbb{F}_p$ if and only if $p = \pm 1 \bmod 8$. Since $p = 1 + 6u \bmod 8$, 2 is a square in $\mathbb{F}_p$ if and only if $u = 0$ or $1 \bmod 4$.

iii) It is not difficult to prove, using the quadratic reciprocity law, that 3 is a square in $\mathbb{F}_p$ if and only if $p = \pm 1 \bmod 12$. Since $p = 1 + 6u \bmod 12$, this holds when $u$ is even.

iv) Again, 5 is a square if and only if $p = \pm 1 \bmod 5$ and it is easy to check that this only holds when $u = 0$ or $4 \bmod 5$.

□

**Proposition 2.** *Let $p = 36u^4 + 36u^3 + 24u^2 + 6u + 1$ be a BN prime, we have*

1. *2 is a cube in $\mathbb{F}_p$ if and only if $u = 0$ mod 3.*
2. *3 is a cube in $\mathbb{F}_p$ if and only if $u = 0, 1$ or 5 modulo 9.*
3. *5 is a cube in $\mathbb{F}_p$ if and only if $u = 0, 2, 4, 6$ or 8 modulo 15.*

*Proof.*
This result is based on two old statements. The first one is a theorem of Fermat stating that every prime $p = 1 \bmod 3$ can be written in the form $a^2 + 3b^2$ and that this representation is unique (up to signs). The second one is known as Euler's conjectures and is proven by Cox in [19]. In the cases interesting us, it states that if $p = a^2 + 3b^2$ as above, we have

- 2 is a cube in $\mathbb{F}_p$ if and only if $3|b$.
- 3 is a cube in $\mathbb{F}_p$ if and only if $9|b$ or $9|a \pm b$.
- 5 is a cube in $\mathbb{F}_p$ if and only if $15|b$ or $3|b, 5|a$ or $15|a \pm b$ or $15|2a \pm b$.

BN primes are congruent to 1 modulo 3 and the Fermat representation can be explicitly given by

$$p = 36u^4 + 36u^3 + 24u^2 + 6u + 1 = \left(6u^2 + 3u + 1\right)^2 + 3u^2.$$

So we have $b = u$ and $a = 6u^2 + 3u + 1$ and it is not difficult to find for which values of $u$ the conditions of Euler's conjectures are satisfied. □

**Proposition 3.** *Let $p$ be an odd prime number which is equal to 1 modulo 3. Then an element is a square (resp. a cube) in $\mathbb{F}_{p^2}$ if and only if its norm is a square (resp. a cube) in $\mathbb{F}_p$.*

*Proof.*
This is not a difficult proof and it can be done by an undergraduate student. It is for example trivial that, if an element is a square (resp. a cube) then its norm is a square (resp. a cube). Let us now reciprocally assume that $a \in \mathbb{F}_{p^2}$ and $N_{\mathbb{F}_{p^2}/\mathbb{F}_p} = b^2$ for some $b$ in $\mathbb{F}_p$. To prove that $a$ is a square, we first need to determine the number of squares in $\mathbb{F}_{p^2}$. For this, we study the linear map

$$\begin{array}{rcl} \psi : \mathbb{F}_{p^2}^* & \to & \mathbb{F}_{p^2}^* \\ x & \mapsto & x^2. \end{array}$$

The kernel of $\psi$ is of course $\{\pm 1\}$ so that its image has cardinality $\frac{p^2-1}{2}$. This means that there are exactly $\frac{p^2-1}{2}$ squares in $\mathbb{F}_{p^2}^*$. Let us now prove that $a$ is one of them. Since the conjugate of $a$ is nothing but $a^p$, the norm of $a$ is $a^{p+1}$. So we have $a^{p+1} = b^2$ and we can raise this equality to the power of $\frac{p-1}{2}$ because $p$ is odd so we get

$$a^{\frac{p^2-1}{2}} = b^{p-1} = 1.$$

This means that $a$ is in the kernel of the linear map

$$\begin{array}{rcl} \varphi : \mathbb{F}_{p^2}^* & \to & \mathbb{F}_{p^2}^* \\ x & \mapsto & x^{\frac{p^2-1}{2}}. \end{array}$$

The kernel of this map is the set of all roots of the polynomial $X^{\frac{p^2-1}{2}} - 1$ and there are at most $\frac{p^2-1}{2}$ such roots since $\mathbb{F}_{p^2}$ is a field. On the other hand, because of Lagrange theorem, all the squares of $\mathbb{F}_{p^2}^*$ are in $\ker \varphi$ and we saw that there are $\frac{p^2-1}{2}$ of them. This means $\ker \varphi$ is exactly the set of squares in $\mathbb{F}_{p^2}^*$ so that $a$ is a square.

The proof is essentially the same for cubes where the kernel of $\psi$ is made of third roots of unity which are all lying in $\mathbb{F}_{p^2}$ since $3|p^2-1$. The condition $p = 1 \bmod 3$ is necessary to raise $a$ to the $\frac{p-1}{3}$. □

**Corollary 1.** *Let $p$ be an odd prime number and $n \in \mathbb{F}_p$, then $n$ is always a square in $\mathbb{F}_{p^2}$.*

*Proof.* $N_{\mathbb{F}_{p^2}/\mathbb{F}_p}(n) = n^2$ is a square in $\mathbb{F}_p$ so that $n$ is a square in $\mathbb{F}_{p^2}$ according to proposition 3. □